ANALISIS EFISIENSI PELABELAN DATA DETEKSI PRODUK RITEL MENGGUNAKAN PENDEKATAN *SEMI-SUPERVISED LEARNING* BERBASIS *COMPUTER VISION* DENGAN YOLOV12



Disusun Oleh:

N a m a : Muhammad Faiz Jauzah
NIM : 22523262

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2026

HALAMAN PENGESAHAN DOSEN PEMBIMBING

ANALISIS EFISIENSI PELABELAN DATA DETEKSI PRODUK RITEL MENGGUNAKAN PENDEKATAN *SEMI-SUPERVISED LEARNING* BERBASIS *COMPUTER VISION* DENGAN YOLOV12

TUGAS AKHIR

Disusun Oleh:

N a m a	: Muhammad Faiz Jauzah
NIM	: 22523262

Yogyakarta, 1 Juli 2026

Pembimbing 1,	Pembimbing 2,
(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)	(Mukhammad Andri Setiawan, S.T., M.Sc., Ph.D.)

HALAMAN PENGESAHAN DOSEN PENGUJI

ANALISIS EFISIENSI PELABELAN DATA DETEKSI PRODUK RITEL MENGGUNAKAN PENDEKATAN *SEMI-SUPERVISED LEARNING* BERBASIS *COMPUTER VISION* DENGAN YOLOV12

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 23 Juli 2026

Tim Penguji

Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. _____

Anggota 1

Dr. Syarif Hidayat, S.Kom., M.I.T. _____

Anggota 2

Elyza Gustri Wahyuni, S.T., M.Cs. _____

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia

(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Muhammad Faiz Jauzah

NIM : 22523262

Tugas akhir dengan judul:

ANALISIS EFISIENSI PELABELAN DATA DETEKSI PRODUK RITEL MENGGUNAKAN PENDEKATAN *SEMI-SUPERVISED LEARNING* BERBASIS *COMPUTER VISION* DENGAN YOLOV12

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 28 Juni 2026



(Muhammad Faiz Jauzah)

HALAMAN PERSEMBAHAN

Laporan tugas akhir ini dipersembahkan kepada seluruh keluarga peneliti, yaitu ayah Jupat Sukaimi, S.E., ibu Faridah Aryani, S.Psi., dan juga adik Meisya Farah Jasmine, yang selalu mendukung dan mengingatkan peneliti dalam pengerjaan laporan tugas akhir ini. Laporan tugas akhir ini juga peneliti persembahkan kepada teman-teman yang telah menemani dan berjuang bersama, dari sebelum pengerjaan dimulai hingga laporan ini selesai, dan semoga akan terus berlanjut untuk ke depannya. Selain itu, peneliti juga mempersembahkan laporan tugas akhir ini kepada diri peneliti sendiri sebagai bentuk apresiasi atas segala usaha, kerja keras, dan kemampuan maksimal yang telah dilakukan hingga tugas akhir ini berhasil diselesaikan.

HALAMAN MOTO

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

- (Q.S. Al-Insyirah: 5-6)

"*No effort is ever wasted. Do not imagine that you 'wasted your time' because something didn't turn out right.*"

- Neale Donald Walsch

"*I think, therefore I am.*"

- Rene Descartes

"*Believe you can and you're halfway there.*"

- Theodore Roosevelt

"*I have not failed. I've just found 10,000 ways that won't work.*"

- Thomas Edison

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Puji syukur senantiasa peneliti panjatkan ke hadirat Allah SWT, karena atas limpahan rahmat, karunia, serta petunjuk-Nya, peneliti dapat menyelesaikan penyusunan laporan tugas akhir yang berjudul "Analisis Efisiensi Pelabelan Data Deteksi Produk Ritel Menggunakan Pendekatan *Semi-Supervised Learning* Berbasis *Computer Vision* dengan YOLOv12" dengan baik.

Penyusunan tugas akhir ini dilaksanakan sebagai salah satu syarat untuk memperoleh gelar Sarjana (S1) pada Program Studi Informatika. Penelitian ini pada dasarnya bertujuan untuk mencari cara agar sistem komputer dapat lebih mudah dan efisien dalam mengenali produk-produk ritel secara otomatis, dengan harapan dapat memberikan manfaat bagi pengembangan teknologi kecerdasan buatan ke depannya. Peneliti menyadari bahwa perjalanan dalam menyusun laporan ini bukanlah proses yang mudah, mengingat adanya berbagai kendala seperti naik-turunnya motivasi, tantangan dalam membagi waktu, hingga rasa jenuh di tengah panjangnya proses pengerjaan. Namun, berkat doa, dukungan, serta bimbingan dari berbagai pihak, segala hambatan tersebut pada akhirnya dapat peneliti lalui. Oleh karena itu, pada kesempatan yang berharga ini, peneliti ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

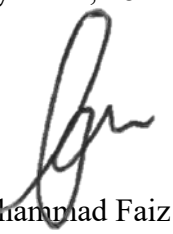
1. Allah SWT, atas segala rahmat, kekuatan, dan kemudahan yang senantiasa diberikan kepada peneliti di setiap langkah dalam menyelesaikan tugas akhir ini.
2. Orang tua dan keluarga yang selalu memberikan dukungan dan doa kepada peneliti agar dapat menyelesaikan kuliah dengan baik.
3. Prof. Dr. Ir. Hari Purnomo, M.T., IPU., ASEAN Eng., selaku Rektor Universitas Islam Indonesia.
4. Prof. Dr. Sri Kusumadewi, S.Si., M.T., selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Jurusan Informatika Universitas Islam Indonesia.
6. Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Ketua Program Studi Informatika Universitas Islam Indonesia.
7. Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Dosen Pembimbing 1, dan Mukhammad Andri Setiawan, S.T., M.Sc., Ph.D., selaku Dosen Pembimbing 2,

yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan bimbingan, arahan, serta motivasi yang sangat berharga selama proses penyusunan skripsi ini.

8. Seluruh Bapak dan Ibu Dosen Informatika Universitas Islam Indonesia yang telah memberikan ilmu pengetahuan yang bermanfaat selama masa perkuliahan.
9. Teman-teman serta semua pihak yang tidak dapat disebutkan satu per satu, terima kasih atas segala bantuan, doa, dan kebersamaannya dari awal perkuliahan hingga tahap ini.

Peneliti menyadari bahwa penyusunan dan penulisan tugas akhir ini masih jauh dari kata sempurna karena keterbatasan pengetahuan dan pengalaman yang peneliti miliki. Oleh karena itu, peneliti sangat terbuka dan mengharapkan segala kritik maupun saran yang membangun dari para pembaca. Akhir kata, peneliti berharap semoga penelitian ini dapat bernilai ibadah, tidak hanya berhenti di lembar laporan ini saja, melainkan memberikan manfaat yang nyata bagi perkembangan ilmu pengetahuan, teknologi, serta bagi siapa saja yang membacanya.

Yogyakarta, 28 Juni 2026



(Muhammad Faiz Jauzah)

SARI

Pengembangan sistem *Computer Vision* pada ruang lingkup ritel, khususnya untuk deteksi produk *Fast-Moving Consumer Goods* (FMCG), masih terhambat oleh tingginya biaya dan waktu yang dibutuhkan untuk anotasi data secara manual. Selain itu, karakteristik produk yang dinamis, tersusun rapat di rak, dan sering mengalami oklusi menuntut model untuk terus diperbarui agar tetap mampu mempertahankan performanya. Kondisi tersebut menyebabkan pendekatan *fully supervised learning* yang sepenuhnya bergantung pada data berlabel menjadi kurang efisien dan sulit diterapkan secara berkelanjutan pada skala yang lebih besar.

Sebagai solusi untuk mengurangi kebutuhan data berlabel dalam jumlah besar, penelitian ini menerapkan pendekatan *Semi-Supervised Learning* (SSL) menggunakan algoritma *Noisy Student* dengan arsitektur YOLOv12. Evaluasi difokuskan untuk mengidentifikasi rasio efisiensi optimal (*sweet spot*) antara jumlah data berlabel dan tidak berlabel melalui pengujian 408 citra produk dalam dua varian standar anotasi, yakni Dataset Parsial (47 kelas) dan Dataset Menyeluruh (49 kelas). Eksperimen dilakukan dengan berbagai variasi skenario rasio data berlabel mulai dari 10% hingga 100%. Pada setiap skenario, model *Teacher* menghasilkan *pseudo-label* untuk data tidak berlabel, kemudian *pseudo-label* tersebut digabungkan dengan data berlabel untuk melatih model *Student*. Selama proses pelatihan, model *Student* diberikan augmentasi *noise* yang lebih agresif untuk meningkatkan ketahanan dan kemampuan generalisasinya.

Hasil evaluasi menunjukkan bahwa penggunaan 50% data berlabel (L50) merupakan rasio pelabelan yang paling efisien dalam penelitian ini. Pada skenario tersebut, model *Student* mencapai nilai mAP@50 sebesar 0,861 pada Dataset Menyeluruh dan 0,854 pada Dataset Parsial. Nilai tersebut setara dengan 97,2% dan 89,9% dari performa model *fully supervised* yang dilatih menggunakan seluruh data berlabel (L100). Hasil ini menunjukkan bahwa pendekatan *Semi-Supervised Learning* dengan algoritma *Noisy Student* mampu mengurangi kebutuhan pelabelan manual hingga 50% tanpa menyebabkan penurunan performa deteksi yang signifikan. Temuan ini memberikan alternatif yang lebih efisien dalam pengembangan sistem deteksi produk berbasis *Computer Vision*, khususnya untuk aplikasi di lingkungan ritel.

Kata kunci: *Computer Vision*, Deteksi Objek Ritel, *Noisy Student*, *Semi-Supervised Learning*, YOLOv12.

GLOSARIUM

Anotasi	proses pemberian label secara manual pada data latih, seperti pembuatan *bounding box* untuk menandai letak produk di dalam gambar.
Bounding Box	kotak pembatas yang digunakan untuk mengidentifikasi dan melokalisasi objek tertentu di dalam citra digital.
Ground Truth	label atau anotasi asli yang dikerjakan dengan benar dan akurat, digunakan sebagai acuan kebenaran untuk mengevaluasi hasil prediksi model.
Noisy Student	algoritma pengembangan dari *Semi-Supervised Learning* di mana model *Teacher* menghasilkan *pseudo-label* untuk melatih model *Student* dengan tambahan gangguan (*noise*) berupa augmentasi agresif untuk meningkatkan ketahanan model.
Oklusi	kondisi tantangan visual di rak ritel nyata di mana suatu produk atau objek terhalang dan saling menutupi sebagian oleh objek lainnya.
Pseudo-label	label semu yang dihasilkan secara otomatis oleh prediksi model *Teacher* pada himpunan data yang tidak memiliki label.
Sweet Spot	titik rasio efisiensi yang paling optimal antara penggunaan data berlabel dan tidak berlabel, di mana peningkatan mAP paling signifikan sebelum memasuki fase kejenuhan metrik.

DAFTAR ISI

HALAMAN JUDUL i
HALAMAN PENGESAHAN DOSEN PEMBIMBING ii
HALAMAN PENGESAHAN DOSEN PENGUJI iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR iv
HALAMAN PERSEMBAHAN v
HALAMAN MOTO vi
KATA PENGANTAR vii
SARI ix
GLOSARIUM x
DAFTAR ISI xi
DAFTAR TABEL xiii
DAFTAR GAMBAR xiv
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah dan Pertanyaan Penelitian 3
1.3 Batasan Masalah dan Lingkup Penelitian 3
1.4 Tujuan dan Manfaat Penelitian 4
1.4.1 Tujuan Penelitian 4
1.4.2 Manfaat Penelitian 4
1.5 Metodologi Penelitian secara Umum 5
1.5.1 Studi Literatur 6
1.5.2 Pengumpulan Data 6
1.5.3 Pra-pemrosesan Data 7
1.5.4 Implementasi Model 7
1.5.5 Evaluasi Model 8
1.5.6 Pembuatan Demo Aplikasi 8
1.5.7 Analisis Hasil 8
1.5.8 Penulisan Laporan 9
1.6 Struktur Laporan 9
BAB II KAJIAN PUSTAKA 11
2.1 Kontekstualisasi Masalah Efisiensi Data pada Deteksi Objek Ritel 11
2.2 Perbandingan Metodologi Deteksi Objek 13
2.3 Teori *Semi-Supervised Learning* sebagai Metodologi Pendeteksian Objek Ritel 14
2.3.1 Arsitektur YOLOv12 (*You Only Look Once* v12) 15
2.3.2 Algoritma *Noisy Student* 17
2.3.3 *Evaluation Metrics* 19
BAB III METODOLOGI PENELITIAN 22
3.1 Penentuan Indikator 23
3.2 Pengumpulan Data 24
3.2.1 Dataset Parsial 26
3.2.2 Dataset Menyeluruh 27
3.3 Pra-pemrosesan Data 29
3.3.1 *Data Annotation* 29
3.3.2 *Data Splitting* 30
3.4 Implementasi Model 32
3.4.1 *Teacher Training* 32
3.4.2 *Pseudo-labeling* 34

3.4.3 *Konfigurasi Dataset Gabungan* 37
3.4.4 *Student Training* 38
3.5 Analisis Hasil 39
3.5.1 Analisis Perbandingan mAP, *Precision*, dan *Recall* antar-skenario 40
3.5.2 Analisis Kurva Konvergensi mAP per *Epoch* 40
3.5.3 Analisis *Training Box Loss* dan *Classification Loss* 41
3.5.4 Analisis Efisiensi dan Identifikasi *Sweet Spot* 42
3.5.5 Analisis Kualitatif Visual Hasil Deteksi 42
3.6 Visualisasi Sederhana 43
BAB IV HASIL DAN PEMBAHASAN 45
4.1 Lingkungan Implementasi 45
4.2 Hasil Pra-pemrosesan Data 45
4.2.1 Hasil Pelabelan Citra 46
4.2.2 Hasil *Data Splitting* 46
4.3 Hasil Implementasi Model 47
4.3.1 Hasil Pelatihan Model *Teacher* 47
4.3.2 Hasil Proses *Pseudo-Labeling* 51
4.3.3 Hasil Pelatihan Model *Student* 54
4.4 Analisis Hasil 59
4.4.1 Perbandingan mAP, *Precision*, dan *Recall* Antar Skenario 59
4.4.2 Kurva Konvergensi mAP per *Epoch* 62
4.4.3 Analisis *Training Box Loss* dan *Classification Loss* 64
4.4.4 Analisis Efisiensi dan Identifikasi *Sweet Spot* 66
4.4.5 Analisis Kualitatif Visual Hasil Deteksi 70
4.5 Demo Aplikasi Visualisasi Sederhana Menggunakan Streamlit 75
4.5.1 Demo Inferensi Gambar 76
4.5.2 Grafik Metrik Pelatihan 77
4.5.3 Perbandingan Semua Skenario 79
BAB V KESIMPULAN DAN SARAN 82
5.1 Kesimpulan 82
5.2 Limitasi 83
5.3 Saran 84
DAFTAR PUSTAKA 87
LAMPIRAN 89

DAFTAR TABEL

Tabel 2.1 Perbandingan Kategori Metodologi Deteksi Objek 14
Tabel 2.2 Perbandingan Arsitektur YOLO 17
Tabel 3.1 Daftar Kelas Dataset Parsial 26
Tabel 3.2 Daftar Kelas Dataset Menyeluruh 28
Tabel 4.1 Spesifikasi Lingkungan Impelementasi 45
Tabel 4.2 Perbandingan Seluruh Skenario Data 47
Tabel 4.3 Hasil Performa Model *Teacher* pada *Train Set* Dataset Parsial 48
Tabel 4.4 Hasil Performa Model *Teacher* pada *Validation Set* Dataset Parsial 48
Tabel 4.5 Hasil Performa Model *Teacher* pada *Test Set* Dataset Parsial 49
Tabel 4.6 Hasil Performa Model *Teacher* pada *Train Set* Dataset Menyeluruh 50
Tabel 4.7 Hasil Performa Model *Teacher* pada *Validation Set* Dataset Menyeluruh 50
Tabel 4.8 Hasil Perfoma Model *Teacher* pada *Test Set* Dataset Menyeluruh 50
Tabel 4.9 Statistik *Pseudo-Labeling* Dataset Parsial 52
Tabel 4.10 Statistik *Pseudo-Labeling* Dataset Menyeluruh 52
Tabel 4.11 Evaluasi Kualitas *Pseudo-Label* pada Dataset Parsial 53
Tabel 4.12 Evaluasi Kualitas *Pseudo-Label* pada Dataset Menyeluruh 53
Tabel 4.13 Hasil Performa Model *Student* pada *Train Set* Dataset Parsial 56
Tabel 4.14 Hasil Performa Model *Student* pada *Validation Set* Dataset Parsial 56
Tabel 4.15 Hasil Performa Model *Student* pada *Test Set* Dataset Parsial 56
Tabel 4.16 Hasil Performa Model *Student* pada *Train Set* Dataset Menyeluruh 58
Tabel 4.17 Hasil Performa Model *Student* pada *Validation Set* Dataset Menyeluruh 58
Tabel 4.18 Hasil Performa Model *Student* pada *Test Set* Dataset Menyeluruh 58
Tabel 4.19 Analisis Delta mAP@50 Model *Student* pada Data Uji Antar-Skenario untuk Identifikasi *Sweet Spot* 68

DAFTAR GAMBAR

Gambar 1.1 Diagram Metodologi secara Umum 6
Gambar 2.1 Diagram Alur Algoritma *Noisy Student* 19
Gambar 3.1 Diagram Alur Penyelesaian 22
Gambar 3.2 Contoh Data Produk Ritel Susu Bubuk Kardus 25
Gambar 3.3 Proses Pelabelan Menggunakan Roboflow 30
Gambar 3.4 Struktur Folder Dataset 32
Gambar 3.5 Kode Konfigurasi *Hyperparameter Teacher* 34
Gambar 3.6 Kode Implementasi *Pseudo-labeling* 35
Gambar 3.7 Kode Penentuan TP, FP, dan FP pada *Pseudo-Label* 36
Gambar 3.8 Kode Evaluasi Nilai mAP pada *Pseudo-Label* 37
Gambar 3.9 Kode Pembuatan Dataset Gabungan 38
Gambar 3.10 Kode Konfigurasi *Hyperparameter Student* 39
Gambar 4.1 Perbandingan mAP@50 Model *Student* pada Data Uji per Skenario 60
Gambar 4.2 Perbandingan mAP@50-95 Model *Student* pada Data Uji per Skenario 61
Gambar 4.3 *Precision* dan *Recall* Model *Student* per Skenario (a) Dataset Parsial dan (b) Dataset Menyeluruh 62
Gambar 4.4 Kurva Konvergensi mAP@50 *Model Student* per Epoch Dataset Parsial pada Skenario Terpilih (L10, L30, L50, L70, L100) 63
Gambar 4.5 Kurva Konvergensi mAP@50 *Model Student* per Epoch Dataset Menyeluruh pada Skenario Terpilih (L10, L30, L50, L70, L100) 64
Gambar 4.6 Kurva *Training Loss* Model *Student* per *Epoch* pada Skenario Terpilih (L10, L30, L50, L100) 66
Gambar 4.7 Perbandingan *Sweet Spot* dengan *Baseline Fully Supervised* 69
Gambar 4.8 Radar Chart Perbandingan L50 (*Sweet Spot*) vs L100 (*Baseline*) (a) Dataset Parsial dan (b) Dataset Menyeluruh 69
Gambar 4.9 Contoh Visual Hasil Deteksi Model *Student* L50 (Kiri = *Ground Truth*, Kanan = Prediksi Model) 73
Gambar 4.10 Tampilan Halaman Utama Aplikasi 75
Gambar 4.11 Tampilan Halaman *Tab* Pertama Demo Inferensi Gambar 77
Gambar 4.12 Tampilan Halaman *Tab* Kedua Grafik Metrik Pelatihan 79
Gambar 4.13 Tampilan Halaman *Tab* Ketiga Perbandingan Semua Skenario 81

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kecerdasan buatan (*Artificial Intellegence*) saat ini sudah berkembang dengan pesat dan salah satu cabang dari kecerdasan buatan yang sering dipakai di bidang industri ritel adalah *Computer Vision*. *Computer Vision* bekerja mirip seperti kerja sistem visual manusia, yaitu sistem dapat mengidentifikasi, memproses, dan menafsirkan visual dari dunia nyata secara otomatis (Wibowo & Corputty, 2024). Salah satu tugas inti dari *Computer Vision* adalah deteksi objek (*object detection*) yang mempunyai tujuan untuk mengidentifikasi dan melokalisasi objek tertentu di dalam citra digital ataupun video (Wang dkk., 2023). Teknologi ini telah digunakan dan menjadi komponen penting untuk industri ritel yang menuntut efisiensi dan akurasi yang tinggi.

Penerapan teknologi deteksi objek dalam lingkup ritel berperan penting dalam modernisasi operasional toko. Teknologi ini dimanfaatkan untuk berbagai kebutuhan, mulai dari manajemen inventaris otomatis, pemantauan ketersediaan produk di rak (*planogram compliance*), hingga sistem kasir mandiri (*self-checkout*) (Song dkk., 2024; Yazdanjouei dkk., 2025). Pentingnya teknologi ini terletak pada kemampuannya untuk mengurangi kesalahan manusia (*human error*), mempercepat proses transaksi, dan memberikan data *real-time* mengenai stok barang (Yazdanjouei dkk., 2025).

Meskipun deteksi objek menawarkan potensi manfaat yang besar, implementasi model *deep learning* dalam bidang *computer vision* masih menghadapi berbagai tantangan teknis, khususnya yang berkaitan dengan ketersediaan dan kualitas data latih. Model deteksi objek konvensional, yang berbasis *supervised learning*, sangat bergantung pada ketersediaan ribuan hingga jutaan gambar yang telah diberi label (*labeled data*) secara akurat untuk dapat mengenali pola dengan baik (Sohn dkk., 2020). Proses anotasi data ini, khususnya pembuatan *bounding box* untuk setiap objek, harus dilakukan secara manual oleh manusia. Sebagai gambaran mengenai besarnya upaya yang diperlukan, anotasi sekitar 164.000 gambar dalam dataset COCO untuk 80 kelas tercatat menghabiskan waktu lebih dari 28.000 jam kerja manusia (Wang dkk., 2023). Hal ini menjadikan proses persiapan data sangat mahal dari segi biaya operasional, menuntut sumber daya manusia yang besar, serta memakan waktu yang sangat lama (*time-consuming*). Ketergantungan pada proses manual ini sering kali menjadi hambatan (*bottleneck)* utama yang menghambat pengembangan sistem cerdas di dunia nyata.

Permasalahan ketersediaan data ini menjadi jauh lebih kompleks ketika diterapkan secara spesifik pada lingkungan ritel yang tidak hanya dinamis, tetapi juga menantang secara visual. Produk-produk ritel atau *Fast-Moving Consumer Goods* (FMCG) memiliki siklus hidup kemasan yang cepat berubah, mulai dari pembaruan desain (*rebranding*), peluncuran edisi spesial musiman, hingga penambahan varian baru dalam tempo singkat (Hou & Huang, 2025). Tantangan ini diperberat oleh kondisi nyata di rak pertokoan (*in-store environment*), di mana produk sering kali disusun sangat rapat atau berdempetan (*dense objects*), saling menutupi satu sama lain (*occlusion*), serta dipengaruhi oleh variasi pencahayaan dan sudut pengambilan gambar yang tidak ideal (Zangana, 2025). Kombinasi antara perubahan tampilan kemasan dan kompleksitas visual tersebut menyebabkan model deteksi objek yang telah dilatih dengan biaya mahal menjadi cepat usang (*obsolete*) atau gagal melakukan deteksi akurat pada kondisi rak yang padat. Kondisi tersebut menyebabkan inefisiensi dalam pengelolaan sistem karena pengelola ritel harus melakukan pengumpulan data secara berkelanjutan sekaligus pelabelan yang teliti pada citra yang kompleks agar kinerja sistem tetap optimal.

Dalam konteks ritel yang menuntut kecepatan, inefisiensi ini menjadi masalah penting yang tidak dapat diabaikan. Mengandalkan metode *fully supervised learning* sepenuhnya menjadi strategi yang tidak berkelanjutan (*unsustainable*) secara ekonomi maupun waktu (Varadarajan & Srivastava, 2020). Oleh karena itu, terdapat kebutuhan mendesak akan sebuah pendekatan penelitian yang mampu meminimalkan ketergantungan pada data berlabel yang sulit didapat, namun tetap mampu memanfaatkan melimpahnya data gambar produk tanpa label untuk menghasilkan performa deteksi yang optimal.

Sebagai solusi atas permasalahan efisiensi data tersebut, penelitian ini mengusulkan penerapan metode *Semi-Supervised Learning* (SSL) dengan algoritma *Noisy Student*. Pendekatan ini memungkinkan model untuk belajar dari sejumlah kecil data berlabel dan sejumlah besar data tidak berlabel, sehingga dapat menurunkan biaya dan waktu anotasi yang dibutuhkan. Penelitian ini akan menggunakan arsitektur YOLOv12 (*You Only Look Once* versi 12) yang dikenal memiliki keseimbangan antara akurasi dan kecepatan komputasi (Tian dkk., 2025). Secara spesifik, penelitian ini difokuskan untuk mencari rasio optimal (*sweet spot*) antara jumlah data berlabel dan tidak berlabel untuk mengetahui seberapa sedikit data berlabel yang diperlukan agar model tetap dapat mendeteksi objek ritel dengan akurasi yang dapat diterima.

1.2 Rumusan Masalah dan Pertanyaan Penelitian

Berdasarkan latar belakang yang telah diuraikan sebelumnya, dapat ditarik beberapa permasalahan sebagai berikut:

a. Penyediaan dataset produk ritel yang teranotasi lengkap (*fully labeled*) membutuhkan alokasi biaya dan waktu yang sangat besar (mencapai ribuan jam kerja), sehingga menjadi hambatan utama dalam penerapan model deteksi objek yang efisien di lingkungan ritel.
b. Produk ritel memiliki karakteristik kemasan yang sangat dinamis (sering mengalami *rebranding* atau perubahan musiman) serta tantangan visual yang kompleks (objek berdempetan dan oklusi) yang menyebabkan model *supervised learning* konvensional menjadi cepat usang dan tidak fleksibel karena harus terus menunggu proses pelabelan ulang secara manual.
c. Belum diketahuinya rasio efisiensi (*sweet spot*) penggunaan jumlah data berlabel yang minim dibandingkan dengan data tidak berlabel yang melimpah pada arsitektur YOLOv12 untuk mempertahankan akurasi deteksi yang optimal menggunakan metode *Semi-Supervised Learning*.

Berdasarkan rumusan masalah di atas, pertanyaan penelitian yang akan dijawab dalam penelitian ini adalah "Berapa rasio optimal antara data berlabel dan tidak berlabel dalam penerapan metode *Semi-Supervised Learning* berbasis YOLOv12 untuk mempertahankan akurasi deteksi produk ritel yang efisien?"

1.3 Batasan Masalah dan Lingkup Penelitian

Penelitian ini difokuskan pada aspek-aspek berikut untuk menjawab pertanyaan penelitian secara spesifik dan terukur:

a. Objek deteksi difokuskan pada produk ritel berupa *Fast-Moving Consumer Goods* (FMCG), spesifiknya pada kategori susu bubuk kemasan kardus (*carton box*).
b. Total dataset yang digunakan berjumlah 408 citra dengan dua varian standar anotasi yang mencakup 47-49 kelas kategori produk. Data ini dibagi menjadi dua kelompok utama untuk simulasi *Semi-Supervised Learning*, yaitu data berlabel (*labeled*) dan data tidak berlabel (*unlabeled*).
c. Penelitian ini menerapkan arsitektur *You Only Look Once* versi 12 (YOLOv12) sebagai model deteksi objek. Penelitian tidak membandingkan kinerja antar versi YOLO (seperti v8 atau v10), melainkan fokus pada optimasi pemanfaatan data pada YOLOv12.

d. Pendekatan *Semi-Supervised Learning* yang diterapkan dibatasi pada penggunaan metode algoritma *Noisy Student*, di mana model Teacher (guru) menghasilkan label semu (*pseudo-labels*) untuk melatih model *Student* (murid) yang lebih besar atau setara.
e. Eksperimen difokuskan pada pencarian rasio efisiensi data melalui sepuluh skenario pembagian data berlabel dan data tidak berlabel, yaitu 10:90 hingga 100:0 dengan kenaikan 10% pada setiap skenario. Rasio 100:0 (*fully supervised*) digunakan sebagai *baseline* untuk mengevaluasi efektivitas pendekatan *Semi-Supervised Learning*.
f. Performa model ditentukan dan dianalisis berdasarkan metrik standar deteksi objek, yaitu *mean Average Precision* (mAP), *Precision*, dan *Recall* untuk mengukur akurasi deteksi.

1.4 Tujuan dan Manfaat Penelitian

1.4.1 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah dipaparkan, tujuan dari penelitian ini adalah:

a. Menerapkan metode *Semi-Supervised Learning* dengan pendekatan algoritma *Noisy Student* pada arsitektur YOLOv12 untuk menyelesaikan permasalahan deteksi produk ritel yang memiliki variasi kemasan tinggi.
b. Menganalisis dan menentukan rasio optimal (*sweet spot*) antara penggunaan data berlabel dan data tidak berlabel yang mampu menghasilkan akurasi deteksi yang paling optimal, untuk meminimalkan kebutuhan proses anotasi manual.

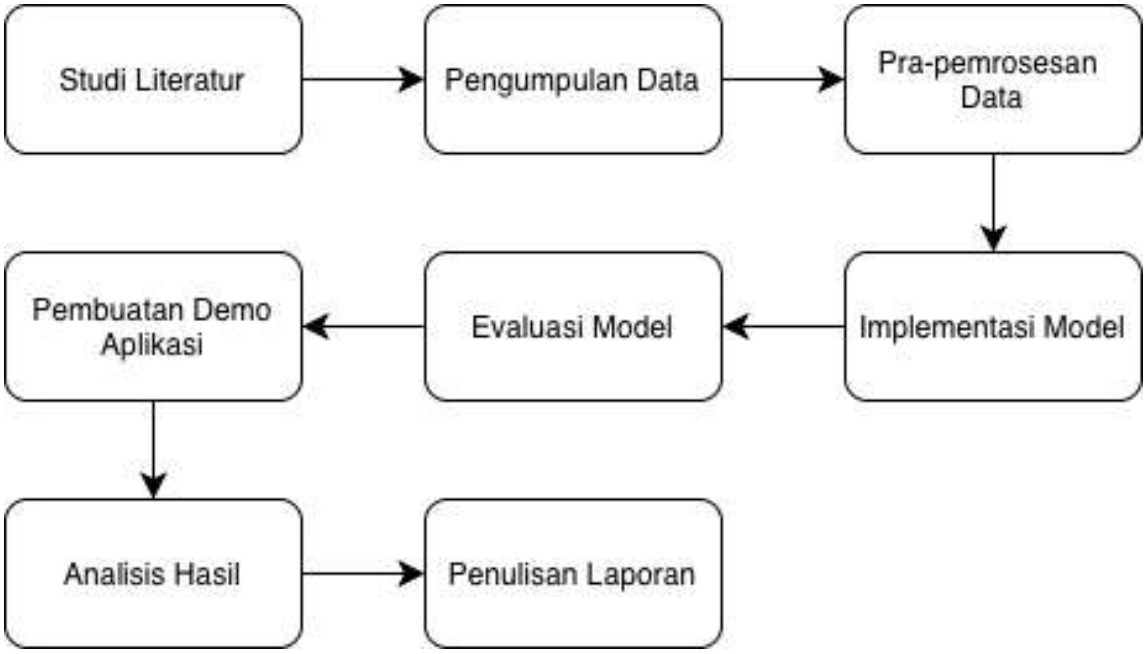
1.4.2 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

a. Memberikan rekomendasi strategi efisiensi pelabelan bagi pengembang sistem cerdas di industri ritel, khususnya mengenai cara memanfaatkan data tidak berlabel untuk meningkatkan kinerja model.
b. Menawarkan solusi praktis yang dapat mengurangi biaya dan waktu operasional dalam tahap persiapan data (*data preparation*) bagi implementasi teknologi *Computer Vision* di ruang lingkup ritel tanpa mengorbankan performa deteksi secara drastis.

1.5 Metodologi Penelitian secara Umum

Untuk mencapai tujuan penelitian yang telah dirumuskan dan menjawab pertanyaan penelitian mengenai rasio optimal data pada metode *Semi-Supervised Learning*, penelitian ini dilaksanakan melalui serangkaian tahapan yang sistematis dan terstruktur. Tahapan penelitian ini diawali dengan Studi Literatur untuk mengkaji berbagai referensi ilmiah terdahulu terkait deteksi objek produk ritel, arsitektur YOLOv12, dan algoritma *Noisy Student*. Selanjutnya, dilakukan tahap Pengumpulan Data berupa citra ritel produk susu bubuk kardus. Data yang telah terkumpul kemudian masuk ke tahap Pra-pemrosesan Data, yang mencakup proses anotasi (pemberian *bounding box*) dan pembagian dataset (*data splitting*) menjadi beberapa skenario rasio data berlabel dan tidak berlabel. Setelah data siap, alur dilanjutkan ke tahap Implementasi Model, di mana model *Teacher* dan *Student* dilatih menggunakan algoritma *Noisy Student* berbasis YOLOv12. Model yang telah dilatih kemudian diuji pada tahap Evaluasi Model untuk mengukur tingkat akurasi dan lokalisasinya menggunakan metrik standar seperti *Mean Average Precision* (mAP), *Precision*, dan *Recall*. Untuk memfasilitasi pengujian visual yang interaktif, dilakukan tahap Pembuatan Demo Aplikasi menggunakan *framework* web sederhana. Seluruh hasil metrik dan inferensi dari aplikasi tersebut kemudian dikaji secara mendalam pada tahap Analisis Hasil untuk mengidentifikasi titik *sweet spot* atau rasio efisiensi pelabelan yang paling optimal. Rangkaian tahapan ini pada akhirnya ditutup dengan tahap Penulisan Laporan untuk mendokumentasikan seluruh proses, analisis, dan temuan penelitian ke dalam dokumen skripsi yang komprehensif. Gambar 1.1 menyajikan keseluruhan alur metodologi penelitian secara umum.



Gambar 1.1 Diagram Metodologi secara Umum

1.5.1 Studi Literatur

Tahap awal penelitian dimulai dengan melakukan pengumpulan dan pengkajian mendalam terhadap referensi ilmiah yang relevan, meliputi jurnal nasional maupun jurnal internasional. Sesuai dengan batasan kebaruan teknologi, literatur yang dipelajari difokuskan pada publikasi 5 tahun terakhir. Fokus utama kajian literatur ini adalah pada topik deteksi objek di lingkungan ritel (*retail object detection*). Secara spesifik, yang dicari dari literatur tersebut adalah analisis mengenai bagaimana penelitian-penelitian terdahulu menghadapi dan menyelesaikan tantangan karakteristik data ritel, seperti masalah oklusi (objek saling menutupi), variasi kemasan yang dinamis, serta kendala efisiensi pelabelan data. Selain itu, studi ini juga menggali perkembangan arsitektur YOLO dan metode *Semi-Supervised Learning* untuk memetakan posisi penelitian ini dibandingkan solusi yang sudah ada.

1.5.2 Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data internal perusahaan dan bersifat rahasia (*confidential*). Kumpulan citra tersebut diambil secara langsung menggunakan perangkat *mobile* pada beberapa toko ritel. Dataset ini terdiri dari 408 citra digital yang menampilkan berbagai skenario susunan rak produk susu bubuk kemasan kardus. Dalam implementasinya, dataset ini dibagi menjadi dua varian dengan standar kualitas anotasi yang berbeda. Varian pertama adalah Dataset Parsial yang mencakup 47 kelas produk dengan

standar anotasi yang longgar dan tidak semua objek susu bubuk kardus yang ada di dalam citra teranotasi. Varian kedua adalah Dataset Menyeluruh yang mencakup 49 kelas produk dengan standar anotasi yang lebih ketat dan seluruh objek susu bubuk kardus yang ada di dalam citra teranotasi. Keberadaan label lengkap dari kedua varian dataset ini difungsikan sebagai *ground truth* untuk keperluan validasi dan pengujian akhir model. Sementara itu, untuk keperluan simulasi eksperimen *Semi-Supervised Learning*, sebagian label dari himpunan data latih disembunyikan secara sengaja saat tahap pelatihan untuk diperlakukan murni sebagai data tidak berlabel (*unlabeled data*) sesuai dengan skenario rasio yang diujikan.

1.5.3 Pra-pemrosesan Data

Tahapan ini dilakukan untuk memastikan dataset siap digunakan secara efektif oleh model. Penelitian ini menggunakan dua varian dataset (Dataset Parsial dan Dataset Menyeluruh) dengan standar kualitas anotasi yang berbeda. Tahapan pra-pemrosesan disesuaikan dengan kebutuhan masing-masing varian. Pada Dataset Parsial (47 kelas), pra-pemrosesan difokuskan pada tahap pembagian data (*data splitting*) karena anotasi bawaan sudah tersedia. Sementara itu, pada Dataset Menyeluruh (49 kelas), tahapan pra-pemrosesan mencakup proses anotasi ulang secara mandiri dan komprehensif menggunakan platform Roboflow. Selanjutnya, pembagian data dilakukan dalam dua tingkatan. Tingkat pertama adalah membagi keseluruhan 408 citra menjadi 80% data latih, 10% data validasi, dan 10% data uji menggunakan metode *Greedy Stratified Sampling* untuk memastikan distribusi kelas yang seimbang. Tingkat kedua difokuskan pada data latih, di mana himpunan data ini dibagi lebih lanjut menjadi data berlabel (*labeled*) dan data tidak berlabel (*unlabeled*) dengan menyembunyikan anotasinya. Hal ini dilakukan untuk menyimulasikan berbagai skenario rasio *Semi-Supervised Learning*, mulai dari rasio 10:90 hingga 100:0 (*fully supervised*).

1.5.4 Implementasi Model

Pada tahap ini, algoritma *Noisy Student* diterapkan pada arsitektur YOLOv12. Proses dimulai dengan melatih model awal yang disebut sebagai model *Teacher* (guru) hanya menggunakan data berlabel yang tersedia sesuai skenario rasio. Model *Teacher* ini kemudian digunakan untuk memprediksi label semu (*pseudo-labels*) pada data tidak berlabel. Selanjutnya, model baru yang disebut model *Student* (murid) dilatih menggunakan gabungan antara data berlabel dan data *pseudo-labels* yang dihasilkan. Proses ini bertujuan agar model *Student* dapat belajar dari volume data yang lebih besar dan meningkatkan generalisasi model.

1.5.5 Evaluasi Model

Evaluasi dilakukan secara komprehensif untuk mengukur performa model di seluruh tahapan eksperimen, yang mencakup data latih (*train set*), data validasi (*validation set*), dan data uji (*test set*). Indikator keberhasilan utama yang digunakan untuk evaluasi dalam penelitian ini adalah metrik *mean Average Precision* (mAP), yang mencakup mAP@50 dan mAP@50-95 untuk mengukur tingkat kepresisian kotak pembatas. Selain mAP, metrik *Precision* dan *Recall* juga digunakan untuk menilai ketepatan prediksi serta kemampuan model dalam mendeteksi objek yang ada. Seluruh hasil evaluasi tersebut digunakan untuk membandingkan performa setiap skenario rasio pelabelan dan menentukan rasio data berlabel dan tidak berlabel yang paling efisien (*sweet spot*) pada pendekatan *Semi-Supervised Learning* yang diterapkan.

1.5.6 Pembuatan Demo Aplikasi

Tahap ini melibatkan pengembangan demo aplikasi visualisasi interaktif berbasis web menggunakan *library* Streamlit. Aplikasi demo ini dirancang dengan antarmuka interaktif yang memuat panel penggeser (*slider*) bagi pengguna untuk memilih persentase rasio data berlabel (mulai dari 10% hingga 100%). Berdasarkan rasio yang dipilih, aplikasi menampilkan hasil ke dalam tiga segmen utama: (1) demo inferensi untuk menjalankan inferensi *bounding box* pada citra baru yang diunggah menggunakan bobot model (*weights*) dari rasio terpilih; (2) visualisasi grafik metrik pelatihan yang merangkum performa deteksi dan penurunan fungsi kerugian (*loss*) dari rasio terpilih; dan (3) analisis komparatif yang menampilkan grafik gabungan dari seluruh skenario rasio secara bersamaan. Antarmuka analisis ini dikembangkan untuk mempermudah identifikasi *sweet spot* secara objektif, yakni titik di mana penambahan data berlabel yang memberikan tingkat efisiensi performa deteksi yang paling maksimal.

1.5.7 Analisis Hasil

Pada tahap ini, dilakukan analisis komparatif yang mendalam terhadap data hasil pengujian dari seluruh skenario. Analisis difokuskan untuk mencari pola hubungan antara jumlah data berlabel dengan tingkat akurasi model untuk mengidentifikasi titik rasio optimal (*sweet spot*). Pembahasan juga mencakup investigasi teknis mengenai alasan peningkatan atau penurunan performa terjadi pada rasio tertentu, serta apakah penambahan data tidak berlabel melalui metode *Noisy Student* efektif dalam mengatasi tantangan spesifik ritel seperti oklusi

dan variasi kemasan. Kesimpulan yang diambil dari analisis ini akan menjawab pertanyaan penelitian yang telah dirumuskan.

1.5.8 Penulisan Laporan

Tahap terakhir dari penelitian ini adalah mendokumentasikan seluruh rangkaian aktivitas penelitian yang telah dilaksanakan ke dalam dokumen Laporan Tugas Akhir. Penyusunan laporan dilakukan secara sistematis mulai dari pendahuluan, kajian pustaka, metodologi penelitian, hasil dan pembahasan, hingga kesimpulan dan saran.

1.6 Struktur Laporan

Sistematika penulisan laporan penelitian ini disusun ke dalam lima bab sebagai berikut:

BAB I PENDAHULUAN

Bab ini membahas latar belakang permasalahan efisiensi pelabelan data ritel, rumusan masalah, pertanyaan penelitian, lingkup penelitian, tujuan dan manfaat, serta metodologi penelitian secara umum.

BAB II KAJIAN PUSTAKA

Bab ini menguraikan kajian terhadap penelitian-penelitian terdahulu yang relevan dan landasan teori mengenai *Semi-Supervised Learning*, algoritma *Noisy Student*, dan arsitektur YOLOv12.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan secara rinci tahapan pelaksanaan penelitian, mulai dari penentuan indikator, pengumpulan dan pra-pemrosesan data, desain skenario SSL, implementasi model *teacher* dan *student*, hingga rencana evaluasi.

BAB IV HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil pengolahan data, keluaran metrik evaluasi dari seluruh proses eksperimen, serta analisis mendalam mengenai penemuan *sweet spot* pada eksperimen yang dilakukan.

BAB V KESIMPULAN DAN SARAN

Bab ini menjelaskan kesimpulan akhir yang menjawab pertanyaan penelitian beserta limitasi studi dan rekomendasi untuk pengembangan penelitian di masa mendatang.

BAB II
KAJIAN PUSTAKA

2.1 Kontekstualisasi Masalah Efisiensi Data pada Deteksi Objek Ritel

Penerapan *computer vision* di industri ritel, seperti pada sistem manajemen stok otomatis dan kasir mandiri (*self-checkout*), menuntut model deteksi objek yang mampu mengenali produk secara akurat dan *real-time*. Salah satu arsitektur deteksi objek yang banyak diandalkan untuk kebutuhan ini adalah YOLO (*You Only Look Once*), yang dikenal karena kecepatan inferensinya. Penelitian terbaru menunjukkan bahwa YOLOv12, dengan integrasi *attention mechanism* ke dalam CNN, mampu mencapai nilai *precision* 0.996 dan F1-score 0.978 pada pendeteksian data plat kendaraan pada kondisi dunia nyata. Hal ini menjadikannya kandidat kuat untuk memprediksi barang secara presisi di lingkungan ritel yang dinamis (Buleu dkk., 2025).

Namun demikian, tantangan terbesar dalam melatih model cerdas seperti YOLO bukanlah pada arsitekturnya, melainkan tingginya ketergantungan pada dataset teranotasi (berlabel) dalam jumlah besar (Chauhan dkk., 2021). Proses pelabelan data secara manual memakan biaya tinggi (*high cost*), waktu yang lama, dan rentan terhadap kesalahan manusia (*human error*). Sebagai gambaran skala kebutuhan anotasi, proses pelabelan 164 ribu gambar pada dataset COCO yang mencakup 80 kelas membutuhkan sekitar 28 ribu jam kerja manusia (Wang dkk., 2023). Beban ini menjadi semakin berat di konteks produk ritel. Kemasan produk sering berubah, produk baru terus bermunculan, dan barang-barang sering tersusun padat di rak sehingga anotasi *bounding box* menjadi sangat rumit (Varadarajan & Srivastava, 2020; Yazdanjouei dkk., 2025). Akibatnya, sangat sulit untuk mendapatkan dataset produk ritel yang benar-benar sempurna dan selalu mutakhir secara manual.

Berbagai penelitian telah mencoba mengatasi keterbatasan data berlabel ini melalui pendekatan yang berbeda-beda. Pada ranah *unsupervised learning*, Wang dkk. (2023) mengusulkan metode *Cut-and-Learn* (CutLER) yang dilatih eksklusif menggunakan 1,3 juta gambar tidak berlabel dari ImageNet tanpa memerlukan anotasi tambahan. Hasilnya menunjukkan bahwa CutLER mampu mendeteksi objek dengan baik selama distribusi data inferensi selaras dengan data pelatihan. Akan tetapi, ketika diterapkan pada domain yang berbeda, performa model mengalami penurunan yang signifikan. Hal ini menunjukkan bahwa pendekatan *unsupervised* murni masih memiliki keterbatasan dalam hal generalisasi lintas domain, yang justru sangat diperlukan di lingkungan ritel dengan variasi produk yang tinggi.

Pendekatan lain yang menjanjikan adalah *contrastive learning* melalui DetCo (Xie dkk., 2021), sebuah *framework pre-training* tak terawasi yang mampu menangkap informasi lokalisasi objek secara lebih baik dibandingkan metode sebelumnya seperti MoCo v2. DetCo terbukti meningkatkan performa deteksi sebesar 0,9 AP pada dataset COCO, dan sangat efektif pada skenario data terbatas, yakni dengan hanya 1–10% data berlabel, model sudah mampu mencapai hasil yang kompetitif. Hal ini membuka peluang pengurangan beban pelabelan secara drastis tanpa mengorbankan akurasi secara signifikan.

Sementara itu, pada pendekatan *weakly supervised*, Varadarajan & Srivastava (2020) mengembangkan sistem berbasis FCN (*Fully Convolutional Network*) yang dilatih menggunakan *synthetic* dataset untuk melokalisasi produk di rak ritel. Hasil penelitian ini menunjukkan bahwa model lebih efektif pada produk dengan bentuk seragam (persegi/kotak) dibandingkan produk dengan bentuk beragam. Di sisi lain, Song dkk. (2024) menggunakan VGG-16 dengan Smooth Grad-CAM++ dan *Curriculum Learning* pada dataset Products-10K (21.874 gambar, 828 kategori) menunjukkan bahwa metode ini unggul dalam mendeteksi jenis dan jumlah produk, meskipun presisi lokalisasi *bounding box*-nya masih terbatas. Kedua studi ini menunjukkan bahwa *weakly supervised* dapat mengurangi kebutuhan anotasi, namun masih perlu digabungkan dengan strategi lain untuk mencapai akurasi lokalisasi yang memadai.

Dari berbagai pendekatan tersebut, *Semi-Supervised Learning* (SSL) muncul sebagai strategi yang paling seimbang antara efisiensi pelabelan dan akurasi deteksi. Lachhab (2023) membuktikan bahwa metode SSL menghasilkan performa yang efektif bahkan ketika data berlabel sangat sedikit. Lebih jauh, Yazdanjouei dkk. (2025) mengimplementasikan *co-training* SSL dengan mengombinasikan dua model (YOLO dan Faster R-CNN) untuk saling menghasilkan *pseudo-label*. Dengan hanya 2.000 gambar berlabel dan 8.000 gambar tak berlabel dari dataset SKU-110k, pendekatan ini menghasilkan skor mAP 0,596 dan AR300 0,627. Hasil ini menunjukkan kemampuan adaptasi tinggi terhadap skenario ritel dunia nyata yang penuh dengan produk *overlapping* dan *occlusion*.

Dari tinjauan berbagai studi di atas, dapat disimpulkan bahwa meskipun pendekatan *unsupervised* dan *weakly supervised* menawarkan reduksi kebutuhan label, keduanya masih memiliki keterbatasan dalam generalisasi dan presisi lokalisasi. SSL, khususnya yang berbasis *co-training* dan *pseudo-labeling*, menjadi pendekatan yang paling relevan untuk mengatasi tantangan efisiensi pelabelan pada deteksi produk ritel. Oleh karena itu, analisis lebih mendalam mengenai efisiensi SSL dalam konteks ini menjadi fokus penelitian ini. Beberapa kriteria yang menjadi dasar penilaian efisiensi tersebut meliputi:

a. Efisiensi penggunaan data berlabel, yakni seberapa jauh model dapat mempertahankan akurasi dengan jumlah label yang minimal.
b. Kemampuan generalisasi terhadap variasi produk ritel, termasuk kemasan yang berubah, produk baru, dan kondisi rak yang padat.
c. Ketahanan terhadap *occlusion* dan *overlapping*, mengingat karakteristik dataset ritel.
d. Biaya komputasi yang wajar, agar solusi dapat diterapkan secara praktis tanpa infrastruktur yang berlebihan.

2.2 Perbandingan Metodologi Deteksi Objek

Setelah memahami konteks dan masalah dalam efisiensi data pada objek ritel, langkah selanjutnya adalah membandingkan berbagai metodologi untuk mendeteksi objek ritel. Tujuannya adalah untuk mengidentifikasi pendekatan mana yang paling sesuai untuk pendeteksian objek ritel. Secara umum, pendekatan pembelajaran dalam deteksi objek dapat dikelompokkan menjadi tiga kategori, yaitu *Supervised Learning*, *Unsupervised Learning,* dan *Semi-Supervised Learning.* (C A Padmanabha Reddy dkk., 2018).

Pendekatan *Supervised Learning* mampu menghasilkan tingkat akurasi deteksi yang sangat tinggi, dengan catatan data yang diberikan memadai (Lachhab, 2023). Namun, metode ini memiliki kelemahan signifikan dalam konteks industri ritel karena menuntut 100% data berlabel. Ketergantungan ini membuat biaya anotasi menjadi sangat tinggi dan memakan waktu lama. Hal ini menjadikannya kurang fleksibel untuk diterapkan pada data ritel yang sangat masif dan dinamis. Sebaliknya, pendekatan *Unsupervised Learning* tidak memerlukan data berlabel sehingga dapat mengurangi kebutuhan anotasi secara signifikan (Jun dkk., 2025). Namun, untuk tugas deteksi objek yang spesifik, pendekatan ini umumnya masih memiliki keterbatasan dalam menghasilkan performa yang sebanding dengan metode yang memanfaatkan data berlabel. Hal tersebut disebabkan oleh tidak adanya informasi label yang dapat digunakan sebagai acuan selama proses pembelajaran, sehingga model lebih sulit membedakan dan memetakan karakteristik visual ke dalam kelas objek yang spesifik (Wang dkk., 2023). Di antara kedua pendekatan tersebut, *Semi-Supervised Learning* (SSL) menawarkan pendekatan yang menggabungkan penggunaan data berlabel dan data tidak berlabel. Dengan memanfaatkan sebagian kecil data berlabel serta sejumlah besar data tidak berlabel, SSL berupaya mempertahankan performa deteksi yang baik sekaligus mengurangi ketergantungan terhadap proses anotasi yang mahal dan memakan waktu.

Berdasarkan karakteristik ketiga pendekatan tersebut, penelitian ini memilih metode *Semi-Supervised Learning* (SSL) karena mampu memanfaatkan data berlabel dan data tidak berlabel secara bersamaan. Pendekatan ini memungkinkan pengurangan kebutuhan data berlabel tanpa mengorbankan performa model secara signifikan. Meskipun proses pelatihannya lebih kompleks dibandingkan *Supervised Learning*, ketersediaan data tidak berlabel yang melimpah di lingkungan ritel menjadikan SSL sebagai pendekatan yang cocok untuk diterapkan pada penelitian ini. Perbandingan karakteristik ketiga kategori metodologi beserta tinjauan literatur yang mendasarinya disajikan pada Tabel 2.1.

Tabel 2.1 Perbandingan Kategori Metodologi Deteksi Objek

Metodologi	**Kebutuhan Label**	**Biaya Anotasi**	**Akurasi Deteksi**	**Kelemahan**
Supervised Learning	Memerlukan 100% data berlabel (Lachhab, 2023).	Sangat Tinggi (Mahal & Lama) (Lachhab, 2023).	Sangat tinggi jika data cukup (Lachhab, 2023).	*Overfitting* pada data sedikit; tidak fleksibel untuk data ritel masif.
Unsupervised Learning	Tidak memerlukan label sama sekali (Jun dkk., 2025; Wang dkk., 2023).	Sangat Rendah (Jun dkk., 2025).	Umumnya rendah untuk tugas deteksi spesifik (klasifikasi tanpa arahan) (Wang dkk., 2023).	Sulit memetakan fitur ke kelas produk spesifik (Xie dkk., 2021).
Semi-Supervised Learning (SSL)	Menggabungkan sedikit data berlabel dan banyak data tidak berlabel (Lachhab, 2023; Yazdanjouei dkk., 2025)	Moderat/Rendah (Efisien) (Lachhab, 2023).	Mendekati s*upervised* dengan data label yang jauh lebih sedikit (Lachhab, 2023).	Kompleksitas pelatihan lebih tinggi (Yazdanjouei dkk., 2025).

2.3 Teori *Semi-Supervised Learning* sebagai Metodologi Pendeteksian Objek Ritel

Subbab ini membahas kerangka kerja teknis yang digunakan untuk menyelesaikan permasalahan kelangkaan data label pada deteksi produk ritel. Solusi yang diajukan menggabungkan arsitektur deteksi objek YOLOv12 dengan strategi pelatihan *Semi-Supervised Learning* (SSL) berbasis algoritma *Noisy Student*.

2.3.1 Arsitektur YOLOv12 (*You Only Look Once* v12)

Bagian ini menjelaskan model dasar yang bertindak sebagai *Teacher* dan *Student* dalam kerangka kerja penelitian. Evolusi arsitektur YOLO dari waktu ke waktu mencerminkan upaya berkelanjutan untuk menyeimbangkan kecepatan inferensi dan akurasi deteksi, di mana setiap versi baru membawa inovasi arsitektural spesifik untuk mengatasi keterbatasan versi pendahulunya (Sapkota dkk., 2025).

YOLOv12 memperkenalkan mekanisme *Attention-Centric* yang memperbaiki kelemahan versi sebelumnya dalam mengekstraksi fitur-fitur halus (Tian dkk., 2025). Berbeda dengan YOLOv8 atau v10 yang sangat bergantung pada CNN *backbone* standar, YOLOv12 mengoptimalkan *receptive field* untuk menangkap konteks visual yang lebih baik. YOLOv8 mengadopsi desain *anchor-free* dengan *decoupled head* yang memisahkan tugas klasifikasi dan regresi, namun masih mengalami kesulitan pada skenario oklusi berat. YOLOv9 mengatasinya melalui mekanisme *Programmable Gradient Information* (PGI) dan *Generalized Efficient Layer Aggregation Network* (GELAN) yang menjaga aliran gradien tetap stabil pada jaringan yang dalam. YOLOv10 selanjutnya menghapus kebutuhan *Non-Maximum Suppression* melalui strategi *dual assignment*, sehingga mengurangi latensi inferensi secara signifikan. YOLOv11 kemudian memperkenalkan blok C3k2 dan modul C2PSA yang menghasilkan kecepatan inferensi yang rendah hingga 2,4 ms, menjadikannya unggul dalam skenario *real-time* (Sapkota dkk., 2025).

Puncak dari evolusi ini adalah YOLOv12 yang mengintegrasikan modul *Area Attention* (A^2) dan *Residual Efficient Layer Aggregation Network* (R-ELAN). Kedua modul tersebut memungkinkan model untuk menangkap konteks global dan lokal secara bersamaan dengan biaya komputasi yang lebih efisien dibanding mekanisme *attention* konvensional (Tian dkk., 2025). Kemampuan ini krusial dalam skenario ritel yang sering kali memiliki latar belakang yang berantakan dan oklusi tinggi (produk saling menutupi), di mana modul *Area Attention* pada YOLOv12 dapat menghasilkan persepsi kontur objek yang lebih jelas dan presisi.

Meski demikian, perlu dicatat bahwa keunggulan akurasi YOLOv12 datang dengan konsekuensi biaya komputasi yang lebih tinggi. Penelitian oleh Putra & Wowor (2025) menunjukkan bahwa YOLOv12 membutuhkan penggunaan memori sekitar 29-34 kali lebih besar dibandingkan YOLOv11 pada ukuran model yang sama (YOLOv12N membutuhkan ~3.674 MB versus YOLOv11N yang hanya ~126 MB), serta waktu inferensi yang sekitar 1,6 kali lebih lama (rata-rata 18,28 ms untuk YOLOv12N vs 11,58 ms untuk YOLOv11N).

Di sisi lain, penerapan YOLO pada konteks pengenalan produk ritel secara khusus juga telah dikaji oleh Wibowo & Corputty (2024), yang menggunakan YOLOv8 sebagai tahap deteksi awal dan berhasil mencapai mAP 77,9% pada 43 kelas produk. Penelitian tersebut menunjukkan bahwa model dari keluarga YOLO secara umum memiliki kemampuan yang baik untuk menangani tugas deteksi produk ritel dengan jumlah kelas yang relatif banyak. Penelitian ini menggunakan YOLOv12 sebagai model deteksi karena menawarkan peningkatan pada kemampuan ekstraksi fitur dan *recall* dibandingkan generasi sebelumnya.

Pemilihan YOLOv12 sebagai arsitektur dalam penelitian ini didasari oleh tiga pertimbangan utama. Pertama, YOLOv12 tercatat memiliki nilai *recall* tertinggi (0,90) dibandingkan generasi YOLO sebelumnya (Sapkota dkk., 2025). Kedua, modul *Area Attention* (A^2) dan R-ELAN pada YOLOv12 dirancang untuk menangkap konteks lokal dan global secara bersamaan (Tian dkk., 2025), yang relevan dengan karakteristik data penelitian ini berupa rak ritel dengan produk yang tersusun rapat dan saling beroklusi. Ketiga, sebagai arsitektur YOLO paling mutakhir pada saat penelitian dilakukan, penggunaan YOLOv12 memberikan kontribusi awal dalam mengevaluasi kompatibilitasnya dengan skema *Semi-Supervised Learning* untuk deteksi objek ritel. Meskipun keunggulan tersebut memiliki konsekuensi berupa kebutuhan memori dan waktu inferensi yang lebih besar dibandingkan YOLOv11 (Putra & Wowor, 2025), *trade-off* ini dapat diterima karena penelitian ini berfokus pada efisiensi pelabelan data, bukan pada efisiensi komputasi saat inferensi. Ringkasan perbandingan karakteristik beberapa versi YOLO disajikan pada Tabel 2.2.

Tabel 2.2 Perbandingan Arsitektur YOLO

Model YOLO	Mekanisme Fokus	Efisiensi Parameter	Kinerja Objek Kecil	Tahun Rilis
YOLOv8	Anchor-free & Decoupled (Sapkota dkk., 2025)	Seimbang (Sapkota dkk., 2025)	Baik (*baseline*): Mampu mendeteksi objek sebagian terlihat, namun kesulitan pada objek yang tertutup (oklusi) berat (Sapkota dkk., 2025).	2023 (Sapkota dkk., 2025)
YOLOv9	PGI & GELAN (Sapkota dkk., 2025)	Tinggi (Sapkota dkk., 2025)	Sangat Baik: Mengatasi kehilangan informasi pada *deep networks*, menjaga akurasi tinggi untuk objek kecil (Sapkota dkk., 2025).	2024 (Sapkota dkk., 2025)
YOLOv10	NMS-Free & Dual Assignment (Sapkota dkk., 2025)	Efisien (Sapkota dkk., 2025)	Presisi Tinggi: Varian YOLOv10x mencapai skor *precision* tertinggi (0.908), namun terkadang gagal mendeteksi pada skenario kompleks (Sapkota dkk., 2025).	2024 (Sapkota dkk., 2025)
YOLOv11	C3k2 & C2PSA (Sapkota dkk., 2025)	Sangat cepat (l*ow latency*) (Sapkota dkk., 2025)	*Robust* (Tahan Gangguan): Sangat baik dalam menangani oklusi dan latar belakang kompleks, serta akurat dalam validasi penghitungan objek (Sapkota dkk., 2025)	2024 (Sapkota dkk., 2025)
YOLOv12	Attention-Centric (A^2 & R-ELAN) (Sapkota dkk., 2025; Tian dkk., 2025)	Unggul (Sapkota dkk., 2025)	*Recall* Tertinggi: Mencapai *recall rate* tertinggi (0.90) dan memiliki persepsi kontur objek yang lebih jelas dibanding pendahulunya (Sapkota dkk., 2025).	2025 (Sapkota dkk., 2025; Tian dkk., 2025)

2.3.2 Algoritma *Noisy Student*

Untuk memanfaatkan data tidak berlabel (*unlabeled data*) yang melimpah, penelitian ini menerapkan algoritma *Noisy Student*. Metode ini merupakan pengembangan dari *self-training* standar dengan penambahan gangguan (*noise*) untuk meningkatkan ketahanan model (Chauhan dkk., 2021). Algoritma *Noisy Student* yang digunakan dalam penelitian ini dapat dilihat pada Gambar 2.1.

Kerangka kerja metode *Semi-Supervised Learning* dalam penelitian ini diawali dengan membagi keseluruhan dataset menjadi tiga bagian utama menggunakan metode *Greedy*

Stratified Sampling untuk memastikan distribusi setiap kelas pada setiap bagian. Hasil pembagian ini terdiri dari 80% data latih (*Train Set*), 10% data validasi (*Validation Set*), dan 10% data uji (*Test Set*). Data validasi dan data uji ini bersifat tetap dan identik untuk seluruh skenario pengujian untuk memastikan bahwa evaluasi performa model konsisten di setiap pengujian.

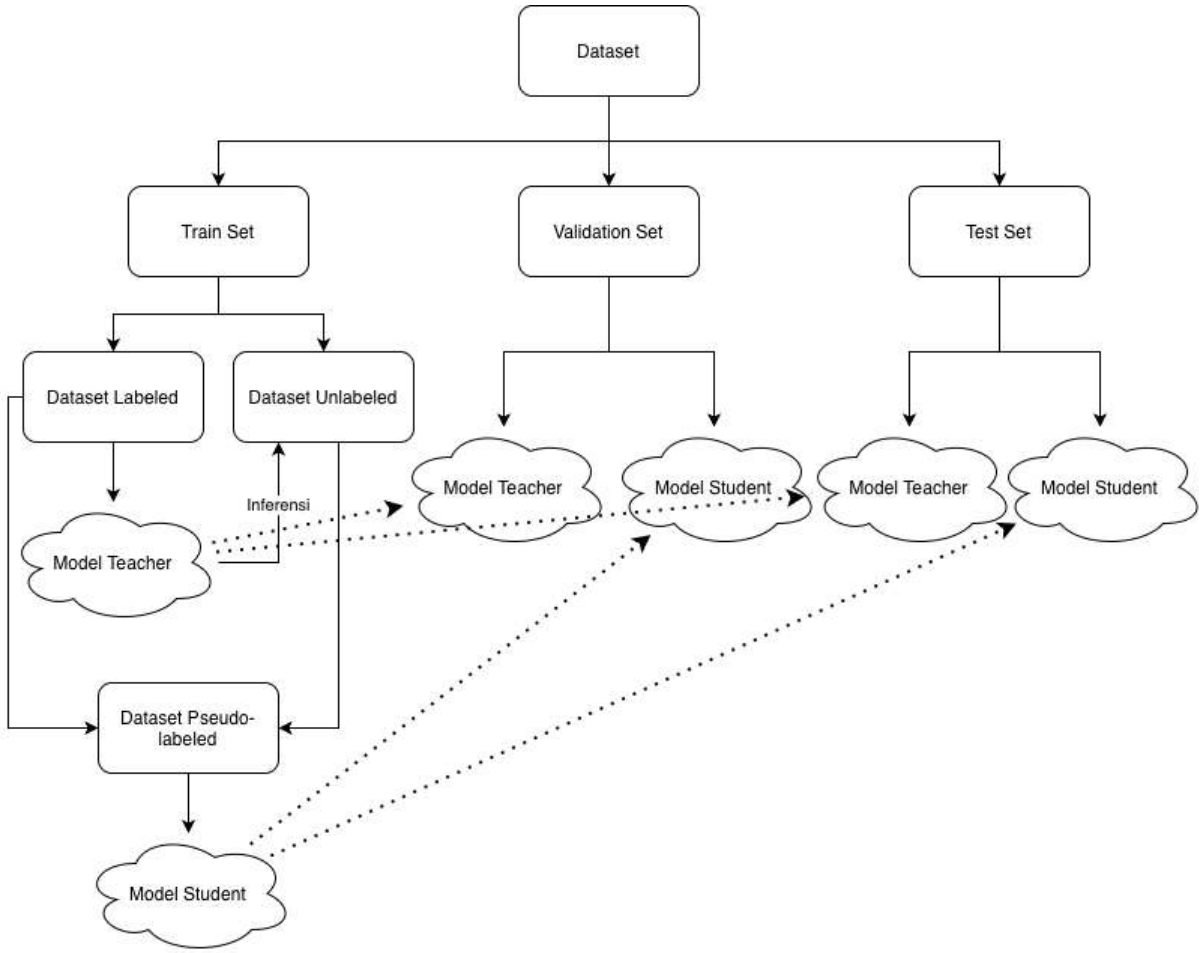
Selanjutnya, data latih dibagi lagi ke dalam sepuluh skenario rasio pelabelan yang berbeda, mulai dari skenario L10 (10% data berlabel dan 90% tidak berlabel) hingga L100 (100% data berlabel). Di dalam setiap direktori skenario, terdapat folder *Dataset Labeled* (data yang memiliki anotasi) dan *Dataset Unlabeled* (data mentah tanpa anotasi). Selain itu, setiap direktori skenario juga memuat salinan folder *Validation Set* dan *Test Set* sebagai acuan evaluasi model.

Proses pelatihan dimulai dengan menginisialisasi model *Teacher* menggunakan model YOLOv12-Nano yang dilatih secara *fully supervised* dengan hanya menggunakan *Dataset Labeled*. Di dalam *framework* YOLOv12, proses validasi dilakukan secara otomatis pada setiap akhir *epoch* menggunakan data validasi (*Validation Set*) untuk memantau perkembangan performa model. Setelah proses pelatihan selesai, model dievaluasi pada data uji (*Test Set*) untuk mengukur kemampuan generalisasinya.

Setelah model ini memiliki kemampuan deteksi dasar, model *Teacher* melakukan inferensi terhadap *Dataset Unlabeled* untuk menghasilkan prediksi kelas dan *bounding box*. Hasil prediksi tersebut kemudian disimpan sebagai label semu (*pseudo-label*) di *Dataset Unlabeled*. Kualitas dari label semu ini dievaluasi menggunakan *Intersection over Union* (IoU) secara terpisah dengan membandingkannya secara langsung terhadap label asli (*ground truth*) dari data tidak berlabel yang sengaja disembunyikan. Selanjutnya, *Dataset Pseudo-labeled* dibuat menggunakan gabungan data berlabel yang ada dengan data hasil inferensi model *Teacher*.

Tahap selanjutnya adalah pelatihan model *Student* yang diinisialisasi ulang dari awal (*cold start*) menggunakan model YOLOv12-Nano untuk mencegah pewarisan bias dari model *Teacher*. Model *Student* dilatih menggunakan *Dataset Pseudo-labeled* yang sudah dibuat. Pada tahap ini, model *Student* belajar untuk menggeneralisasi fitur dengan penambahan augmentasi *noise* yang agresif yang bertujuan untuk melampaui performa model *Teacher*. Sama seperti tahapan sebelumnya, *framework* YOLOv12 secara otomatis mengevaluasi proses pelatihan model *Student* menggunakan data validasi (*Validation Set*) untuk memantau jalannya konvergensi model. Setelah seluruh siklus pelatihan selesai, YOLOv12 menjalankan evaluasi

final pada data uji (*Test Set*). Evaluasi otomatis ini menghasilkan metrik evaluasi akhir berupa *Mean Average Precision* (mAP), *precision*, dan *recall*, yang akan digunakan untuk mengukur efektivitas model pada berbagai skenario rasio pelabelan data ritel. Rangkaian alur dari pembagian data dan penerapan *Noisy Student* ini dapat dilihat secara utuh pada Gambar 2.1.



Gambar 2.1 Diagram Alur Algoritma *Noisy Student*

2.3.3 *Evaluation Metrics*

Mengingat tujuan utama penelitian ini adalah mengidentifikasi titik optimal (*sweet spot*) rasio data, pengukuran keberhasilan model dilakukan dengan mengacu pada standar metrik deteksi objek COCO (*Common Objects in Context*).

a. *Precision* (Presisi)

Presisi mengukur tingkat keakuratan model dalam mengidentifikasi objek positif. Nilai ini merepresentasikan rasio prediksi benar terhadap total prediksi positif yang dihasilkan model, yang dirumuskan dalam Persamaan (2.1).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2.1)$$

Dengan keterangan sebagai berikut:

1. TP (*True Positive*) = Jumlah objek yang dideteksi dengan benar (sesuai *ground truth* dengan IoU > *threshold*).
2. FP (*False Positive*) = Jumlah objek yang dideteksi namun sebenarnya salah (salah kelas atau salah lokasi).

b. *Recall* (Sensitivitas)

Recall mengukur kemampuan model dalam menemukan seluruh objek yang seharusnya terdeteksi dalam dataset. Metrik ini sangat krusial dalam konteks ritel untuk meminimalkan produk yang tidak terhitung (*missed detection*), sebagaimana ditunjukkan pada Persamaan (2.2).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2.2)$$

Dengan keterangan sebagai berikut:

1. FN (*False Negative*) = Jumlah objek asli yang gagal dideteksi oleh model.

c. *Intersection over Union* (IoU)

Intersection over Union (IoU) mengukur tingkat tumpang tindih (*overlap*) antara *bounding box* yang di prediksi oleh model dengan *bounding box ground truth*. Nilai ini digunakan sebagai ambang batas (*threshold*) untuk menentukan apakah sebuah prediksi diklasifikasikan sebagai *True Positive* (TP), *False Positive* (FP), atau *False Negative* (FN). Nilai IoU dirumuskan dengan Persamaan (2.3).

$$\text{IoU} = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (2.3)$$

Dengan keterangan sebagai berikut:

1. *Area of Overlap* = Luas wilayah irisan atau tumpang tindih antara perpotongan koordinat *bounding box* prediksi dengan koordinat *bounding box ground truth*.
2. *Area of Union* = Total wilayah gabungan yang mencakup keseluruhan area *bounding box* prediksi dan *bounding box ground truth*.

d. *Mean Average Precision* (mAP)

Mean Average Precision (mAP) adalah rata-rata dari *Average Precision* (AP) seluruh kelas objek. AP sendiri dihitung sebagai luas area di bawah kurva *Precision-Recall* ($P(R)$). Untuk sekumpulan data dengan N kelas, mAP dihitung menggunakan Persamaan (2.4) dan Persamaan (2.5).

$$\text{AP} = \int_0^1 P(R)\, dr \qquad (2.4)$$

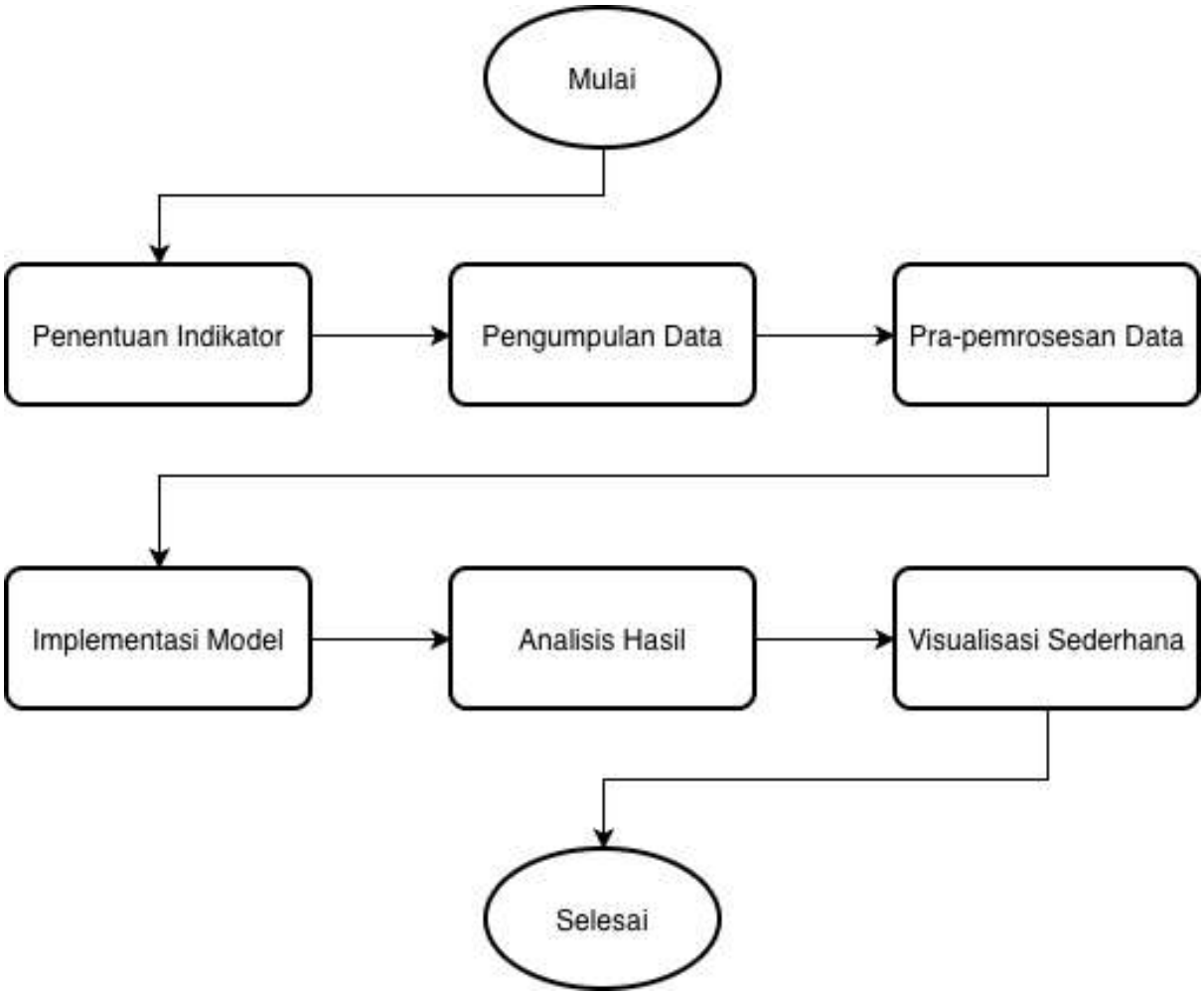
$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} AP_i \qquad (2.5)$$

Dengan keterangan sebagai berikut:

1. N = Jumlah seluruh kelas yang ada.
2. P = Nilai *precision.*
3. R = Nilai *recall.*
4. mAP@50 = Ini adalah standar tradisional (PASCAL VOC). Metrik ini menganggap prediksi "benar" jika irisan area (*Intersection over Union*) antara kotak prediksi dan kotak asli minimal 50%. Ini mengukur apakah model berhasil menemukan objek.
5. mAP@50-95 = Ini adalah standar modern yang lebih ketat. Nilai ini merupakan rata-rata dari mAP yang dihitung pada 10 tingkat IoU berbeda, mulai dari 0.50, 0.55, 0.60, ..., hingga 0.95. Metrik ini mengukur seberapa presisi kotak pembatas yang dibuat model.

BAB III
METODOLOGI PENELITIAN

Penelitian ini berfokus pada penerapan model *Semi-Supervised Learning* (SSL) dengan algoritma *Noisy Student* untuk pendeteksian objek ritel menggunakan arsitektur YOLOv12. Alur penelitian ini dirancang untuk mengevaluasi efektivitas metode tersebut dalam mencari rasio optimal (*sweet spot*) antara data berlabel dan data tidak berlabel. Gambar 3.1 menunjukkan keseluruhan alur metodologi dari penelitian. Tahapan penelitian ini ada enam dan diawali dari Penentuan Indikator, Pengumpulan Data, Pra-pemrosesan Data, Implementasi Model, Analisis Hasil, dan Visualisasi Sederhana.



Gambar 3.1 Diagram Alur Penyelesaian

3.1 Penentuan Indikator

Tujuan utama implementasi model adalah menjawab pertanyaan penelitian mengenai rasio optimal antara data berlabel dan data tidak berlabel yang mampu mempertahankan akurasi deteksi produk ritel. Untuk dapat menjawab tujuan tersebut secara terukur, penelitian ini memerlukan metrik yang mampu merepresentasikan tiga aspek performa, yaitu (1) tingkat kepresisian lokalisasi dan klasifikasi objek, (2) tingkat kelengkapan deteksi terhadap keseluruhan objek yang ada, dan (3) skor komposit yang menggabungkan kedua aspek tersebut sebagai dasar pembandingan antar-skenario. Ketiga aspek inilah yang selanjutnya digunakan melalui metrik *Precision*, *Recall*, dan *mean Average Precision* (mAP). Selain itu, penelitian ini juga bertujuan membandingkan pengaruh standar kualitas anotasi terhadap performa model, yaitu antara Dataset Parsial yang memiliki anotasi lebih longgar dan tidak semua objek teranotasi dengan Dataset Menyeluruh yang memiliki anotasi yang lebih rapat dan semua objek yang ada di dalam citra teranotasi.

Indikator keberhasilan utama adalah menemukan rasio data berlabel (*labeled*) terhadap data tidak berlabel (*unlabeled*) yang paling optimal. Dalam konteks ini, "optimal" didefinisikan sebagai proporsi data berlabel terkecil yang tetap mampu menghasilkan performa deteksi objek yang tinggi. Parameter untuk menentukan performa dan keberhasilan indikator adalah skor *mean Average Precision* (mAP@50 dan mAP@50-95), *Precision*, dan *Recall*. Metrik *Precision* dan *Recall* digunakan untuk mengukur tingkat ketepatan tebakan kelas dan kemampuan model mendeteksi keseluruhan produk. Sementara itu, mAP digunakan sebagai metrik utama karena secara komprehensif mengukur akurasi deteksi objek dengan mempertimbangkan tingkat tumpang tindih *bounding box*.

Variabel dari penelitian ini adalah serangkaian skenario rasio pembagian data berlabel terhadap data tidak berlabel yang disusun secara sistematis. Skenario ini terdiri dari sepuluh variasi, yaitu sembilan variasi metode SSL dengan kenaikan data berlabel sebesar 10% (mulai dari rasio 10:90 hingga 90:10) dan satu skenario *baseline* dengan rasio 100:0 yang dilatih secara *fully supervised* tanpa melibatkan data tidak berlabel. Skenario *baseline fully supervised* digunakan untuk mengukur ambang batas atas kemampuan model dengan cara membandingkan hasil rasio optimal pelabelan dengan hasil *baseline fully supervised*-nya. Model akan dilatih dan dievaluasi di setiap skenario rasio ini, kemudian performa skor mAP dari setiap skenario akan dianalisis dan dibandingkan besar delta (selisih) nilainya untuk menentukan titik rasio yang paling optimal antara data berlabel dan data tidak berlabel.

3.2 Pengumpulan Data

Data yang akan digunakan terdiri dari kumpulan citra digital yang menampilkan berbagai skenario rak-rak produk susu bubuk kardus dalam ruang lingkup ritel. Data ini bersumber dari basis data internal sebuah perusahaan ritel multinasional berskala besar dan berstatus rahasia (*confidential*) atau terikat oleh protokol kerahasiaan data industri. Data ini diambil menggunakan perangkat *mobile* pada beberapa toko ritel di Indonesia.

Fokus pengambilan data dilakukan pada area rak pajangan (*display*) untuk menangkap kondisi riil di lapangan, termasuk variasi pencahayaan, sudut pandang (*angle*), dan kepadatan produk. Hal ini bertujuan untuk mereplikasi secara presisi tantangan-tantangan fundamental yang sering dihadapi oleh model *computer vision* pada skenario industri riil. Gambar 3.2 merupakan contoh gambar dari dataset yang akan digunakan untuk deteksi objek pada penelitian ini. Dataset ini terdiri dari 408 citra digital dengan resolusi tinggi yang menampilkan berbagai skenario rak ritel produk susu bubuk kardus. Objek utama dalam citra tersebut mencakup berbagai varian produk susu bubuk kardus dengan berbagai ukuran berat (gramasi) serta kategori usia. Kumpulan objek ini menghadirkan tantangan klasifikasi *fine-grained* (kategorisasi visual tingkat halus), mengingat pembeda antar kelas sering kali hanya bergantung pada detail tekstual dan visual minor, seperti variasi ukuran berat (gramasi), label kategori peruntukan usia, maupun varian rasa. Kompleksitas struktur kemasan dan variabilitas lingkungan dalam dataset ini menjadikannya objek uji yang ideal untuk mengevaluasi efektivitas serta kemampuan pendekatan *Semi-Supervised Learning* dalam mempelajari representasi fitur dengan kuantitas label yang dibatasi.



Gambar 3.2 Contoh Data Produk Ritel Susu Bubuk Kardus

Penelitian ini menggunakan dua varian dataset yang berasal dari kumpulan citra yang sama, namun berbeda dalam standar kualitas anotasinya. Kedua varian dataset tersebut, yaitu Dataset Parsial dan Dataset Menyeluruh, akan dijelaskan secara mendalam di subbab berikutnya.

3.2.1 Dataset Parsial

Dataset Parsial merupakan dataset yang berisi anotasi bawaan yang telah tersedia sebelumnya. Anotasi pada dataset ini telah dibuat sebelum penelitian ini dilaksanakan sehingga tidak dilakukan proses pelabelan ulang. Anotasi bawaan tersebut langsung digunakan sebagaimana adanya sebagai *ground truth* untuk kebutuhan pelatihan dan evaluasi model dalam penelitian ini. Karakteristik anotasi ini cenderung lebih longgar dan tidak mencakup keseluruhan objek susu bubuk kardus yang ada di dalam citra. Tabel 3.1 menyajikan daftar kelas produk yang ada dalam Dataset Parsial yang berjumlah 47 kelas.

Tabel 3.1 Daftar Kelas Dataset Parsial

ID Kelas	Nama Kelas
0	00_dcw_1plus_madu_1000g
1	01_dcw_1plus_madu_800g
2	02_dcw_1plus_madu_400g
3	03_dcw_1plus_madu_200g
4	04_dcw_1plus_van_1000g`
5	05_dcw_1plus_van_800g
6	06_dcw_1plus_van_400g
7	07_dcw_1plus_van_200g
8	08_dcw_3plus_madu_1000g
9	09_dcw_3plus_madu_800g
10	10_dcw_3plus_madu_400g
11	11_dcw_3plus_van_1000g
12	12_dcw_3plus_van_800g
13	13_dcw_3plus_van_400g
14	14_dcw_5plus_madu_1000g
15	15_dcw_5plus_madu_800g
16	16_dcw_5plus_madu_400g
17	17_lact_1_happy_1kg
18	18_lact_1_happy_750g
19	19_lact_1_happy_350g
20	20_lact_1_happy_180g
21	21_lact_2_happy_1kg
22	22_lact_2_happy_750g
23	23_lact_2_happy_350g
24	24_lact_2_happy_180g
25	25_lact_pro_0_6_1kg
26	26_lact_pro_0_6_735g
27	27_lact_pro_0_6_350g
28	28_lact_pro_0_6_180g
29	29_lact_pro_6_12_1kg
30	30_lact_pro_6_12_735g
31	31_lact_pro_6_12_350g

32	32_lact_pro_6_12_180g
33	33_lactg_3_happy_madu_1kg
34	34_lactg_3_happy_madu_750g
35	35_lactg_3_happy_madu_350g
36	36_lactg_3_happy_madu_180g
37	37_lactg_3_happy_van_1kg
38	38_lactg_3_happy_van_750g
39	39_lactg_3_happy_van_350g
40	40_lactg_pro_1plus_madu_1kg
41	41_lactg_pro_1plus_madu_735g
42	42_lactg_pro_1plus_madu_350g
43	43_lactg_pro_1plus_madu_145g
44	44_lactg_pro_1plus_van_1kg
45	45_lactg_pro_1plus_van_735g
46	46_lactg_pro_1plus_van_350g

3.2.2 Dataset Menyeluruh

Dataset Menyeluruh menggunakan kumpulan citra yang sama dengan Dataset Parsial, namun seluruh proses anotasinya dilakukan secara independen dan manual menggunakan platform Roboflow. Pelabelan ulang ini dilakukan tanpa merujuk pada anotasi bawaan Dataset Parsial untuk menghasilkan *ground truth* yang bersifat mandiri dengan *bounding box* yang lebih ketat.

Selain melakukan anotasi ulang terhadap 47 kelas produk yang telah ada, peneliti juga menambahkan dua kelas baru pada Dataset Parsial, yaitu kelas “47_other_variant” dan kelas “48_other_brand”. Penambahan ini dilakukan karena pada Dataset Parsial terdapat objek-objek dalam citra yang tidak teranotasi (seperti varian produk lain dan merek produk lain yang terlihat di rak) yang berpotensi mengganggu proses pembelajaran model. Tabel 3.2 menyajikan daftar kelas produk yang ada dalam Dataset Menyeluruh yang berjumlah 49 kelas.

Tabel 3.2 Daftar Kelas Dataset Menyeluruh

ID Kelas	Nama Kelas
0	00_dcw_1plus_madu_1000g
1	01_dcw_1plus_madu_800g
2	02_dcw_1plus_madu_400g
3	03_dcw_1plus_madu_200g
4	04_dcw_1plus_van_1000g`
5	05_dcw_1plus_van_800g
6	06_dcw_1plus_van_400g
7	07_dcw_1plus_van_200g
8	08_dcw_3plus_madu_1000g
9	09_dcw_3plus_madu_800g
10	10_dcw_3plus_madu_400g
11	11_dcw_3plus_van_1000g
12	12_dcw_3plus_van_800g
13	13_dcw_3plus_van_400g
14	14_dcw_5plus_madu_1000g
15	15_dcw_5plus_madu_800g
16	16_dcw_5plus_madu_400g
17	17_lact_1_happy_1kg
18	18_lact_1_happy_750g
19	19_lact_1_happy_350g
20	20_lact_1_happy_180g
21	21_lact_2_happy_1kg
22	22_lact_2_happy_750g
23	23_lact_2_happy_350g
24	24_lact_2_happy_180g
25	25_lact_pro_0_6_1kg
26	26_lact_pro_0_6_735g
27	27_lact_pro_0_6_350g
28	28_lact_pro_0_6_180g
29	29_lact_pro_6_12_1kg
30	30_lact_pro_6_12_735g
31	31_lact_pro_6_12_350g
32	32_lact_pro_6_12_180g
33	33_lactg_3_happy_madu_1kg
34	34_lactg_3_happy_madu_750g
35	35_lactg_3_happy_madu_350g
36	36_lactg_3_happy_madu_180g
37	37_lactg_3_happy_van_1kg
38	38_lactg_3_happy_van_750g
39	39_lactg_3_happy_van_350g
40	40_lactg_pro_1plus_madu_1kg
41	41_lactg_pro_1plus_madu_735g
42	42_lactg_pro_1plus_madu_350g
43	43_lactg_pro_1plus_madu_145g
44	44_lactg_pro_1plus_van_1kg

45	45_lactg_pro_1plus_van_735g
46	46_lactg_pro_1plus_van_350g
47	47_other_variant
48	48_other_brand

3.3 Pra-pemrosesan Data

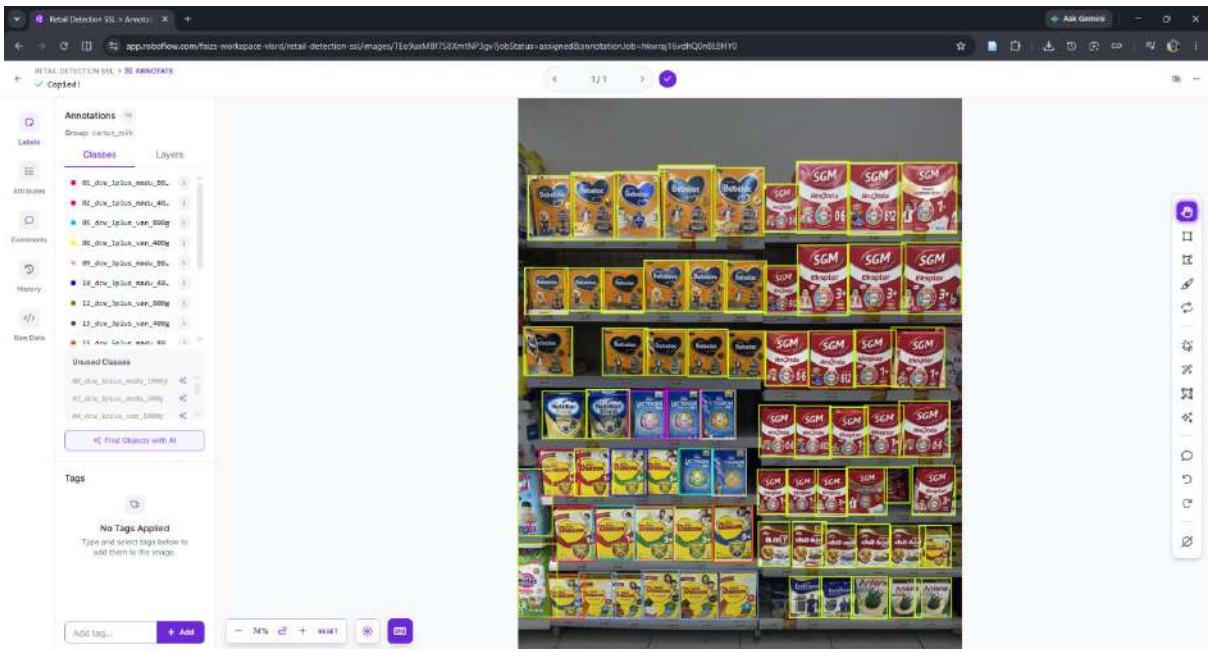
Setelah data dikumpulkan, tahapan selanjutnya adalah pra-pemrosesan data untuk memastikan dataset siap digunakan secara efektif oleh model. Dikarenakan penelitian ini melibatkan dua varian dataset, tahapan pra-pemrosesan data dilakukan secara berbeda untuk masing-masing dataset. Pada Dataset Parsial, pra-pemrosesan data hanya mencakup satu langkah, yaitu *data splitting*, dikarenakan anotasi sudah tersedia. Sementara itu, pada Dataset Menyeluruh, tahapan pra-pemrosesan data mencakup dua langkah, yaitu *data annotation* dan *data splitting*.

3.3.1 *Data Annotation*

Tahapan ini hanya diterapkan pada Dataset Menyeluruh. Proses anotasi secara manual dilakukan untuk membentuk himpunan data berlabel (*ground truth*). Proses ini dilakukan menggunakan platform Roboflow, sebuah alat bantu manajemen dataset *computer vision* yang memungkinkan untuk melakukan pelabelan objek secara presisi. Anotasi dilakukan dengan *Bounding Box Tool*, di mana setiap produk dalam citra ditandai dengan kotak pembatas yang rapat sesuai dengan dimensi fisik produk tersebut di rak ritel.

Proses anotasi diawali dengan mengunggah seluruh dataset citra yang akan digunakan ke platform Roboflow. Selanjutnya, pelabelan dilakukan satu per satu pada setiap citra menggunakan *Bounding Box Tool* untuk membuat kotak pembatas dari masing-masing objek. Untuk menjaga tingkat presisi, fitur perbesaran (*zoom*) dimanfaatkan saat memetakan area objek agar kotak pembatas yang ditarik ketat dan presisi mengelilingi batas fisik produk. Tahapan ini dilakukan secara konsisten hingga keseluruhan 408 citra selesai teranotasi. Setelah seluruh 408 citra teranotasi dengan lengkap, dataset diekspor menggunakan format standar YOLO.

Gambar 3.3 merupakan proses pelabelan pada gambar dengan platform Roboflow. Setiap produk dilabeli satu per satu secara manual sesuai dengan daftar 49 kelas yang telah ditentukan sebelumnya. Hasil dari tahap ini adalah *file* anotasi dalam format teks (.txt) standar YOLO yang memuat koordinat kotak pembatas (kelas, titik tengah X, titik tengah Y, lebar, tinggi) yang nilainya telah dinormalisasi pada rentang 0 hingga 1.



Gambar 3.3 Proses Pelabelan Menggunakan Roboflow

3.3.2 *Data Splitting*

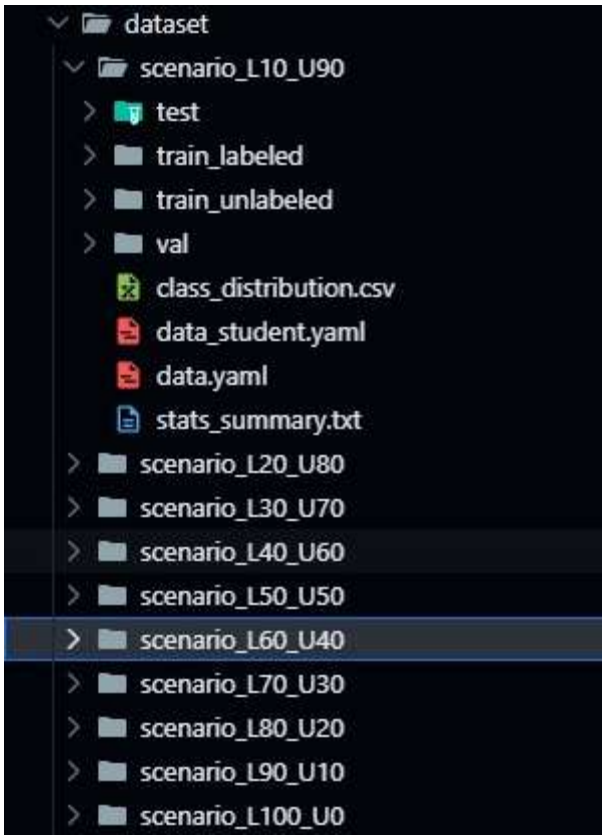
Tahapan pembagian data (*data splitting*) berlaku untuk kedua varian dataset (Dataset Parsial dan Dataset Menyeluruh) dan dilakukan dengan prosedur yang identik. Dikarenakan jumlah citra yang relatif kecil (408 citra) dan jumlah kelas yang banyak (47-49 kelas), proses pembagian ini dilakukan menggunakan metode *Greedy Stratified Sampling*. Metode ini bekerja dengan cara menempatkan citra ke dalam himpunan data secara iteratif (serakah) berdasarkan kelangkaan label kelas yang dimiliki, untuk memastikan distribusi kelas yang seimbang di seluruh himpunan dataset. Penggunaan metode ini untuk menjamin bahwa setiap kelas produk susu bubuk kardus tetap memiliki representasi pada data latih, data validasi, dan data uji, sehingga performa generalisasi model YOLOv12 dapat diukur secara adil pada seluruh varian produk.

Tahapan pembagian data ini dibagi menjadi dua tingkat. Tingkat Pertama adalah pembagian dataset secara umum untuk evaluasi model. Keseluruhan dataset sebanyak 408 citra dibagi menjadi tiga himpunan terpisah dengan rasio 80% untuk data latih (*train set*), 10% untuk data validasi (*validation set*), dan 10% untuk data uji (*test set*) sesuai dengan standar distribusi *machine learning* (Buleu dkk., 2025). Data latih adalah data yang akan dilihat oleh model selama proses *training* untuk mempelajari pola dan fitur objek. Data validasi digunakan untuk

memantau performa selama pelatihan dan penentuan parameter terbaik. Dan data uji berfungsi sebagai data “baru” yang hanya digunakan satu kali di akhir proses untuk mengukur performa final dan kemampuan generalisasi model secara objektif.

Tingkat kedua dari tahapan pembagian data ini adalah inti dari desain eksperimen *Semi-Supervised Learning* (SSL) dan hanya diterapkan pada data latih yang telah dipisahkan sebelumnya. Untuk menyimulasikan skenario kelangkaan data berlabel, 80% data latih tersebut dibagi lebih lanjut ke dalam sepuluh skenario rasio. Setiap skenario membagi data menjadi sub-himpunan data berlabel (*labeled set*) dan data tidak berlabel (*unlabeled set*), mulai dari rasio L10 (10% berlabel dan 90% tidak berlabel) hingga L100 (100% berlabel dan 0% tidak berlabel). Anotasi *ground truth* (label dan *bounding box*) pada himpunan *unlabeled* akan disembunyikan dari model selama proses pelatihan.

Struktur hasil pembagian data ditunjukkan pada Gambar 3.4. Pada struktur tersebut, folder dataset terdiri atas sepuluh folder skenario yang merepresentasikan setiap rasio pelabelan. Masing-masing folder skenario memiliki folder *val* dan *test* yang identik, sedangkan folder *train* dibagi menjadi dua subfolder, yaitu *train_labeled* dan *train_unlabeled*, sesuai dengan rasio pelabelan yang diterapkan pada skenario tersebut.



Gambar 3.4 Struktur Folder Dataset

3.4 Implementasi Model

Alur kerja implementasi model dalam penelitian ini dirancang untuk mengikuti skema *Semi-Supervised Learning* (SSL) dengan metode *Noisy Student* menggunakan model YOLOv12. Proses ini dirancang untuk memanfaatkan data tidak berlabel untuk meningkatkan performa deteksi objek pada produk ritel. Seluruh tahapan, mulai dari pelatihan model *Teacher* hingga model *Student*, dilakukan secara iteratif untuk masing-masing varian dataset (Dataset Parsial dan Dataset Menyeluruh) sebanyak sepuluh kali sesuai dengan skenario rasio yang diuji, yaitu L10 hingga L100. Dengan demikian, metodologi implementasi model yang dijelaskan pada subbab berikut berlaku identik untuk kedua varian dataset. Perbedaan antara kedua dataset terletak hanya pada standar kualitas anotasi yang digunakan, sebagaimana telah dijelaskan pada bagian Pengumpulan Data.

3.4.1 *Teacher Training*

Tahap awal implementasi dilakukan dengan melatih model *Teacher* menggunakan arsitektur YOLOv12-Nano (yolov12n.pt). Pemilihan varian Nano ini didasarkan pada pertimbangan teknis terkait ukuran dataset penelitian yang relatif kecil yang hanya berjumlah

408 citra. Penggunaan model dengan parameter yang lebih besar pada skala data yang terbatas berisiko tinggi menyebabkan *overfitting*, di mana model cenderung menghafal data latihan daripada mempelajari fitur secara *general*. Pelatihan difokuskan sepenuhnya pada dataset berlabel (*labeled set*) yang telah ditentukan dalam skenario rasio tertentu untuk membentuk basis pengetahuan awal model. Selama dan setelah proses pelatihan ini selesai, seluruh nilai metrik evaluasi seperti *mean Average Precision* (mAP), *Precision*, dan *Recall* dicatat secara sistematis. Pencatatan ini mencakup evaluasi performa pada data latih (*train set*), data validasi (*validation set*), dan data uji (*test set*) untuk menganalisis karakteristik konvergensi model awal pada tiap skenario.

Konfigurasi pelatihan diatur secara terperinci melalui serangkaian *hyperparameter* untuk mencapai konvergensi yang optimal. Gambar 3.5 merupakan kode konfigurasi *hyperparameter* untuk melatih model *Teacher*. Baris 5-7 mengatur resolusi gambar 640 piksel selama 200 *epoch* dengan *batch size* sebesar 16. Pemilihan resolusi 640 piksel bertujuan untuk menjaga keseimbangan antara ketelitian deteksi objek dan efisiensi komputasi. Jumlah *epoch* sebanyak 200 digunakan agar model memiliki waktu yang cukup untuk mempelajari pola data secara menyeluruh tanpa menyebabkan *overfitting* yang berlebihan. Sementara itu, *batch size* 16 dipilih untuk menyesuaikan keterbatasan memori komputasi sekaligus menjaga stabilitas proses pelatihan. Digunakan juga *hyperparameter* regularisasi pada baris 14 berupa *dropout* sebesar 0,1 untuk memitigasi risiko *overfitting* pada data yang terbatas. Dalam proses ini, pelatihan model *Teacher* hanya menerapkan konfigurasi augmentasi standar yang tersedia pada *framework* YOLOv12 untuk menjaga integritas fitur asli dari data berlabel.

```
1  model = YOLO("yolov12n.pt")
2
3  results = model.train(
4      data=DATA_YAML,
5      epochs=200,
6      imgsz=640,
7      batch=16,
8      optimizer='auto',
9      lr0=0.01,
10     lrf=0.01,
11     momentum=0.937,
12     weight_decay=0.0005,
13     warmup_epochs=5.0,
14     dropout=0.1,
15     mosaic=1.0,
16     close_mosaic=10,
17     mixup=0.0,
18     hsv_h=0.015,
19     hsv_s=0.7,
20     hsv_v=0.4,
21     project=PROJECT_NAME,
22     name=RUN_NAME,
23     device=0,
24     patience=30,
25 )
```

Gambar 3.5 Kode Konfigurasi *Hyperparameter Teacher*

3.4.2 *Pseudo-labeling*

Tahap selanjutnya adalah proses *pseudo-labeling*, di mana model *Teacher* yang telah menghasilkan bobot terbaik (best.pt) digunakan untuk memberikan label semu (*pseudo-label*) pada dataset tidak berlabel (*unlabeled set*). Proses ini bertujuan untuk mengubah data mentah menjadi data yang memiliki anotasi tanpa campur tangan manusia secara langsung, sehingga memperbesar volume data latih untuk tahap berikutnya. Implementasi ini dilakukan dengan memuat model *Teacher* untuk menginferensi seluruh gambar yang terdapat dalam direktori "../train_unlabeled".

Gambar 3.6 merupakan kode untuk melakukan proses *pseudo-labeling*. Untuk menjamin kualitas label semu yang dihasilkan, pada baris 5 ditetapkan ambang batas keyakinan (*confidence threshold*) yang cukup tinggi sebesar 0,70. Hal ini dilakukan agar hanya prediksi dengan tingkat kepastian tinggi yang akan dijadikan label sehingga meminimalisir masuknya data *noise* yang salah ke dalam proses pelatihan model *Student*. Selain itu, pada baris 6 digunakan juga IoU *threshold* sebesar 0,60 untuk mekanisme *Non-Maximum Suppression* (NMS) agar tidak terjadi tumpang tindih *bounding box* pada objek yang sama.

Setiap hasil prediksi kemudian dikonversi ke dalam format YOLO standar (*class*, *x_center*, *y_center*, *width*, *height*) dengan nilai koordinat yang telah dinormalisasi antara 0

hingga 1 pada baris 13-25. Langkah ini memastikan seluruh koordinat berada di dalam bingkai gambar meskipun terdapat proses augmentasi. Label-label tersebut disimpan dalam *file* teks (.txt) pada direktori "../train_unlabeled/labels" yang sejajar dengan folder citranya. Untuk menjaga konsistensi dataset, gambar yang tidak menghasilkan deteksi tetap dibuatkan *file* label kosong, yang secara teknis menandakan bahwa gambar tersebut tidak mengandung objek atau dianggap sebagai latar belakang (*background*).

```
1  model = YOLO(MODEL_PATH)
2
3  results = model.predict(
4      source=image_files,
5      conf=0.70,
6      iou=0.60,
7      augment=True,
8      save=False,
9      stream=True,
10     verbose=False
11 )
12
13 for result in results:
14     labels_str = ""
15     if len(result.boxes) > 0:
16         for box in result.boxes:
17             cls_id = int(box.cls[0].item())
18             x, y, w, h = box.xywhn[0].tolist()
19             x = max(0.0, min(1.0, x))
20             y = max(0.0, min(1.0, y))
21             w = max(0.0, min(1.0, w))
22             h = max(0.0, min(1.0, h))
23             labels_str += f"{cls_id} {x:.6f} {y:.6f} {w:.6f} {h:.6f}\n"
24     with open(txt_path, "w") as f:
25         f.write(labels_str)
```

Gambar 3.6 Kode Implementasi *Pseudo-labeling*

Evaluasi *Pseudo-Label*

Untuk mengukur seberapa layak *pseudo-label* ini digunakan sebagai bahan ajar bagi model *Student*, dilakukan evaluasi kualitas secara mendalam. Evaluasi ini dilakukan dengan membandingkan *pseudo-label* yang dihasilkan oleh model *Teacher* terhadap label asli (*ground truth*) dari data tidak berlabel yang sengaja disembunyikan (*hidden ground truth*). Pengukuran menggunakan batas *Intersection over Union* (IoU) sebesar 0,5 untuk mengekstrak tiga parameter utama, yaitu:

a. *True Positive* (TP): Merupakan tebakan benar. Model *Teacher* berhasil mendeteksi objek dengan kelas yang tepat dan kotak (*bounding box*) yang akurat. Semakin tinggi TP, semakin tinggi juga informasi yang akan dipelajari oleh model *Student*.

b. *False Positive* (FP): Merupakan tebakan salah (halusinasi). Model *Teacher* menebak area latar belakang (seperti rak kosong atau dinding) sebagai objek produk, atau model salah mengenali kelas produk. FP yang tinggi sangat berbahaya karena akan memberikan ilmu yang salah kepada model *Student*.
c. *False Negative* (FN): Merupakan objek yang terlewatkan. Model *Teacher* gagal melihat produk yang sebenarnya ada di rak dan menganggapnya sebagai latar belakang. FN yang tinggi menyebabkan model *Student* menjadi buta terhadap fitur-fitur objek tersebut.

Algoritma pencocokan *bounding box* antara *pseudo-label* dan *hidden ground truth* dibangun menggunakan perulangan yang mengevaluasi kecocokan kelas dan nilai ambang *Intersection over Union* (IoU). Gambar 3.7 menunjukkan logika inti dari penentuan *True Positive* (TP), *False Positive* (FP), dan *False Negative* (FN). Baris 1-4 melakukan iterasi setiap *bounding box* dari hasil prediksi model pada tahap *pseudo-labeling*. Baris 6-15 membandingkan *bounding box* hasil prediksi dengan *hidden ground truth* yang ada. Baris 17-21 menghitung total TP dan FP berdasarkan nilai IoU sebesar 0,5. Sedangkan, baris 23 menghitung FN dengan menghitung sisa *ground truth* yang tidak tertandai.

```
1  for p_box in pred_boxes:
2      best_iou = 0
3      best_gt_idx = -1
4      p_cls = p_box[0]
5
6      for idx, g_box in enumerate(gt_boxes):
7          if gt_matched[idx]:
8              continue
9
10         g_cls = g_box[0]
11         if p_cls == g_cls:
12             iou = calculate_iou(p_box[1:], g_box[1:])
13             if iou > best_iou:
14                 best_iou = iou
15                 best_gt_idx = idx
16
17     if best_iou >= IOU_THRESHOLD:
18         total_tp += 1
19         gt_matched[best_gt_idx] = True
20     else:
21         total_fp += 1
22
23 total_fn += gt_matched.count(False)
```

Gambar 3.7 Kode Penentuan TP, FP, dan FP pada *Pseudo-Label*

Selain ketiga metrik kuantitatif di atas, dijalankan modul validasi YOLOv12 pada direktori evaluasi sementara untuk mendapatkan skor metrik *mean Average Precision* (mAP)

untuk *pseudo-label* yang sudah dibuat. Kombinasi metrik mAP dan sebaran TP-FP-FN ini memberikan gambaran besar mengenai seberapa layak *pseudo-label* digunakan pada tahap selanjutnya. Gambar 3.8 merupakan kode untuk mengevaluasi nilai mAP dari *pseudo-label*. Baris 2-3 membuat dataset sementara yang berisikan data tidak berlabel (*unlabeled set*) dan juga *hidden ground truth* yang sengaja disembunyikan. Baris 5 memuat model *Teacher* yang akan digunakan untuk melihat bagaimana performa inferensinya. Baris 7-12 berfungsi untuk menjalankan validasi dengan parameter *confidence treshold* dan juga IoU yang sama pada saat *pseudo-labeling*, yaitu 0,70 dan 0,60.

```
1   def evaluate_pseudo_map():
2       img_dest, lbl_dest = prepare_evaluation_data()
3       yaml_path = create_yaml(img_dest)
4
5       model = YOLO(MODEL_PATH)
6
7       metrics = model.val(
8           data=yaml_path,
9           split="val",
10          conf=0.70,
11          iou=0.60
12      )
13
14      return metrics
```

Gambar 3.8 Kode Evaluasi Nilai mAP pada *Pseudo-Label*

3.4.3 *Konfigurasi Dataset Gabungan*

Setelah proses *pseudo-labeling* selesai, tahap berikutnya adalah penyusunan konfigurasi dataset yang menggabungkan data berlabel asli (*labeled*) dengan data berlabel semu (*pseudo-labeled*). Gambar 3.9 merupakan kode untuk membuat dataset gabungan data berlabel dan data *pseudo-labeled*. Proses ini diawali pada baris 4-5 dengan validasi terhadap keberadaan *file* label di direktori "../train_unlabeled/labels" untuk memastikan data tersebut tidak dianggap sebagai gambar latar belakang (*background*) oleh model yang dapat menurunkan akurasi deteksi secara signifikan. Baris 7 mengatur parameter *train* sebagai daftar (*list*) yang mencakup jalur direktori "../train_labeled/images" dan "../train_unlabeled/images" untuk menggabungkan data berlabel dengan data *pseudo-label*. Baris 8 dan 9 mengunci direktori *val* dan juga *test*. Parameter ini memastikan bahwa model *Student* nantinya hanya akan divalidasi dan diuji menggunakan *ground truth* asli, sehingga menghindarkan sistem dari kontaminasi data (*data leakage*). Pada baris 11-13 konfigurasi dilakukan dengan membuat *file* YAML konfigurasi baru bernama "data_student.yaml" tanpa mengubah urutan parameter awal.

Pendekatan ini memungkinkan model *Student* untuk memperluas basis pengetahuannya melalui volume data yang lebih besar tanpa mengorbankan integritas set data validasi (*validation set*) dan pengujian (*test set*) yang tetap merujuk pada data murni berlabel manusia.

```
1  with open(source_yaml, 'r') as f:
2      data = yaml.safe_load(f)
3
4  check_labels_exist(path_labeled)
5  check_labels_exist(path_unlabeled)
6
7  data['train'] = [path_labeled, path_unlabeled]
8  data['val']   = os.path.join(scenario_dir, "val",  "images")
9  data['test']  = os.path.join(scenario_dir, "test", "images")
10
11 output_yaml = os.path.join(scenario_dir, "data_student.yaml")
12 with open(output_yaml, 'w') as f:
13     yaml.dump(data, f, sort_keys=False)
```

Gambar 3.9 Kode Pembuatan Dataset Gabungan

3.4.4 *Student Training*

Tahapan terakhir dari implementasi *Semi-Supervised Learning* dengan *Noisy Student* ini adalah pelatihan model *Student* menggunakan konfigurasi dataset gabungan yang telah disusun sebelumnya. Pada tahap ini, model *Student* tidak dilatih dengan melanjutkan bobot (*weights*) terbaik dari model *Teacher*, melainkan diinisialisasi ulang dari awal (*cold start*) menggunakan model YOLOv12-Nano (yolov12n.pt). Pendekatan dilakukan untuk mencegah model *Student* mewarisi bias atau kesalahan spesifik yang mungkin telah dipelajari oleh model *Teacher*.

Gambar 3.10 merupakan kode konfigurasi *hyperparameter* untuk melatih model *Student*. Karakteristik utama pada tahap ini, yang menjadi dasar dari metode *Noisy Student*, adalah penerapan *noise* dalam tingkat tinggi melalui augmentasi data yang lebih agresif dibandingkan pelatihan model *Teacher*. Pada baris 16-19, injeksi *noise* dilakukan baik secara geometris maupun visual, mencakup penggunaan *mixup* sebesar 0,1 (menggabungkan dua gambar), variasi skala (*scale*) sebesar 0,5, translasi (*translate*) sebesar 0,1, serta pembalikan horizontal (*fliplr*) sebesar 0,5. Augmentasi ini secara sengaja dirancang untuk membuat proses pembelajaran model *Student* menjadi lebih menantang. Dengan kondisi data yang lebih bervariasi dan "berisik", model didorong untuk mempelajari representasi fitur objek ritel yang lebih umum dan tahan terhadap variasi (*robust*) sehingga mampu mengoreksi ketidaksempurnaan pada *pseudo-label* dan melampaui kemampuan model *Teacher*. Baris 23 mengaktifkan *Cosine Annealing Learning Rate*. Dikarenakan data gabungan kini mengandung *pseudo-label* yang mungkin memiliki sedikit kesalahan, parameter ini mengubah penurunan

learning rate dari bentuk linier menjadi kurva kosinus. Hal ini membantu model *Student* untuk mengurangi laju pembelajarannya secara perlahan dan halus di tahap akhir (*epoch* atas) agar tidak merusak konvergensi bobot yang sudah stabil.

```
model = YOLO("yolov12n.pt")

model.train(
    data=DATA_YAML,
    epochs=200,
    imgsz=640,
    batch=16,
    optimizer='auto',
    lr0=0.01,
    lrf=0.01,
    momentum=0.937,
    weight_decay=0.0005,
    warmup_epochs=5.0,
    mosaic=1.0,
    close_mosaic=10,
    mixup=0.1,
    translate=0.1,
    scale=0.5,
    fliplr=0.5,
    hsv_h=0.015,
    hsv_s=0.7,
    hsv_v=0.4,
    cos_lr=True,
    project=PROJECT_NAME,
    name=RUN_NAME,
    device=0,
    patience=30,
)
```

Gambar 3.10 Kode Konfigurasi *Hyperparameter Student*

Sama seperti pada tahap model *Teacher*, seluruh nilai metrik evaluasi (mAP@50, mAP@50-95, *Precision*, *Recall*) pada fase pelatihan model *Student* ini dicatat secara komprehensif. Nilai evaluasi akhir dari model *Student* inilah yang akan dikompilasi dan menjadi metrik utama untuk dianalisis secara komparatif antar-skenario guna menemukan rasio pelabelan yang paling optimal (*sweet spot*) dalam menghasilkan performa deteksi tertinggi pada ruang lingkup ritel.

3.5 Analisis Hasil

Pada tahapan analisis hasil, dilakukan evaluasi mendalam dan pembahasan komprehensif terhadap seluruh log metrik yang terekam selama eksperimen penerapan metode *Semi-Supervised Learning* (SSL) dengan algoritma *Noisy Student* menggunakan arsitektur

YOLOv12. Analisis ini dirancang secara sistematis untuk mengungkap efektivitas konvergensi model, pengaruh kualitas anotasi, serta menentukan titik efisiensi optimal penggunaan data berlabel.

3.5.1 Analisis Perbandingan mAP, *Precision*, dan *Recall* antar-skenario

Tahap ini membahas evaluasi performa akhir model *Student* pada data uji (*test set*) yang tidak digunakan selama proses pelatihan. Evaluasi ini dilakukan untuk mengetahui kemampuan model dalam melakukan generalisasi terhadap data baru yang memiliki kondisi serupa dengan lingkungan ritel sebenarnya. Pengujian dilakukan dengan menggunakan beberapa metrik evaluasi, yaitu *mean Average Precision* (mAP@50 dan mAP@50-95), *Precision*, dan *Recall*. Seluruh metrik tersebut dibandingkan pada setiap skenario rasio data berlabel, mulai dari L10 hingga L100.

Perbandingan antar-skenario digunakan untuk melihat perubahan performa model seiring dengan bertambahnya jumlah data berlabel. Hasil tersebut kemudian digunakan untuk mengetahui seberapa besar pengaruh penambahan data berlabel terhadap peningkatan akurasi model. Dengan demikian, dapat diketahui skenario yang memberikan keseimbangan terbaik antara jumlah data yang perlu dilabeli secara manual dan performa deteksi yang dihasilkan.

Selain melihat perubahan nilai mAP, evaluasi juga dilakukan terhadap *Precision* dan *Recall* untuk mengetahui karakteristik hasil prediksi model. *Precision* menunjukkan kemampuan model dalam menghasilkan prediksi yang benar dan mengurangi *False Positive*, sedangkan *Recall* menunjukkan kemampuan model dalam menemukan objek yang terdapat pada data uji sehingga dapat mengurangi *False Negative*. Sementara itu, perbandingan antara mAP@50 dan mAP@50-95 digunakan untuk melihat kemampuan model dalam mempertahankan ketepatan lokasi *bounding box* ketika menggunakan kriteria IoU yang lebih ketat. Hasil evaluasi ini menjadi dasar dalam menentukan efektivitas pendekatan *Semi-Supervised Learning* pada berbagai rasio data berlabel.

3.5.2 Analisis Kurva Konvergensi mAP per *Epoch*

Tahap ini berfokus pada evaluasi proses pembelajaran model dengan mengamati kurva mAP terhadap jumlah *epoch*. Analisis dilakukan untuk melihat perkembangan performa model selama proses pelatihan, bukan hanya berdasarkan hasil akhirnya. Kurva konvergensi dibandingkan pada beberapa skenario terpilih yang mewakili kondisi dengan jumlah data berlabel rendah hingga *fully supervised*, yaitu L10, L30, L50, L70, dan L100.

Pengamatan terhadap kurva tersebut digunakan untuk mengetahui kestabilan proses pembelajaran pada setiap skenario. Model dengan jumlah data berlabel yang sangat terbatas dapat mengalami perubahan nilai mAP yang cukup besar selama pelatihan. Kondisi tersebut dapat menunjukkan bahwa model masih kesulitan mempelajari pola objek secara konsisten karena informasi dari data berlabel yang tersedia masih terbatas. Dalam pendekatan *Semi-Supervised Learning*, kondisi ini juga dapat dipengaruhi oleh kualitas *pseudo-label* yang digunakan selama proses pelatihan.

Selain melihat kestabilan kurva, analisis ini digunakan untuk mengetahui pada rasio data berlabel berapa model mulai menunjukkan proses konvergensi yang lebih stabil. Kurva yang semakin stabil menunjukkan bahwa performa model tidak lagi mengalami perubahan yang terlalu besar antar *epoch*. Hasil tersebut kemudian dibandingkan dengan skenario L100 sebagai *baseline* untuk melihat apakah model dengan jumlah data berlabel yang lebih sedikit mampu mencapai pola konvergensi yang mendekati model *fully supervised*.

Analisis kurva konvergensi ini menjadi salah satu dasar untuk menilai efisiensi penggunaan data berlabel. Dengan membandingkan kestabilan proses pelatihan pada setiap skenario, dapat diketahui apakah penambahan data berlabel memberikan perubahan yang berarti terhadap proses pembelajaran model atau mulai menunjukkan peningkatan yang semakin kecil.

3.5.3 Analisis *Training Box Loss* dan *Classification Loss*

Tahap ini mengevaluasi performa model selama masa pelatihan berdasarkan grafik penurunan fungsi kerugian (*loss function*). Evaluasi pada *Training Box Loss* untuk mengukur akurasi lokalisasi koordinat *bounding box*, dan *Training Classification Loss* untuk mengukur ketepatan klasifikasi kategori produk ritel. Analisis ini berfungsi sebagai instrumen diagnostik untuk mendeteksi adanya kendala pembelajaran, memvalidasi temuan dari kurva konvergensi, dan mengonfirmasi stabilitas fungsional model di setiap skenario rasio.

Tahap ini berfokus pada analisis proses pembelajaran model selama pelatihan melalui pengamatan terhadap grafik *loss*. Analisis dilakukan pada dua komponen utama, yaitu T*raining Box Loss* dan *Training Classification Loss*. *Training Box Loss* digunakan untuk melihat tingkat kesalahan model dalam menentukan posisi dan ukuran *bounding box* objek. Semakin kecil nilai *loss* menunjukkan bahwa prediksi lokasi objek semakin mendekati anotasi yang diberikan. Sementara itu, *Training Classification Loss* digunakan untuk melihat kemampuan model dalam menentukan kelas dari objek yang telah dideteksi.

Perbandingan kedua nilai *loss* pada setiap skenario rasio data berlabel digunakan untuk melihat bagaimana jumlah data berlabel memengaruhi proses pembelajaran model. Nilai *loss* yang menurun dan kemudian cenderung stabil menunjukkan bahwa model dapat mempelajari pola dari data secara konsisten. Sebaliknya, perubahan nilai yang terlalu besar atau tidak stabil dapat menunjukkan bahwa proses pembelajaran masih mengalami kesulitan.

Analisis *loss* juga digunakan untuk melengkapi hasil pengamatan pada kurva mAP. Jika kurva mAP menunjukkan peningkatan yang stabil dan nilai *loss* mengalami penurunan secara konsisten, maka proses pembelajaran dapat dikatakan berjalan dengan baik. Sebaliknya, apabila terjadi fluktuasi yang besar atau perbedaan pola antara *loss* dan mAP, kondisi tersebut perlu diperhatikan karena dapat berkaitan dengan keterbatasan data berlabel maupun kualitas *pseudo-label* yang digunakan dalam proses *Semi-Supervised Learning*.

3.5.4 Analisis Efisiensi dan Identifikasi *Sweet Spot*

Tahap ini mengidentifikasi titik optimal rasio data atau *sweet spot*. Analisis ini dilakukan secara matematis dengan menghitung selisih (delta) peningkatan mAP@50 antar-skenario yang berurutan. Evaluasi berfokus pada penemuan fase pertumbuhan performa yang signifikan (*tipping point*) dan fase kejenuhan metrik (*diminishing returns*). Hasil dari analisis komparatif ini akan menyimpulkan rasio paling *cost-effective* bagi industri ritel, di mana target akurasi tinggi dapat dipertahankan tanpa harus melakukan anotasi data secara penuh.

Selanjutnya, hasil dari analisis *sweet spot* tersebut akan dibandingkan secara langsung dengan *baseline fully supervised* (L100) yang hanya dilatih menggunakan 100% data berlabel. Perbandingan terhadap *baseline fully supervised* ini dilakukan untuk mengukur seberapa dekat performa skenario *sweet spot* (yang hanya memanfaatkan sebagian data berlabel) dalam mendekati batas performa maksimal (*upper bound*) dari model *fully supervised* murni.

3.5.5 Analisis Kualitatif Visual Hasil Deteksi

Tahap ini berfokus pada evaluasi kualitatif secara visual terhadap hasil prediksi model *Student* pada skenario optimal. Analisis ini bertujuan untuk membandingkan anotasi *ground truth* secara langsung dengan *bounding box* hasil prediksi model untuk memahami karakteristik keberhasilan deteksi (*True Positive*). Selain itu, evaluasi ini dirancang untuk melihat pola kegagalan model, seperti objek yang gagal terdeteksi (*False Negative*) maupun kesalahan klasifikasi antar varian produk (*False Positive*). Melalui pendekatan visual ini, dapat diidentifikasi apakah kelemahan model disebabkan oleh faktor eksternal (seperti pencahayaan

dan oklusi pada rak) atau oleh keterbatasan internal dari metode Noisy Student saat mempelajari data berlabel parsial.

3.6 Visualisasi Sederhana

Untuk memudahkan evaluasi dan observasi terhadap hasil penelitian, akan dirancang sebuah visualisasi sederhana berupa aplikasi web interaktif menggunakan *library* Streamlit pada bahasa pemrograman Python. Tujuan utama dari aplikasi demo ini adalah untuk memvisualisasikan secara dinamis dampak dari setiap skenario rasio data berlabel terhadap performa model YOLOv12 yang telah dianalisis pada subbab sebelumnya. Aplikasi demo ini dirancang dengan antarmuka yang ramah pengguna dan memuat komponen interaktif utama berupa panel penggeser (*slider*). *Slider* ini memungkinkan pengguna untuk memilih persentase rasio data berlabel secara *real-time*, mulai dari 10% hingga 100%, sementara opsi pemilihan dataset memungkinkan pengguna untuk memilih antara Dataset Parsial dan Dataset Menyeluruh.

Berdasarkan *input* dari *slider* tersebut, aplikasi akan menampilkan hasil analisis yang dibagi ke dalam tiga segmen tabulasi utama. Segmen pertama difokuskan pada demo inferensi gambar, yang menyediakan fitur unggah gambar (*file uploader*) agar pengguna dapat memasukkan gambar rak ritel baru. Pada tahap ini, sistem akan memuat bobot model (*weights*) yang sesuai dengan rasio *slider* terpilih dan menjalankan inferensi untuk menampilkan perbandingan antara gambar asli dengan gambar hasil deteksi objek beserta *bounding box*-nya.

Selanjutnya, segmen kedua memuat grafik metrik pelatihan yang akan membaca data hasil pelatihan (*log* CSV) dari rasio yang dipilih dan memvisualisasikannya dalam bentuk grafik garis. Metrik yang ditampilkan pada bagian ini mencakup performa deteksi seperti *mean Average Precision*, *Precision*, dan *Recall*, serta grafik penurunan fungsi kerugian yang meliputi *Box Loss*, *Cls Loss*, dan *DFL Loss*.

Terakhir, segmen ketiga dikhususkan untuk analisis komparatif semua skenario dengan menampilkan grafik gabungan dari seluruh rasio secara bersamaan untuk kedua varian dataset. Melalui perbandingan visual antar kurva mAP@50 dan mAP5@0-95 dari Dataset Parsial maupun Dataset Menyeluruh, serta tabel perhitungan persentase kenaikan (delta), fitur ini dirancang secara khusus untuk mempermudah penarikan kesimpulan secara objektif. Analisis komprehensif pada antarmuka ini dapat membantu dalam menemukan *sweet spot* atau titik optimal dari implementasi metode *Noisy Student* pada masing-masing varian dataset, sekaligus

memberikan wawasan mengenai pengaruh kualitas anotasi terhadap efisiensi *Semi-Supervised Learning* di domain ritel.

BAB IV
HASIL DAN PEMBAHASAN

4.1 Lingkungan Implementasi

Seluruh rangkaian eksperimen dalam penelitian ini dilaksanakan dalam sebuah lingkungan komputasi yang terkonfigurasi secara konsisten untuk memastikan reproduksibilitas hasil. Dari sisi perangkat keras, proses pelatihan dilakukan dengan GPU NVIDIA RTX 2060 SUPER dengan memori video 8 GB VRAM. Kinerja GPU tersebut didukung oleh prosesor Intel Core i5-10400F dan memori utama sebesar 16 GB DDR4 RAM yang bertugas menangani operasi pra-pemrosesan data, manajemen *pipeline* pemuatan data, serta koordinasi alur algoritma *Noisy Student* di bawah sistem operasi Windows 11 (64-bit).

Sementara itu, dari sisi perangkat lunak, penelitian ini menggunakan bahasa pemrograman Python versi 3.10 dengan framework utama Ultralytics YOLOv12 versi 8.3.63. Untuk mendukung proses pengujian model secara interaktif, penelitian ini juga mengembangkan sebuah demo aplikasi berbasis web menggunakan *framework* Streamlit versi 1.54.0. Rincian spesifikasi teknis dari keseluruhan perangkat keras dan perangkat lunak yang digunakan dalam lingkungan implementasi ini dirangkum secara komprehensif pada Tabel 4.1.

Tabel 4.1 Spesifikasi Lingkungan Impelementasi

Komponen	Spesifikasi
Sistem Operasi	Windows 11 (64-bit)
GPU	NVIDIA RTX 2060 SUPER 8GB VRAM
CPU	Intel Core i5-10400F
RAM	16 GB DDR4
Framework	Ultralytics YOLOv12 versi 8.3.63
Python	Python 3.10
Visualisasi Demo	Streamlit versi 1.54.0
Durasi *Training*	+- 25 menit/skenario, 200 *epoch*

4.2 Hasil Pra-pemrosesan Data

Subbab ini menyajikan hasil dari tahapan pengolahan dataset ritel yang digunakan dalam penelitian. Penelitian ini melibatkan dua varian dataset yang dibandingkan secara sistematis: (1) Dataset Parsial, yang merupakan dataset awal dengan anotasi bawaan yang cenderung longgar dan tidak mencakup seluruh objek; dan (2) Dataset Menyeluruh, yang merupakan dataset hasil anotasi ulang secara independen menggunakan platform Roboflow dengan

pendekatan yang lebih presisi, ketat, dan mencakup seluruh objek yang terlihat. Kedua dataset tersebut bersumber dari kumpulan citra yang sama, yaitu 408 citra produk susu bubuk kardus.

4.2.1 Hasil Pelabelan Citra

Proses pelabelan ulang secara mandiri menghasilkan perbedaan kepadatan anotasi yang sangat signifikan pada Dataset Menyeluruh. Pada Dataset Menyeluruh, pelabelan dilakukan dengan tingkat kepadatan yang lebih tinggi dengan menerapkan dua kelas tambahan, yaitu kelas "47_other_variant" dan "48_other_brand". Secara kuantitatif, Dataset Menyeluruh menghasilkan total sebanyak 5665 label, sementara Dataset Parsial hanya memiliki 2614 label. Hal ini merupakan peningkatan kepadatan anotasi sebesar 117% dari Dataset Parsial. Perbedaan ini menunjukkan bahwa Dataset Menyeluruh berhasil menangkap objek-objek yang sebelumnya tidak teranotasi pada Dataset Parsial, sehingga mengurangi risiko *false negative* pada saat pelatihan model YOLOv12.

4.2.2 Hasil *Data Splitting*

Proses pembagian data ini dilakukan pada kedua dataset. Pembagian skenario pada masing-masing dataset dilakukan menggunakan metode *Greedy Stratified Sampling* untuk memastikan representasi seluruh kelas pada setiap himpunan. Dataset dibagi dengan proporsi 80% untuk data latih (*train set*), 10% untuk data validasi (*validation set*), dan 10% untuk data uji (*test set*). Dengan total 408 citra, hasil pembagian tersebut menghasilkan sekitar 328 citra untuk data latih, 40 citra untuk data validasi, dan 40 citra untuk data uji. Dari 80% data latih inilah kemudian dilakukan pembagian lebih lanjut sesuai skenario rasio data berlabel dan data tidak berlabel yang diujikan dalam penelitian ini. Tabel 4.2 menyajikan perbandingan jumlah data berlabel dan data tidak berlabel. Terlihat skenario L10 hanya memiliki 32 data berlabel sedangkan total kelas yang ada di Dataset Parsial sebanyak 47 kelas dan di Dataset Menyeluruh sebanyak 49 kelas. Hal ini membuat tidak semua kelas dapat dilihat oleh model selama proses pelatihan.

Tabel 4.2 Perbandingan Seluruh Skenario Data

Skenario	% Berlabel	Jumlah Data Berlabel	Jumlah Data Tidak Berlabel	Total Data Latih	Keterangan
L10	10%	32	298	328	SSL
L20	20%	65	263	328	SSL
L30	30%	98	230	328	SSL
L40	40%	131	197	328	SSL
L50	50%	164	164	328	SSL
L60	60%	196	132	328	SSL
L70	70%	229	99	328	SSL
L80	80%	262	66	328	SSL
L90	90%	295	33	328	SSL
L100	100%	328	0	328	Supervised

4.3 Hasil Implementasi Model

Subbab ini menyajikan hasil implementasi model yang dilakukan pada penelitian menggunakan pendekatan *Semi-Supervised Learning* (SSL) dengan algoritma *Noisy Student*. Implementasi dilakukan dengan menguji berbagai skenario rasio data berlabel dan data tidak berlabel pada dua jenis dataset, yaitu Dataset Parsial dan Dataset Menyeluruh.

4.3.1 Hasil Pelatihan Model *Teacher*

Model *Teacher* merupakan model YOLOv12-Nano yang dilatih secara *supervised* hanya menggunakan data berlabel pada masing-masing skenario. Seluruh model *Teacher* dilatih dengan konfigurasi seragam yaitu resolusi 640 piksel, *batch size* 16, maksimum 200 *epoch* dengan *early stopping* (*patience* 30), dan *dropout* 0,10. Proses evaluasi dilakukan secara menyeluruh pada data latih (*train set*), data validasi (*validation set*), dan data uji (*test set*) untuk menganalisis karakteristik konvergensi model pada tiap skenario. Metrik evaluasi yang digunakan adalah *Precision* yang dihitung menggunakan Persamaan (2.1) , *Recall* yang dihitung menggunakan Persamaan (2.2), dan *Mean Average Precision* (mAP) yang dihitung menggunakan Persamaan (2.4) dan Persamaan (2.5).

Performa Model *Teacher* pada Dataset Parsial

Bagian ini menyajikan hasil performa model *Teacher* yang dilatih menggunakan anotasi bawaan dari Dataset Parsial. Hasil performa pada data latih (*train set*), data validasi (*validation set*), dan data uji (*test set*) pada Dataset Parsial disajikan pada Tabel 4.3, Tabel 4.4, dan Tabel

4.5. Berdasarkan Tabel 4.3, Tabel 4.4, dan Tabel 4.5, terlihat adanya indikasi *overfitting* yang cukup kuat pada skenario dengan data minim (khususnya L20). Pada skenario L20, model mampu mencapai mAP@50 sebesar 0,948 pada *train set*, namun kinerjanya menurun drastis menjadi 0,546 saat dievaluasi pada *test set*. Kesenjangan (*gap*) yang sangat besar ini menunjukkan bahwa model cenderung menghafal data latih, namun gagal melakukan generalisasi pada data yang belum pernah dilihat. Meskipun demikian, seiring bertambahnya persentase rasio data berlabel, *gap* performa antara data latih dan data uji semakin menyempit. Hal ini menunjukkan bahwa stabilitas generalisasi model mulai membaik secara signifikan mulai dari skenario L50 ke atas.

Tabel 4.3 Hasil Performa Model *Teacher* pada *Train Set* Dataset Parsial

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,545	0,513	0,530	0,502
L20	0,870	0,872	0,948	0,908
L30	0,891	0,955	0,978	0,949
L40	0,964	0,983	0,992	0,963
L50	0,984	0,982	0,992	0,964
L60	0,965	0,969	0,988	0,949
L70	0,989	0,991	0,994	0,965
L80	0,978	0,982	0,993	0,957
L90	0,991	0,996	0,995	0,970
L100	0,994	0,996	0,995	0,970

Tabel 4.4 Hasil Performa Model *Teacher* pada *Validation Set* Dataset Parsial

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,553	0,291	0,325	0,299
L20	0,664	0,566	0,637	0,594
L30	0,730	0,762	0,827	0,772
L40	0,801	0,821	0,869	0,821
L50	0,850	0,839	0,899	0,846
L60	0,841	0,863	0,912	0,863
L70	0,848	0,897	0,916	0,873
L80	0,869	0,900	0,935	0,883
L90	0,870	0,950	0,949	0,904
L100	0,906	0,947	0,962	0,918

Tabel 4.5 Hasil Performa Model *Teacher* pada *Test Set* Dataset Parsial

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,427	0,372	0,351	0,313
L20	0,583	0,518	0,546	0,501
L30	0,766	0,764	0,840	0,784
L40	0,807	0,780	0,855	0,802
L50	0,875	0,743	0,856	0,800
L60	0,883	0,843	0,902	0,844
L70	0,832	0,894	0,924	0,864
L80	0,863	0,886	0,938	0,877
L90	0,931	0,907	0,948	0,898
L100	0,929	0,914	0,950	0,897

Performa Model *Teacher* pada Dataset Menyeluruh

Bagian ini menyajikan hasil pelatihan model *Teacher* menggunakan Dataset Menyeluruh yang telah dianotasi ulang secara independen dan lebih presisi oleh peneliti. Hasil performa pada data latih (*train set*), data validasi (*validation set*), dan data uji (*test set*) pada Dataset Menyeluruh disajikan pada Tabel 4.6, Tabel 4.7, dan Tabel 4.8. Berdasarkan Tabel 4.6, Tabel 4.7, dan Tabel 4.8, hasil evaluasi menunjukkan pola konvergensi yang sedikit berbeda pada Dataset Menyeluruh. Pada skenario L10, model mengalami kesulitan untuk mempelajari fitur, ditunjukkan dengan nilai mAP@50 *train set* yang hanya mencapai 0,338 dan kembali anjlok menjadi 0,194 pada *test set*. Kondisi ini mengindikasikan bahwa gaya anotasi mandiri yang rapat, konsisten, dan komprehensif (termasuk mengakomodasi varian merek lain) menuntut representasi data latih awal yang jauh lebih memadai agar model terhindar dari ketidakseimbangan kelas (*class imbalance*) yang parah. Namun demikian, model berhasil menunjukkan lonjakan pemahaman yang progresif setelah melampaui skenario L30.

Tabel 4.6 Hasil Performa Model *Teacher* pada *Train Set* Dataset Menyeluruh

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,648	0,332	0,338	0,320
L20	0,704	0,770	0,786	0,740
L30	0,793	0,899	0,930	0,876
L40	0,909	0,858	0,953	0,899
L50	0,912	0,950	0,968	0,916
L60	0,964	0,960	0,979	0,928
L70	0,972	0,971	0,987	0,932
L80	0,964	0,970	0,987	0,927
L90	0,977	0,982	0,990	0,928
L100	0,984	0,985	0,992	0,938

Tabel 4.7 Hasil Performa Model *Teacher* pada *Validation Set* Dataset Menyeluruh

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,487	0,231	0,195	0,178
L20	0,597	0,531	0,569	0,519
L30	0,630	0,714	0,743	0,681
L40	0,777	0,706	0,769	0,711
L50	0,783	0,864	0,873	0,811
L60	0,868	0,850	0,907	0,845
L70	0,882	0,888	0,925	0,858
L80	0,883	0,886	0,926	0,862
L90	0,891	0,904	0,938	0,873
L100	0,917	0,886	0,944	0,882

Tabel 4.8 Hasil Perfoma Model *Teacher* pada *Test Set* Dataset Menyeluruh

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,584	0,182	0,194	0,172
L20	0,624	0,511	0,548	0,490
L30	0,720	0,746	0,783	0,718
L40	0,694	0,735	0,773	0,706
L50	0,809	0,806	0,830	0,760
L60	0,873	0,822	0,853	0,784
L70	0,821	0,851	0,860	0,790
L80	0,843	0,874	0,881	0,817
L90	0,893	0,867	0,883	0,823
L100	0,904	0,863	0,886	0,826

Secara komparatif, analisis pada kedua varian dataset mengonfirmasi bahwa keterbatasan kuantitas data berlabel berbanding lurus dengan kecenderungan terjadinya *overfitting*. Hal yang perlu disoroti adalah Dataset Menyeluruh memiliki *gap* performa antara *train* dan *test* yang cenderung lebih konsisten serta menghasilkan nilai mAP akhir (L100) yang tampak lebih

rendah (0,886) dibandingkan Dataset Parsial (0,950). Penurunan angka metrik ini bukan bermakna performa model pada Dataset Menyeluruh lebih buruk secara objektif, melainkan merupakan konsekuensi dari standar kualitas anotasi *bounding box* pada Dataset Menyeluruh yang jauh lebih ketat dan rapat. Akibatnya, prediksi dari model dipaksa untuk melampaui ambang batas *Intersection over Union* (IoU) secara jauh lebih sempurna jika dibandingkan dengan toleransi evaluasi pada label yang longgar milik Dataset Parsial.

4.3.2 Hasil Proses *Pseudo-Labeling*

Setelah model *Teacher* dari masing-masing skenario selesai dilatih, langkah selanjutnya dalam metode *Noisy Student* adalah menghasilkan label semu (*pseudo-label*) pada himpunan data tidak berlabel (*unlabeled data*). Setiap model *Teacher* melakukan inferensi untuk menghasilkan *pseudo-label* dengan *confidence threshold* sebesar 0,70 dan *IoU threshold* sebesar 0,60. Penggunaan *confidence threshold* sebesar 0,70 dilakukan untuk meningkatkan kepastian prediksi sehingga mengurangi kemungkinan model *Student* belajar dari prediksi yang kurang akurat. Sementara itu, penetapan IoU sebesar 0,60 digunakan untuk menjaga keseimbangan antara *precision* dan *recall*, khususnya pada kondisi produk ritel yang disusun saling berdempetan (*densely packed*), sehingga objek yang berdekatan tetap dipertahankan selama proses *Non-Maximum Suppression* (NMS). Kualitas *pseudo-label* ini dievaluasi secara mendalam dengan cara membandingkan *bounding box* hasil prediksi model terhadap label asli (*ground truth*) dari citra tersebut, yang anotasinya sengaja disembunyikan selama fase pelatihan. Analisis hasil *pseudo-labeling* ini dibagi menjadi dua aspek, yaitu evaluasi kuantitas (jumlah *bounding box*) dan evaluasi kualitas (akurasi prediksi terhadap *ground truth* yang disembunyikan).

Kuantitas dan Statistik Distribusi *Pseudo-Label*

Bagian ini menampilkan jumlah *bounding box* yang berhasil diproduksi oleh model *Teacher* pada data tidak berlabel (*unlabeled set*). Statistik hasil *pseudo-labeling* pada Dataset Parsial dan Dataset Menyeluruh disajikan pada Tabel 4.9 dan Tabel 4.10. Analisis hasil Tabel 4.9 dan Tabel 4.10 menunjukkan bahwa terdapat perbedaan yang besar pada total *bounding box* yang dihasilkan antara Dataset Parsial dan Dataset Menyeluruh. Pada L10, Dataset Parsial hanya menghasilkan 336 *bounding box* dari 296 citra (rata-rata 1,14 *bounding box*/citra), sementara Dataset Menyeluruh menghasilkan 1235 *bounding box* (rata-rata 4,17 *bounding box*/citra). Hal ini mencerminkan bahwa model *Teacher* dari Dataset Menyeluruh dapat

menghasilkan lebih banyak prediksi per gambar meskipun mAP-nya lebih rendah. Hal ini disebabkan oleh perbedaan standar anotasi yang ada. Dataset Menyeluruh memiliki anotasi yang lebih rapat dan menyeluruh per gambar, sehingga model *Teacher* cenderung memprediksi lebih banyak objek per citra. Jumlah label yang lebih banyak juga berperan besar untuk menghasilkan *bounding box* yang lebih banyak pada Dataset Menyeluruh.

Tabel 4.9 Statistik *Pseudo-Labeling* Dataset Parsial

Skenario	**Jumlah Data Tidak Berlabel**	**Jumlah *Bounding Box* yang Dihasilkan**	**Rata-rata *Bounding Box*/Citra**
L10	298	336	1.12
L20	263	1157	4.39
L30	230	1196	5.20
L40	197	1224	6.21
L50	164	1026	6.25
L60	132	724	5.48
L70	99	654	6.60
L80	66	387	5.86
L90	33	223	6.75

Tabel 4.10 Statistik *Pseudo-Labeling* Dataset Menyeluruh

Skenario	**Jumlah Data Tidak Berlabel**	**Jumlah *Bounding Box* yang Dihasilkan**	**Rata-rata *Bounding Box*/Citra**
L10	298	1235	4.14
L20	263	2652	10.08
L30	230	2527	10.98
L40	197	2704	13.72
L50	164	2338	14.25
L60	132	1947	14.75
L70	99	1438	14.52
L80	66	823	12.46
L90	33	485	14.69

Kualitas dan Evaluasi *Pseudo-Label* terhadap *Ground Truth*

Untuk mengukur kelayakan *pseudo-label* sebagai bahan ajar bagi model *Student*, evaluasi kualitas dilakukan dengan membandingkan hasil prediksi model *Teacher* terhadap label asli yang sengaja disembunyikan (*hidden ground truth*). Evaluasi ini mengekstrak parameter berupa *True Positive* (TP), *False Positive* (FP), dan *False Negative* (FN) menggunakan batas *Intersection over Union* (IoU) sebesar 0,50 yang dihitung menggunakan Persamaan (2.3). Selain ketiga metrik tersebut, kualitas *pseudo-label* juga direpresentasikan melalui metrik *Precision*, *Recall*, dan *mean Average Precision* (mAP). *Precision*, *Recall,* dan

mean Average Precision (mAP) didapatkan menggunakan Persamaan (2.1), Persamaan (2.2), dan Persamaan (2.5).

Hasil evaluasi pada kedua dataset disajikan pada Tabel 4.11 dan Tabel 4.12. Berdasarkan Tabel 4.11 dan Tabel 4.12, terlihat jelas mengapa skenario dengan data latih sedikit, seperti L10 dan L20, berisiko tinggi jika langsung digunakan dalam metode *Noisy Student*. Pada skenario L10 Dataset Menyeluruh, meskipun model *Teacher* berhasil menebak benar 878 objek (TP), model ini melewatkan jumlah objek yang sangat masif, yaitu 3255 objek (FN). Hal ini mengakibatkan nilai mAP@50 turun hingga 0,088. Jika *pseudo-label* dengan FN setinggi ini dipaksakan untuk melatih model *Student*, maka model *Student* akan dilatih untuk menganggap ribuan produk di rak sebagai latar belakang (*background*), yang berpotensi merusak performa model *Student* itu sendiri. Tingginya angka FN pada Dataset Menyeluruh dibandingkan Dataset Parsial di skenario awal juga kembali menegaskan bahwa gaya anotasi mandiri yang padat menuntut lebih banyak contoh awal agar model bisa mendeteksi seluruh objek di satu citra.

Tabel 4.11 Evaluasi Kualitas *Pseudo-Label* pada Dataset Parsial

Skenario	TP	FP	FN	Precision	Recall	mAP@50	mAP@50-95
L10	203	133	1592	0,604	0,113	0,150	0,139
L20	870	287	764	0,751	0,532	0,573	0,538
L30	990	206	466	0,827	0,679	0,676	0,636
L40	1022	202	217	0,835	0,824	0,821	0,777
L50	857	169	189	0,835	0,819	0,850	0,814
L60	601	123	131	0,830	0,821	0,840	0,802
L70	579	75	60	0,885	0,906	0,911	0,870
L80	346	41	34	0,894	0,910	0,901	0,865
L90	202	21	9	0,905	0,957	0,943	0,916

Tabel 4.12 Evaluasi Kualitas *Pseudo-Label* pada Dataset Menyeluruh

Skenario	TP	FP	FN	Precision	Recall	mAP@50	mAP@50-95
L10	878	357	3255	0,710	0,212	0,088	0,081
L20	2176	476	1541	0,820	0,585	0,429	0,391
L30	2131	396	1024	0,843	0,675	0,572	0,530
L40	2288	416	555	0,846	0,804	0,680	0,634
L50	2001	337	303	0,855	0,868	0,815	0,756
L60	1732	215	171	0,889	0,910	0,880	0,813
L70	1289	149	134	0,896	0,905	0,887	0,822
L80	721	102	58	0,876	0,925	0,880	0,823
L90	429	56	39	0,884	0,916	0,930	0,884

Di sisi lain, parameter *False Positive* (FP) menunjukkan pola yang menarik. Pada Dataset Menyeluruh, angka tebakan salah tertinggi terjadi pada skenario L20 (476 FP), bukan L10 (357 FP). Hal ini mengindikasikan bahwa pada L10, model masih terlalu "takut" atau belum cukup pintar untuk menebak, sehingga lebih banyak melewatkan objek. Memasuki L20, model mulai berani memprediksi lebih banyak kotak, namun karena pemahamannya belum matang, model justru banyak berhalusinasi (FP meningkat). Halusinasi (FP) inilah yang menjadi *noise* negatif terbesar yang mengganggu proses belajar model *Student*.

Kualitas *pseudo-label* mulai menunjukkan stabilitas yang ideal saat memasuki skenario L40 hingga L60. Pada rentang *sweet spot* ini (misalnya pada L50 Dataset Menyeluruh), model *Teacher* berhasil memproduksi 2001 tebakan benar (TP) dengan menekan tebakan salah (FP) menjadi 337 dan objek terlewat (FN) menjadi 303, menghasilkan mAP@50 sebesar 0,815. Kondisi di mana kurva TP melambung tinggi sementara kurva FP dan FN menurun tajam merupakan indikator krusial bahwa *pseudo-label* yang dihasilkan sudah cukup bersih dan layak. Label buatan yang bersih dari halusinasi dan kaya akan deteksi objek inilah yang akan membuat model *Student* untuk melampaui kemampuan *Teacher*-nya di tahap selanjutnya.

4.3.3 Hasil Pelatihan Model *Student*

Model *Student* adalah model YOLOv12-Nano yang dilatih menggunakan gabungan data berlabel asli dan *pseudo-label* dari model *Teacher*. Untuk memaksa model *Student* membangun representasi fitur yang lebih kuat (*robust*) dan mampu mengoreksi kesalahan *pseudo-label* dari model *Teacher*, diterapkan mekanisme *noise* berupa augmentasi data yang lebih agresif. Konfigurasi augmentasi meliputi *mixup* sebesar 0,1, variasi *scale* sebesar 0,5, *translate* sebesar 0,1, serta skema penurunan *learning rate* berbasis *cosine learning rate decay*.

Selain itu, model *Student* diinisialisasi menggunakan bobot YOLOv12-Nano baru (*cold start*) dan bukan dari bobot model *Teacher*. Langkah ini diambil untuk memutus rantai pewarisan bias (*bias propagation*) dari model *Teacher*, sehingga model *Student* memiliki ruang metrik yang independen untuk berkembang melampaui batas kemampuan model *Teacher*. Evaluasi komprehensif dilakukan pada tiga partisi (*train set*, *validation set*, dan *test set*) untuk menguji keberhasilan model *Student* ini. Metrik evaluasi yang digunakan adalah *Precision* yang dihitung menggunakan Persamaan (2.1) , *Recall* yang dihitung menggunakan Persamaan (2.2), dan *mean Average Precision* (mAP) yang dihitung menggunakan Persamaan (2.4) dan Persamaan (2.5).

Performa Model *Student* pada Dataset Parsial

Bagian ini menyajikan hasil performa model *Student* yang dilatih menggunakan kombinasi data berlabel dan *pseudo-label* dari Dataset Parsial. Hasil performa pada data latih (*train set*), data validasi (*validation set*), dan data uji (*test set*) pada Dataset Parsial disajikan pada Tabel 4.13, Tabel 4.14, dan Tabel 4.15. Berdasarkan data tersebut, penambahan komponen *noise* berupa augmentasi agresif (*mixup* dan *scaling*) berhasil memperkecil *generalization gap* antara data latih dan data uji, khususnya pada skenario data sangat terbatas. Sebagai komparasi, model *Teacher* L20 memiliki mAP@50 *train* sebesar 0,948 dan *test* sebesar 0,546 (selisih 0,402). Sementara pada model *Student* L20, mAP@50 *train* berada di angka 0,938 dan *test* di angka 0,589 (selisih mengecil menjadi 0,349). Hal ini menunjukkan bahwa pemaksaan regularisasi melalui *noise* efektif mencegah model YOLOv12 dari kondisi *overfitting* berat dan memaksa model mempelajari struktur objek ritel secara lebih general.

Tabel 4.13 Hasil Performa Model *Student* pada *Train Set* Dataset Parsial

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,637	0,627	0,660	0,627
L20	0,859	0,935	0,938	0,914
L30	0,925	0,913	0,963	0,940
L40	0,967	0,904	0,933	0,908
L50	0,975	0,974	0,989	0,964
L60	0,962	0,955	0,985	0,948
L70	0,967	0,974	0,990	0,954
L80	0,971	0,981	0,991	0,958
L90	0,968	0,982	0,989	0,949

Tabel 4.14 Hasil Performa Model *Student* pada *Validation Set* Dataset Parsial

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,684	0,344	0,391	0,366
L20	0,710	0,619	0,662	0,620
L30	0,766	0,729	0,807	0,759
L40	0,777	0,875	0,872	0,820
L50	0,838	0,865	0,884	0,832
L60	0,846	0,868	0,906	0,853
L70	0,863	0,886	0,907	0,859
L80	0,887	0,882	0,925	0,880
L90	0,881	0,923	0,949	0,900

Tabel 4.15 Hasil Performa Model *Student* pada *Test Set* Dataset Parsial

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,614	0,324	0,359	0,330
L20	0,643	0,516	0,589	0,543
L30	0,735	0,749	0,808	0,762
L40	0,856	0,789	0,858	0,799
L50	0,873	0,787	0,854	0,801
L60	0,834	0,875	0,895	0,838
L70	0,867	0,900	0,934	0,876
L80	0,923	0,871	0,947	0,895
L90	0,882	0,920	0,957	0,897

Performa Model *Student* pada Dataset Menyeluruh

Bagian ini menyajikan hasil performa model *Student* yang dilatih menggunakan kombinasi data berlabel dan *pseudo-label* dari Dataset Menyeluruh. Hasil performa pada data latih (*train*

set), data validasi (*validation set*), dan data uji (*test set*) pada Dataset Menyeluruh disajikan pada Tabel 4.16,
Tabel 4.17, dan Tabel 4.18. Berdasarkan data tersebut, terlihat ada tren peningkatan performa yang konsisten pada Dataset Menyeluruh seiring dengan bertambahnya rasio data berlabel. Pada *train set* (Tabel 4.16), nilai mAP@50 meningkat secara masif dari 0,477 pada skenario L10 hingga mencapai 0,993 pada skenario L90. Hal ini menunjukkan kemampuan model dalam mengenali pola data latih dengan sangat baik. Namun, pada skenario rendah (L10), terdapat *generalization gap* yang cukup lebar di mana *test* mAP@50 hanya berada di angka 0,235 karena keterbatasan variasi sampel berlabel di awal. Tren pada *validation set* (
Tabel 4.17) dan *test set* (Tabel 4.18) menunjukkan fase stagnasi sekaligus fluktasi kecil pada rentang skenario L30 (mAP@50 sebesar 0,755) dan L40 (mAP@50 sebesar 0,744) pada data uji. Kondisi ini mengindikasikan bahwa arsitektur YOLOv12-Nano sedang beradaptasi keras terhadap beban regularisasi berupa *noise augmentation*. Titik balik optimal (*turning point*) akhirnya terjadi pada skenario L50, di mana performa pada data uji melonjak tajam ke angka 0,861 dan performa pada data validasi mencapai 0,890, menandakan bahwa model telah berhasil mencapai stabilitas konvergensi dan mampu menggeneralisasi fitur objek ritel secara matang.

Tabel 4.16 Hasil Performa Model *Student* pada *Train Set* Dataset Menyeluruh

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,545	0,461	0,477	0,452
L20	0,759	0,752	0,816	0,782
L30	0,847	0,798	0,870	0,829
L40	0,937	0,812	0,865	0,832
L50	0,924	0,942	0,969	0,934
L60	0,955	0,966	0,979	0,938
L70	0,975	0,966	0,986	0,940
L80	0,957	0,972	0,985	0,933
L90	0,984	0,977	0,993	0,937

Tabel 4.17 Hasil Performa Model *Student* pada *Validation Set* Dataset Menyeluruh

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,475	0,197	0,270	0,242
L20	0,629	0,590	0,619	0,568
L30	0,645	0,715	0,777	0,714
L40	0,796	0,724	0,792	0,734
L50	0,814	0,823	0,890	0,827
L60	0,872	0,864	0,914	0,855
L70	0,884	0,866	0,919	0,856
L80	0,853	0,902	0,921	0,860
L90	0,912	0,887	0,947	0,881

Tabel 4.18 Hasil Performa Model *Student* pada *Test Set* Dataset Menyeluruh

Skenario	Precision	Recall	mAP@50	mAP@50-95
L10	0,557	0,200	0,235	0,210
L20	0,670	0,540	0,581	0,525
L30	0,707	0,733	0,755	0,689
L40	0,809	0,652	0,744	0,683
L50	0,813	0,807	0,861	0,794
L60	0,883	0,806	0,848	0,789
L70	0,856	0,838	0,869	0,805
L80	0,856	0,848	0,878	0,817
L90	0,885	0,844	0,879	0,815

Selanjutnya, analisis komparatif antara performa model *Student* pada Dataset Menyeluruh dan Dataset Parsial menunjukkan hasil yang unik terkait pengaruh kerapatan anotasi terhadap algoritma *Semi-Supervised Learning*. Pada kondisi data berlabel yang minim (skenario L10 hingga L40), model *Student* dengan Dataset Parsial secara konsisten menghasilkan performa data uji yang lebih unggul dibandingkan dengan Dataset Menyeluruh (seperti pada skenario L10 di mana Dataset Parsial mencapai mAP@50 sebesar 0,359

dibandingkan 0,235 pada Dataset Menyeluruh). Hal ini membuktikan bahwa dengan kuantitas data yang sangat terbatas, model lebih mudah memetakan fitur dan memenuhi ambang batas IoU dari standar *bounding box* Dataset Parsial yang cenderung lebih longgar.

Sebaliknya, standar anotasi yang sangat rapat dan ketat pada Dataset Menyeluruh membutuhkan tingkat presisi lokalisasi objek yang jauh lebih tinggi. Meskipun demikian, pola yang berbeda mulai terlihat saat memasuki skenario L50. Pada jumlah 50% data berlabel ini, Dataset Menyeluruh berhasil menghasilkan mAP@50 sebesar 0,861, melampaui Dataset Parsial yang berada di angka 0,854. Keunggulan Dataset Menyeluruh yang stabil dari skenario L50 hingga L90 membuktikan bahwa ketika volume data dasar sudah mencukupi, kualitas anotasi yang menyeluruh dan presisi mampu diserap secara maksimal oleh algoritma *Noisy Student* untuk menghasilkan model deteksi produk ritel yang lebih bagus. Hasil ini juga menguatkan skenario L50 sebagai *sweet spot* yang paling efisien dalam menyeimbangkan kebutuhan anotasi dan performa model pada penelitian ini.

4.4 Analisis Hasil

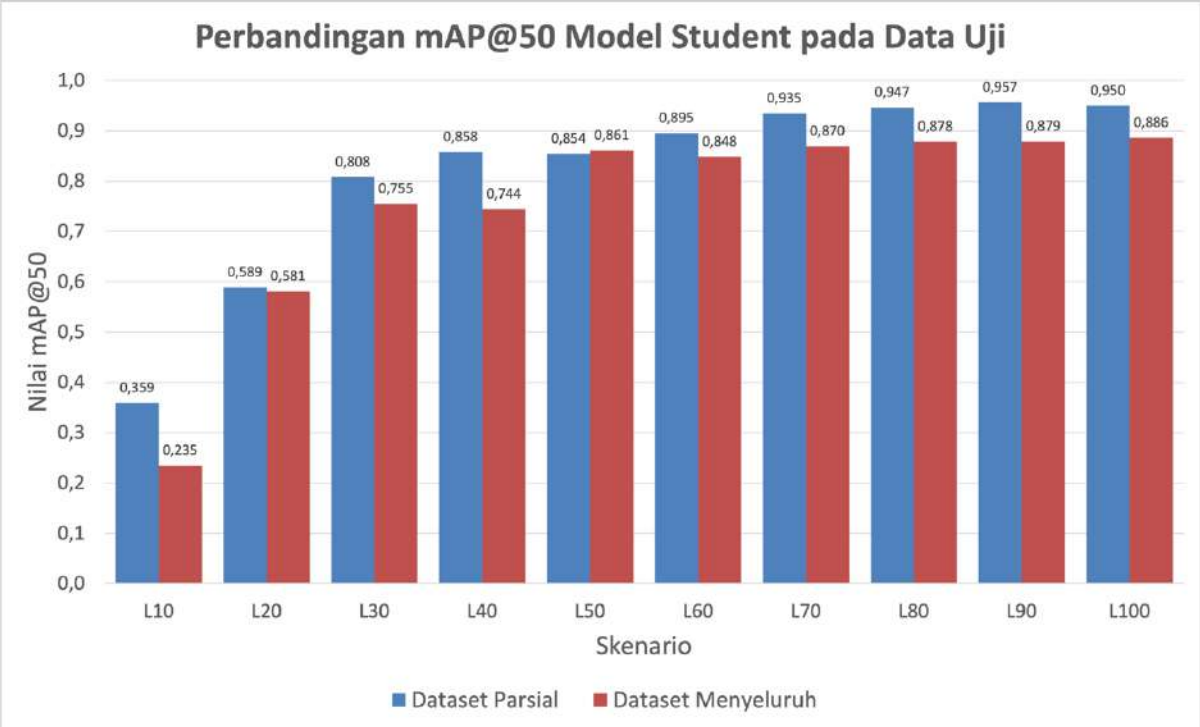
Subbab ini menyajikan analisis mendalam dan pembahasan komprehensif terhadap seluruh hasil eksperimen penerapan metode *Noisy Student* berbasis YOLOv12 yang telah dipaparkan pada subbab sebelumnya. Fokus utama analisis ini meliputi evaluasi metrik performa model, karakteristik konvergensi, pola penurunan *loss*, identifikasi *sweet spot* untuk menentukan titik efisiensi optimal dalam penggunaan data berlabel, hingga analisis visual hasil deteksi model.

4.4.1 Perbandingan mAP, *Precision*, dan *Recall* Antar Skenario

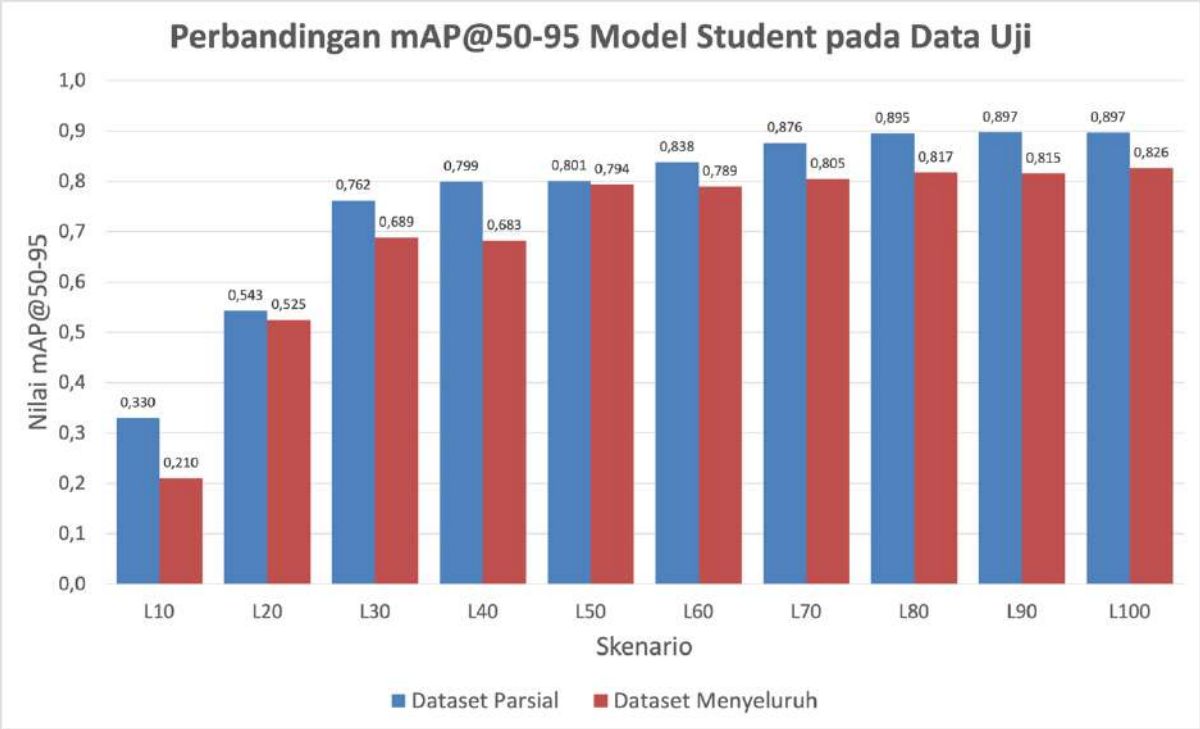
Evaluasi performa utama dari model *Student* pada setiap skenario diukur menggunakan metrik *mean Average Precision* (mAP) pada *test set*, khususnya mAP@50 dan mAP@50-95. Perbandingan metrik ini dilakukan untuk memetakan sejauh mana kualitas prediksi dan kemampuan deteksi model ketika dilatih menggunakan Dataset Parsial dibandingkan dengan Dataset Menyeluruh pada berbagai tingkat rasio data berlabel. Gambar 4.1 dan Gambar 4.2 menyajikan grafik perbandingan performa *test* mAP@50 dan *test* mAP@50-95 model *Student* di Dataset Parsial dan Dataset Menyeluruh per skenario.

Berdasarkan Gambar 4.1 dan Gambar 4.2, pada skenario dengan data berlabel terbatas (L10–L40) terdapat *gap* atau kesenjangan performa antara kedua dataset. Dataset Parsial secara konsisten menghasilkan mAP yang lebih tinggi pada skenario tersebut, namun *gap* ini semakin

mengecil seiring bertambahnya persentase data berlabel. Pada L50 ke atas, kedua dataset menunjukkan performa yang semakin konvergen.



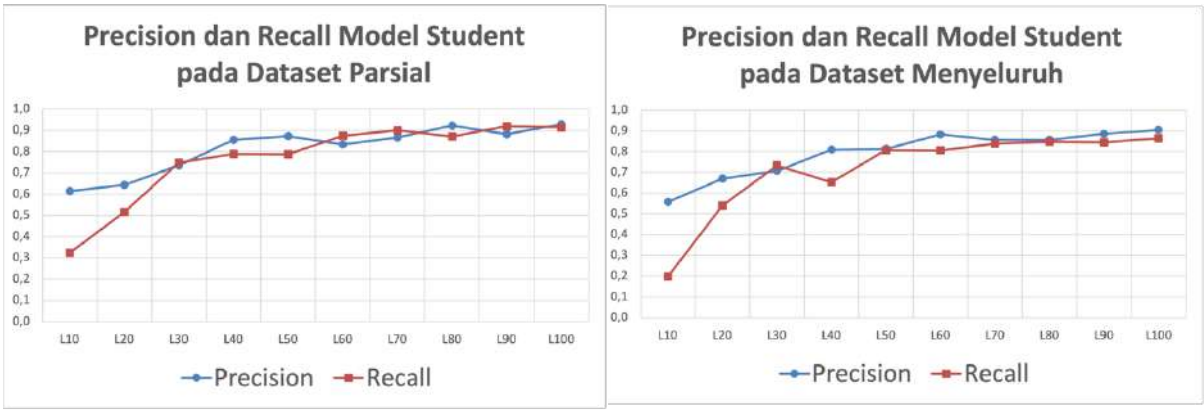
Gambar 4.1 Perbandingan mAP@50 Model *Student* pada Data Uji per Skenario



Gambar 4.2 Perbandingan mAP@50-95 Model *Student* pada Data Uji per Skenario

Selain mAP, metrik *Precision* juga dievaluasi untuk mengukur tingkat ketepatan atau kepastian model dalam menebak kelas objek. Tingkat *Precision* ini menunjukkan seberapa proporsional tebakan objek yang benar dari keseluruhan prediksi positif yang dihasilkan oleh model di masing-masing skenarionya. Di sisi lain, metrik *Recall* dievaluasi untuk mengukur kepekaan model dalam menemukan atau mendeteksi keseluruhan objek *ground truth* yang sebenarnya ada di dalam citra. Pengamatan terhadap metrik ini penting untuk mengetahui tingkat kebocoran (*false negative*) model. Pergerakan nilai *Precision* dan *Recall* untuk keseluruhan skenario pada kedua dataset disajikan pada Gambar 4.3.

Gambar 4.3 menunjukkan bahwa nilai *Precision* dan *Recall* tidak selalu bergerak secara bersamaan. Pada beberapa skenario, misalnya pada L50 di Dataset Parsial, nilai *Precision* tercatat tinggi namun *Recall* lebih rendah. Kondisi ini mengindikasikan bahwa model cenderung bersifat konservatif dalam membuat prediksi. Model hanya memprediksi objek yang benar-benar diyakininya, namun berpotensi melewatkan sebagian objek lain yang sebenarnya ada di dalam citra.



(a) (b)

Gambar 4.3 *Precision* dan *Recall* Model *Student* per Skenario (a) Dataset Parsial dan (b) Dataset Menyeluruh

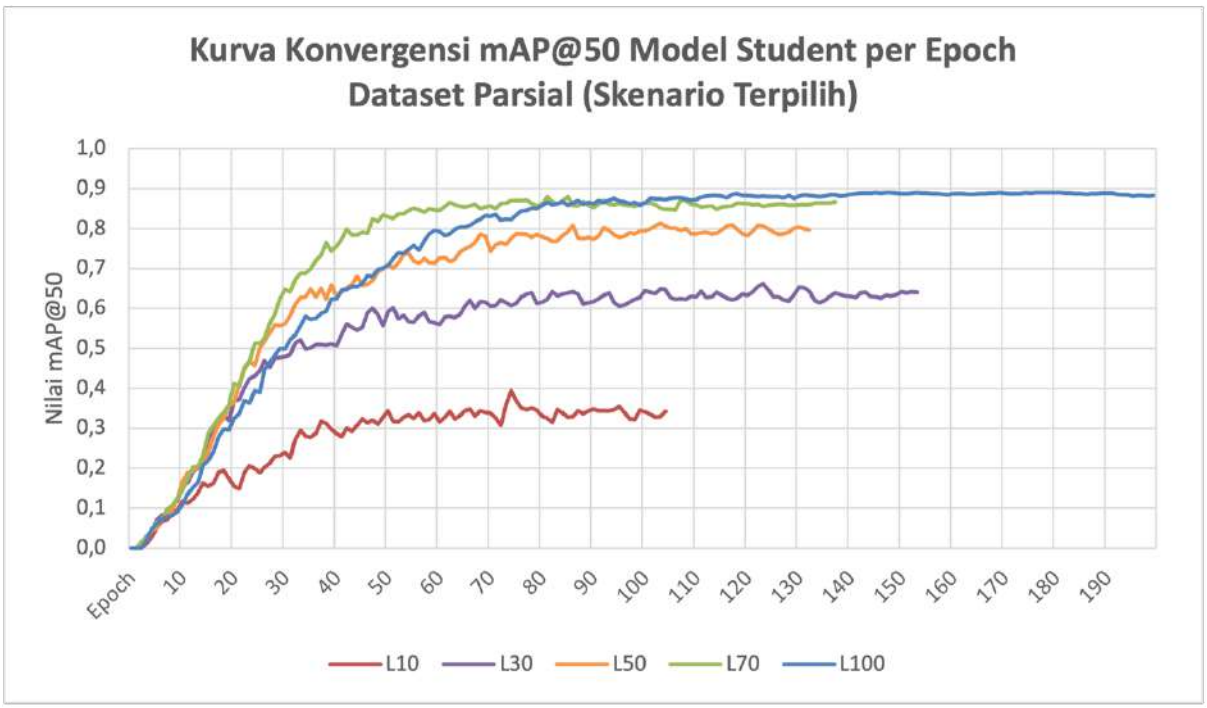
4.4.2 Kurva Konvergensi mAP per *Epoch*

Analisis terhadap kurva konvergensi dilakukan untuk mengevaluasi stabilitas dan efisiensi proses pembelajaran model *Student* pada setiap skenario selama siklus pelatihan (*training*). Visualisasi mAP@50 terhadap jumlah *epoch* ini bertujuan untuk mengamati bagaimana model mencapai titik jenuh performanya dan bagaimana ketersediaan data berlabel memengaruhi kestabilan grafik pembelajaran. Melalui kurva ini, dapat diidentifikasi apakah model mengalami kendala dalam mempelajari fitur objek pada kondisi data yang sangat terbatas dibandingkan dengan kondisi data yang melimpah. Gambar 4.4 dan Gambar 4.5 menyajikan grafik kurva konvergensi mAP@50 model *Student* per *epoch* Dataset Parsial dan Dataset Menyeluruh pada skenario terpilih (L10, L30, L50, L70, L100).

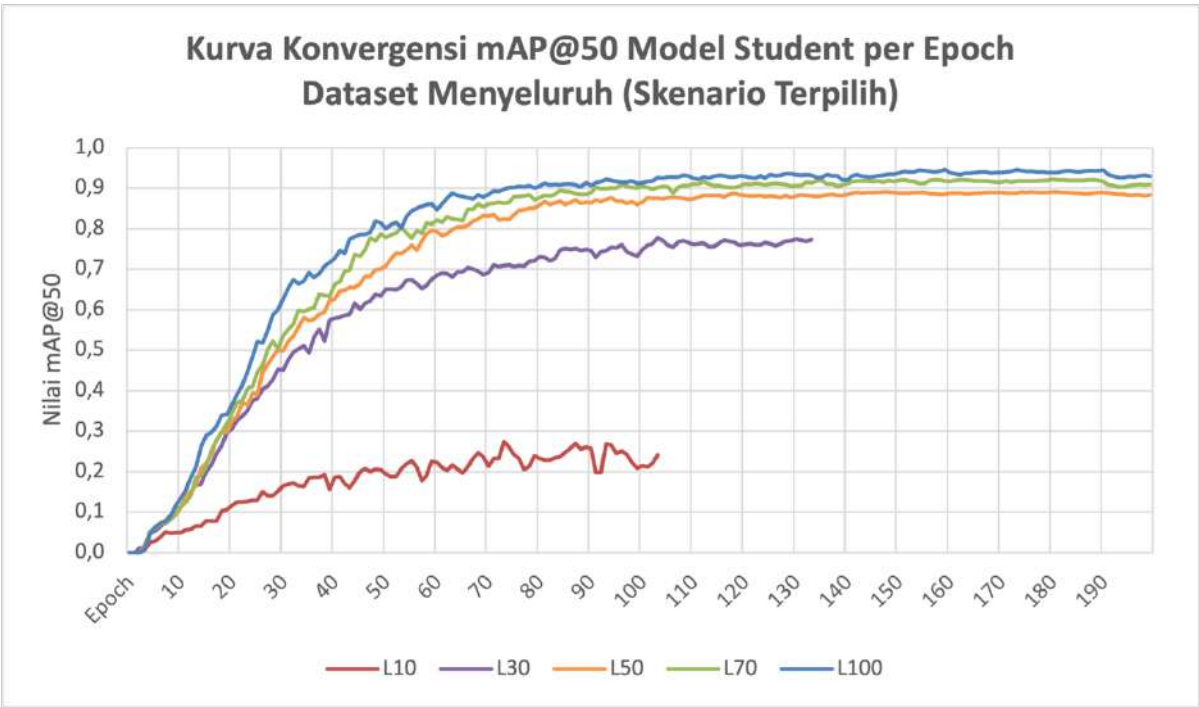
Berdasarkan Gambar 4.4 dan Gambar 4.5, terlihat perbedaan karakteristik konvergensi yang signifikan antar skenario eksperimen. Skenario L10 menunjukkan kurva yang tidak stabil dan fluktuatif di sepanjang proses pelatihan, yang mengindikasikan kesulitan model dalam mempelajari pola objek akibat kualitas *pseudo-label* yang rendah dari model *Teacher*. Sebaliknya, skenario L100 sebagai *baseline* menunjukkan konvergensi yang sangat mulus dan konsisten menuju nilai maksimumnya.

Temuan yang menarik terdapat pada skenario L50, di mana model mampu mencapai konvergensi yang relatif stabil mendekati nilai maksimum, namun dengan durasi *epoch* yang lebih banyak jika dibandingkan dengan skenario L30. Fenomena ini menunjukkan bahwa integrasi data tidak berlabel melalui metode *pseudo-labeling* memerlukan waktu pelatihan yang lebih panjang agar model *Student* dapat mengenali informasi tambahan tersebut secara

efektif dan stabil. Hal ini memperkuat indikasi bahwa L50 merupakan titik di mana model mulai mampu mengoptimalkan data tambahan dengan baik meskipun memerlukan waktu komputasi yang sedikit lebih lama.



Gambar 4.4 Kurva Konvergensi mAP@50 *Model Student* per Epoch Dataset Parsial pada Skenario Terpilih (L10, L30, L50, L70, L100)



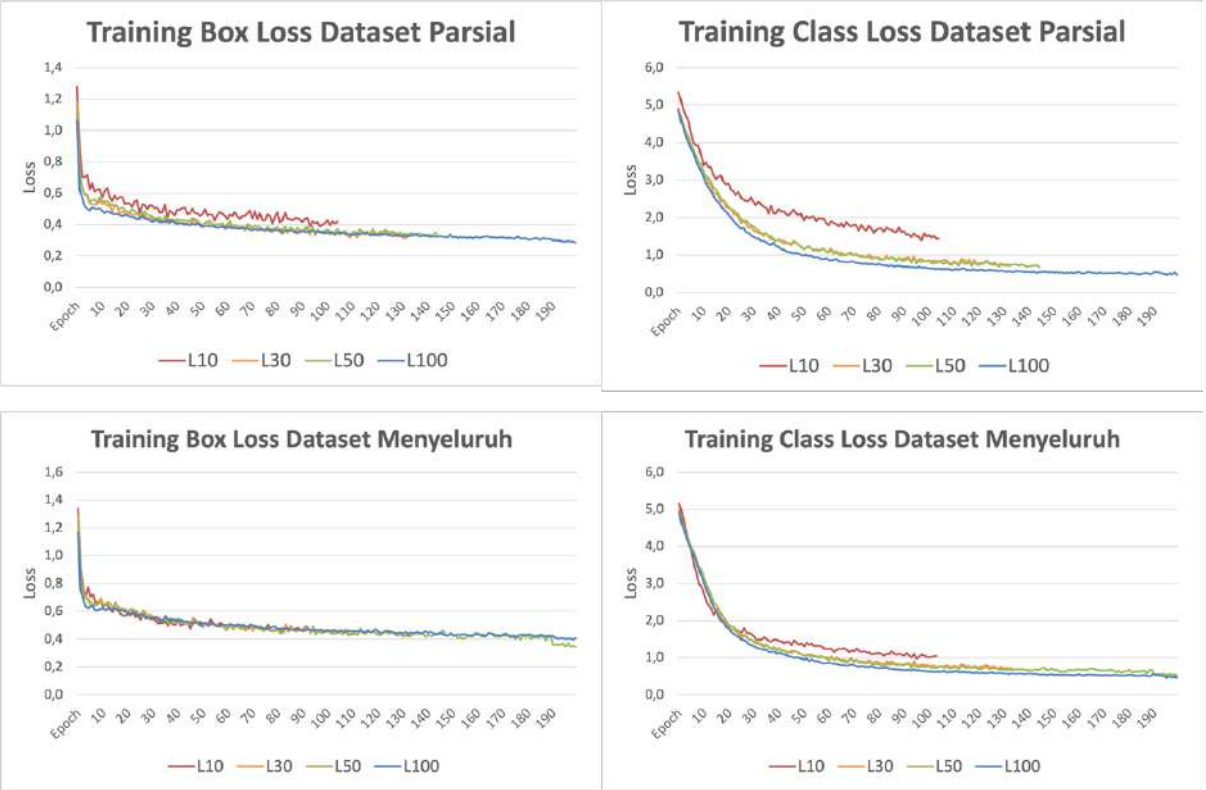
Gambar 4.5 Kurva Konvergensi mAP@50 *Model Student* per Epoch Dataset Menyeluruh pada Skenario Terpilih (L10, L30, L50, L70, L100)

4.4.3 Analisis *Training Box Loss* dan *Classification Loss*

Evaluasi performa model *Student* selama proses pelatihan (*training*) juga ditinjau berdasarkan grafik penurunan fungsi kerugian (*loss function*), yang secara spesifik terbagi menjadi *Training Box Loss* dan *Training Classification Loss*. *Box Loss* digunakan untuk mengukur tingkat akurasi model *Student* dalam melokalisasi koordinat pembatas (*bounding box*) objek produk ritel, sedangkan *Classification Loss* mengukur efektivitas model dalam mengklasifikasikan kategori produk secara tepat. Analisis terhadap kedua jenis *loss* ini bertujuan untuk mengamati stabilitas kemajuan belajar model pada setiap *epoch* serta mendeteksi ada tidaknya kendala seperti kesalahan pembelajaran akibat kualitas pelabelan mandiri. Lebih jauh lagi, kurva *loss* ini berfungsi sebagai instrumen diagnostik untuk mendeteksi anomali pelatihan, seperti indikasi *overfitting*, gradien yang tidak stabil, atau efek perambatan kesalahan (*error propagation*) yang dipicu oleh rendahnya kualitas *pseudo-label* yang dihasilkan dari model *Teacher*. Perbandingan kurva *Training Box Loss* dan *Training Classification Loss* untuk beberapa skenario terpilih disajikan pada Gambar 4.6.

Berdasarkan Gambar 4.6, kurva *training loss* memperkuat temuan yang ada pada kurva konvergensi mAP. Skenario L10 terlihat mengalami lintasan penurunan *loss* yang lambat dan tidak monoton, yang mencerminkan adanya kesulitan nyata pada model dalam mempelajari

fitur objek dari hasil *pseudo-label* yang berkualitas rendah (hanya 336 *bounding box* dari 298 gambar). Sebaliknya, kurva pada skenario L100 (pendekatan *fully supervised*) merepresentasikan kondisi pelatihan yang ideal dengan tren penurunan *loss* yang konsisten, eksponensial di awal, dan mulus (*smooth*) hingga akhir *epoch*. Hal ini dimungkinkan karena pembaruan bobot jaringan (*weight update*) sepenuhnya dipandu oleh *ground truth* absolut yang bebas dari anomali label. Sementara itu, skenario L50 berada di antara kedua skenario tersebut dengan menunjukkan pola penurunan yang teratur dan stabil. Karakteristik penurunan grafik *loss* yang terstruktur pada skenario L50 ini menegaskan kembali posisinya sebagai *sweet spot*, di mana model *Student* mampu memanfaatkan tambahan data tidak berlabel dengan baik tanpa mengorbankan stabilitas fungsional model.



Gambar 4.6 Kurva *Training Loss* Model *Student* per *Epoch* pada Skenario Terpilih (L10, L30, L50, L100)

4.4.4 Analisis Efisiensi dan Identifikasi *Sweet Spot*

Identifikasi *sweet spot* dilakukan dengan menganalisis delta mAP@50 model *Student* pada data uji antar-skenario yang berurutan. *Sweet spot* dalam eksperimen ini ditetapkan pada titik kritis di mana penambahan proporsi data berlabel mampu memberikan kontribusi peningkatan mAP yang paling signifikan dan proporsional sebelum model memasuki fase kejenuhan atau *diminishing returns* (hukum penurunan hasil marjinal). Pendekatan ini bertujuan untuk menemukan rasio data minimal yang paling *cost-effective* bagi industri ritel, sehingga target akurasi tinggi dapat dicapai tanpa harus melakukan proses anotasi data secara penuh. Grafik interpretasi delta mAP serta visualisasi perbandingan skenario optimal menggunakan diagram radar disajikan pada Tabel 4.19. Hasil evaluasi menunjukkan adanya tiga pola utama yang melandasi proses identifikasi efisiensi model *Student*.

Pola pertama adalah terjadinya lonjakan performa yang sangat besar pada transisi awal skenario L10 ke L20, di mana Dataset Parsial mencatat delta mAP@50 sebesar +0,2302 (+64,2%) dan Dataset Menyeluruh mencatat lonjakan sebesar +0,3461 (+147,3%). Hal ini

terjadi karena pada skenario L10, model *Teacher* berada dalam kondisi kekurangan jumlah data latih berlabel (rata-rata kurang dari 1 gambar per kelas untuk 49 kelas produk ritel), sehingga *pseudo-label* yang dihasilkan sangat tidak lengkap. Penambahan sekitar 33 gambar berlabel pada skenario L20 secara instan meningkatkan kemampuan model *Teacher* untuk menghasilkan *bounding box* baru yang jauh lebih representatif bagi model *Student*.

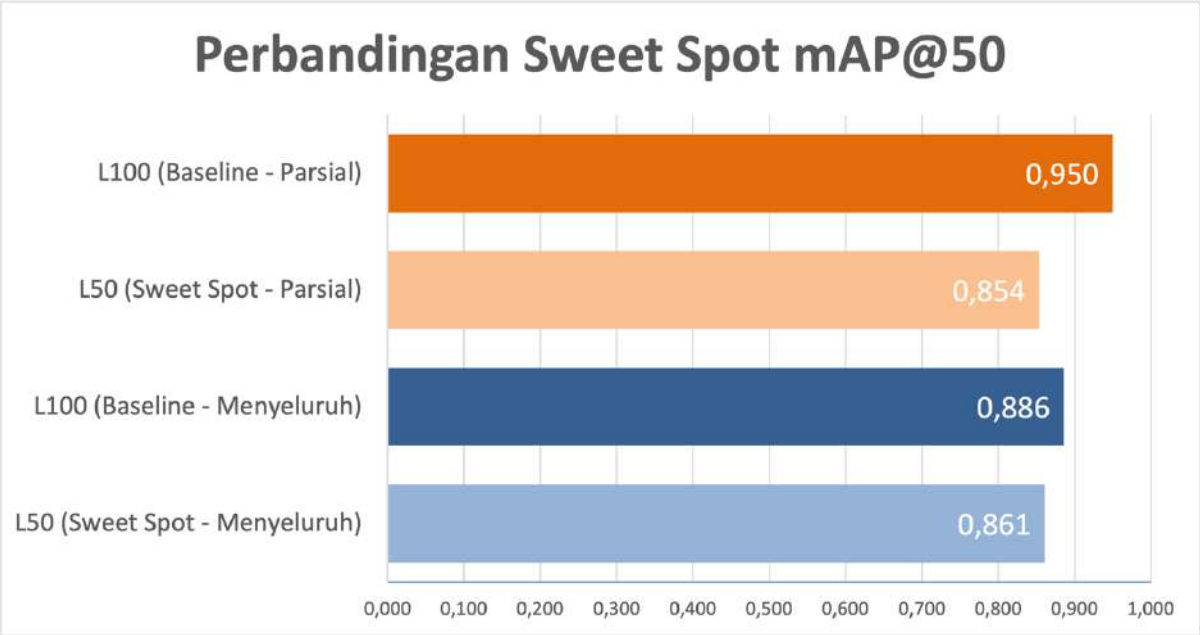
Pola kedua yang menjadi temuan inti penelitian ini adalah adanya *tipping point* atau lonjakan sekunder yang signifikan pada transisi skenario L40 ke L50. Setelah performa sempat mengalami pelambatan pada transisi skenario L30 ke L40, Dataset Menyeluruh kembali memperlihatkan lonjakan pertumbuhan mAP@50 yang impresif sebesar +0,1169 (+15,7%). Karakteristik ini menandakan bahwa batas 40% data berlabel bertindak sebagai ambang kecukupan bagi model *Teacher* untuk memperoleh representasi visual yang merata dari seluruh kelas produk ritel, sehingga data gabungan asli dan *pseudo-label* yang dihasilkan pada skenario L50 memiliki akurasi yang sangat tinggi.

Sebaliknya, pola ketiga menunjukkan terbentuknya fase *plateau* (landai) yang konsisten setelah melampaui skenario L60, di mana setiap penambahan investasi 10% data berlabel manual hanya memberikan pengembalian mAP marjinal di bawah 3%. Kondisi ini menegaskan berlakunya hukum *diminishing returns*, di mana pengeluaran biaya dan tenaga untuk pelabelan manual tambahan sudah tidak lagi sebanding dengan hasil akurasi model yang didapatkan. Rincian persentase dan selisih absolut untuk setiap transisi skenario yang mendasari penemuan ketiga pola tersebut disajikan secara komprehensif pada Tabel 4.19.

Tabel 4.19 Analisis Delta mAP@50 Model *Student* pada Data Uji Antar-Skenario untuk Identifikasi *Sweet Spot*

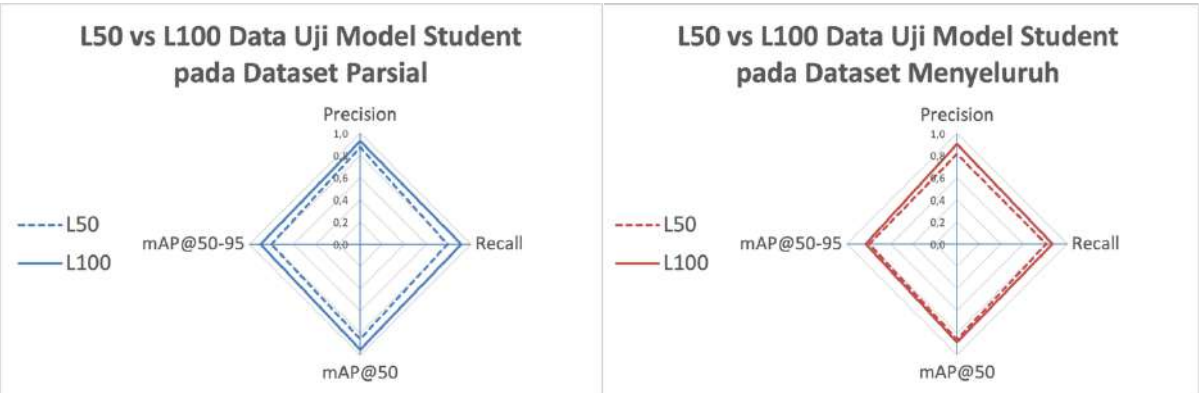
Transisi Skenario	**Δ mAP@50 Dataset Parsial**	**Δ mAP@50 Dataset Parsial (%)**	**Δ mAP@50 Dataset Menyeluruh**	**Δ mAP@50 Dataset Menyeluruh (%)**
L10 → L20	+0,2302	+64,2%	+0,3461	+147,3%
L20 → L30	+0,2189	+37,2%	+0,1737	+29,9%
L30 → L40	+0,0499	+6,2%	-0,0104	-1,4%
L40 → L50	-0,0033	-0,4%	+0,1169	+15,7%
L50 → L60	+0,0408	+4,8%	-0,0128	-1,5%
L60 → L70	+0,0393	+4,4%	+0,0211	+2,5%
L70 → L80	+0,0123	+1,3%	+0,0088	+1,0%
L80 → L90	+0,0099	+1,0%	+0,0003	+0,0%
L90 → L100	-0,0066	-0,7%	+0,0076	+0,9%

Melalui analisis ketiga komparasi pola tersebut, skenario L50 (50% data berlabel) teridentifikasi sebagai titik *sweet spot* terbaik pada kedua dataset. Untuk melihat kelayakan titik optimal ini, dilakukan perbandingan performa mAP@50 antara skenario *sweet spot* (L50) dengan skenario *baseline fully supervised* yang menggunakan 100% data berlabel (L100). Perbandingan *sweet spot* (L50) dengan *baseline fully supervised* (L100) model *Student* pada data uji dapat dilihat pada Gambar 4.7. Pada titik optimal ini, model *Student* mampu menghasilkan performa mAP@50 mencapai 0,861 pada Dataset Menyeluruh dan 0.854 pada Dataset Parsial. Hasil tersebut membuktikan bahwa dengan memotong setengah (50%) kewajiban anotasi manual, model deteksi berbasis YOLOv12 tetap mampu mempertahankan kekuatan akurasinya hingga setara dengan 97,2% dari pencapaian performa skenario *fully supervised baseline* (L100 = 0,886) pada Dataset Menyeluruh dan 89,9% dari performa skenario *fully supervised baseline* (L100 = 0,950) pada Dataset Parsial.



Gambar 4.7 Perbandingan *Sweet Spot* dengan *Baseline Fully Supervised*

Penemuan ini diperkuat secara visual melalui *radar chart* pada Gambar 4.8, yang memperlihatkan area cakupan fungsional metrik skenario L50 Dataset Menyeluruh sudah sangat berimpit dengan skenario L100 di seluruh dimensi pengukuran, baik pada metrik *Precision*, *Recall*, mAP@50, maupun mAP@50-95. Ketiadaan selisih area yang signifikan ini memberikan kesimpulan praktis yang kuat bahwa metode *Noisy Student* sukses merekayasa efisiensi data tanpa mengorbankan reliabilitas performa akhir model deteksi produk ritel.



(a) (b)

Gambar 4.8 Radar Chart Perbandingan L50 (*Sweet Spot*) vs L100 (*Baseline*) (a) Dataset Parsial dan (b) Dataset Menyeluruh

4.4.5 Analisis Kualitatif Visual Hasil Deteksi

Untuk mengevaluasi kemampuan lokalisasi dan klasifikasi model secara kualitatif, dilakukan analisis visual terhadap hasil prediksi yang dihasilkan oleh model *Student* pada skenario optimal L50 di Dataset Menyeluruh. Analisis ini membandingkan anotasi *ground truth* dengan prediksi model untuk memahami karakteristik keberhasilan serta mengidentifikasi pola kesalahan deteksi pada citra rak produk ritel. Contoh visual hasil deteksi model *Student* pada skenario L50 disajikan pada Gambar 4.9. Terdapat tiga kasus yang dianalisis, yaitu kasus berhasil deteksi (*True Positive*), kasus objek terlewat (*False Negative*), dan kasus salah klasifikasi (*False Positive*). Setiap gambar ditampilkan berdampingan, di mana sisi kiri merupakan anotasi *ground truth* dan sisi kanan merupakan hasil prediksi model. Warna kotak pembatas (*bounding box*) merepresentasikan kelas produk yang berbeda dan bersifat konsisten antar kedua sisi untuk mempermudah perbandingan, sedangkan teks di atas setiap kotak pada sisi *ground truth* menunjukkan nama kelas, dan pada sisi prediksi menunjukkan nama kelas yang diikuti oleh angka desimal yang merepresentasikan *confidence score* atau tingkat keyakinan model terhadap prediksi tersebut (skala 0 hingga 1).

Gambar 4.9(a) menunjukkan contoh keberhasilan deteksi (*True Positive*) pada kondisi rak yang sangat padat. Pada citra ini terdapat lima objek pada *ground truth*, yaitu dua objek kelas “48_other_brand”, satu objek kelas “02_dcw_1plus_madu_400g”, satu objek kelas “06_dcw_1plus_van_400g”, dan satu objek kelas “10_dcw_3plus_madu_400g”. Kelima objek tersebut berhasil dideteksi model dengan presisi dan kelas yang tepat. Hal ini menunjukkan tidak ada kesalahan lokalisasi maupun klasifikasi pada kondisi rak yang padat ini. Keberhasilan ini menunjukkan bahwa arsitektur YOLOv12, ketika dikombinasikan dengan metode *Noisy Student* pada rasio data berlabel sebanyak 50%, mampu mengekstraksi fitur spasial dengan baik meskipun produk disusun rapat. Hal ini sangat penting mengingat produk-produk *Fast-Moving Consumer Goods* (FMCG) sering kali memiliki desain kemasan dan palet warna yang hampir identik (*high inter-class similarity*), di mana perbedaan antar kelas hanya terletak pada detail tekstual minor, seperti varian rasa (“madu” berbanding “vanila”) atau label peruntukan usia. Pemberian *pseudo-label* pada data yang tidak berlabel (50% sisanya) terbukti berhasil memperkaya pemahaman model terhadap variasi pencahayaan dan skala kemasan, sehingga model memiliki ketahanan (*robustness*) terhadap tumpukan objek yang padat.

Sebaliknya, Gambar 4.9(b) menunjukkan contoh kasus *False Negative*, yaitu objek terdapat *ground truth* tetapi tidak berhasil dideteksi oleh model. Pada citra tersebut, *ground truth* menunjukkan keberadaan produk, namun model *Student* tidak menghasilkan *bounding*

box untuk objek tersebut. *Ground truth* memuat tujuh objek, yaitu dua objek kelas "31_lact_pro_6_12_350g", satu objek kelas "46_lactg_pro_1plus_van_350g", satu objek kelas "10_dcw_3plus_madu_400g", satu objek kelas "13_dcw_3plus_van_400g", satu objek kelas "27_lact_pro_0_6_350g", dan satu objek kelas "42_lactg_1plus_madu_350g". Model hanya berhasil mendeteksi dua dari tujuh objek tersebut (kelas "10_dcw_3plus_madu_400g" dan "13_dcw_3plus_van_400g"), sehingga lima objek lain gagal terdeteksi sama sekali dan tidak dihasilkan *bounding box* oleh model. Kondisi ini mengindikasikan bahwa masih terdapat objek yang belum dapat dikenali secara konsisten oleh model meskipun telah melalui proses pelatihan. Faktor yang memengaruhi kondisi ini adalah karakteristik visual objek yang lebih menantang, seperti oklusi sebagian (*partial occlusion*) atau pantulan cahaya (*glare*) pada rak. Apabila kondisi seperti ini kurang terwakili dalam data berlabel, model *Teacher* berpotensi gagal menghasilkan *pseudo-label* yang sesuai pada data tidak berlabel. Akibatnya, informasi pembelajaran yang diterima model *Student* menjadi kurang lengkap sehingga objek tersebut tidak terdeteksi dan cenderung diperlakukan sebagai bagian dari latar belakang (*background*).

Sementara itu, Gambar 4.9(c) memperlihatkan contoh kasus *False Positive*, yaitu ketika model berhasil melokalisasi objek dengan tepat tetapi salah dalam menentukan kelas produk. Pada area objek yang dianotasi *ground truth* sebagai kelas "02_dcw_1plus_madu_400g", model berhasil menempatkan *bounding box* pada lokasi yang tepat, namun memprediksi kelas yang salah, yaitu "47_other_variant". Lokalisasi *bounding box* yang diprediksi model sudah sangat akurat menutupi kemasan produk, namun model salah memprediksi kelas objek. Fenomena ini merupakan salah satu tantangan utama pada deteksi produk FMCG karena banyak produk memiliki bentuk kemasan yang serupa, sementara perbedaannya hanya terletak pada detail visual kecil seperti warna, logo, atau varian produk. Akibatnya, model relatif mudah menentukan lokasi objek, tetapi masih mengalami kesulitan dalam membedakan antar kelas yang memiliki kemiripan visual tinggi.

Lebih lanjut, dalam konteks penggunaan metode *Semi-Supervised Learning* berbasis *Noisy Student* pada skenario L50, fenomena misklasifikasi atau salah klasifikasi ini juga dapat diindikasi sebagai akibat dari label *noise* yang muncul selama proses iterasi *pseudo-labeling*. Adanya keterbatasan jumlah 50% data berlabel murni (*ground truth* asli) berpotensi membuat model *Teacher* kurang percaya diri (*low confidence*) dalam membedakan fitur *fine-grained* dari varian spesifik tersebut, sehingga menghasilkan pseudo-label yang keliru ("47_other_variant") pada data *unlabeled*. Kesalahan pelabelan semu ini kemudian dipelajari secara berulang oleh model *Student*. Temuan ini secara analitis menegaskan bahwa, meskipun

secara umum pendekatan L50 sangat efisien dalam mengekstraksi fitur spasial, kepekaan model dalam menangkap detail diskriminatif berskala mikro antar kelas produk ritel masih menjadi *bottleneck* (hambatan) teknis yang memerlukan perhatian khusus dalam riset *computer vision* di masa depan.



(a) Kasus Berhasil Deteksi (*True Positive*)



(b) Kasus Objek Terlewat (*False Negative*)



(c) Kasus Salah Klasifikasi (*False Positive*)

Gambar 4.9 Contoh Visual Hasil Deteksi Model *Student* L50 (Kiri = *Ground Truth*, Kanan = Prediksi Model)

Ketiga kasus di atas mengilustrasikan secara komprehensif konsekuensi praktis dari fluktuasi nilai *Precision* dan *Recall* dalam konteks penerapan model deteksi objek pada lingkungan ritel yang dinamis. Kasus *False Negative* pada Gambar 4.9(b) merupakan cerminan dari *Recall* yang rendah, yaitu kondisi ketika model gagal mengenali atau melewatkan objek yang secara aktual (*ground truth*) sebenarnya ada. Jika situasi ini terjadi pada sistem pemantauan stok otomatis di dunia nyata, produk yang sebenarnya masih tersedia di rak dapat tercatat sebagai *out-of-stock* (habis). Hal ini tidak hanya memicu keputusan *restock* (pengisian ulang) yang sebenarnya tidak diperlukan, tetapi juga berujung pada pemborosan sumber daya operasional, inefisiensi tenaga kerja, serta mengganggu optimasi alur rantai pasok (*supply chain*).

Sebaliknya, kasus *False Positive* pada Gambar 4.9(c) mencerminkan konsekuensi dari *Precision* yang rendah, yaitu kondisi ketika model salah mengenali kelas produk meskipun berhasil melokalisasi keberadaannya. Pada sistem inventaris otomatis, kesalahan ini menyebabkan laporan komposisi stok menjadi tidak akurat. Sebagai contoh, produk varian tertentu dapat tercatat memiliki volume lebih banyak dari jumlah sebenarnya, sementara varian lain tercatat kurang. Dalam skala operasional yang besar, anomali data ini dapat merusak akurasi sistem tata kelola gudang, menyebabkan kesalahan perencanaan pengadaan barang (*procurement*), hingga berpotensi merugikan konsumen jika terjadi selisih harga akibat kesalahan identifikasi sistem.

Secara keseluruhan, hasil analisis kualitatif menunjukkan bahwa skenario L50 mampu menghasilkan performa yang kompetitif dengan kebutuhan data berlabel yang lebih rendah dibandingkan pendekatan *fully supervised*. Kemampuan model *Student* L50 dalam melokalisasi objek secara sangat presisi, bahkan pada kondisi susunan rak yang padat, membuktikan bahwa penggunaan 50% data berlabel dari Dataset Menyeluruh telah memberikan informasi yang cukup bagi model untuk mempelajari karakteristik berbagai produk ritel.

Meskipun pada hasil pengujian model masih ditemukan kasus *False Negative* dan *False Positive* akibat faktor eksternal seperti tingkat oklusi (penghalangan antar objek), fluktuasi intensitas pencahayaan di lorong rak, dan tingginya tingkat kemiripan visual antar produk (*intra-class variance* pada barang *Fast-Moving Consumer Goods*), hal tersebut sejatinya merupakan tantangan fundamental dalam *computer vision*. Jenis-jenis kesalahan klasifikasi dan lokalisasi ini terbukti sulit diatasi hanya dengan strategi penambahan kuantitas anotasi manual secara besar. Oleh karena itu, pendekatan L50 ini terbukti berhasil menjadi titik

keseimbangan (*trade-off*) yang paling optimal antara efisiensi biaya serta waktu pelabelan dengan tingkat keandalan deteksi yang dibutuhkan untuk mendukung otomatisasi industri ritel.

4.5 Demo Aplikasi Visualisasi Sederhana Menggunakan Streamlit

Tahap akhir dari penelitian ini adalah merancang dan membangun sebuah aplikasi demonstrasi interaktif berbasis web menggunakan *framework* Streamlit. Aplikasi ini dikembangkan untuk mengintegrasikan seluruh model yang telah dilatih serta menyajikan visualisasi data eksperimen secara terpadu. Tujuan utamanya adalah untuk memudahkan pengujian visual secara *real-time* sekaligus memfasilitasi analisis komparatif performa antar-skenario dan antar-dataset melalui antarmuka pengguna (*Graphical User Interface*) yang intuitif.

Gambar 4.10 menyajikan tampilan halaman utama dari aplikasi visualisasi sederhana. Aplikasi demo ini dirancang dengan struktur navigasi yang dinamis dan terpusat. Pada bagian atas antarmuka, terdapat dua parameter kontrol utama, yaitu tombol pilihan (*radio button*) untuk menentukan "Sumber Dataset" (Dataset Parsial atau Dataset Menyeluruh) serta panel geser (*slider*) untuk menentukan "Rasio Data Berlabel (%)". Interaksi pengguna pada kedua kontrol ini akan secara otomatis memperbarui data dan model yang digunakan pada ketiga *tab* fungsional di bawahnya yaitu Demo Inferensi Gambar, Grafik Metrik Pelatihan, dan Perbandingan Semua Skenario.

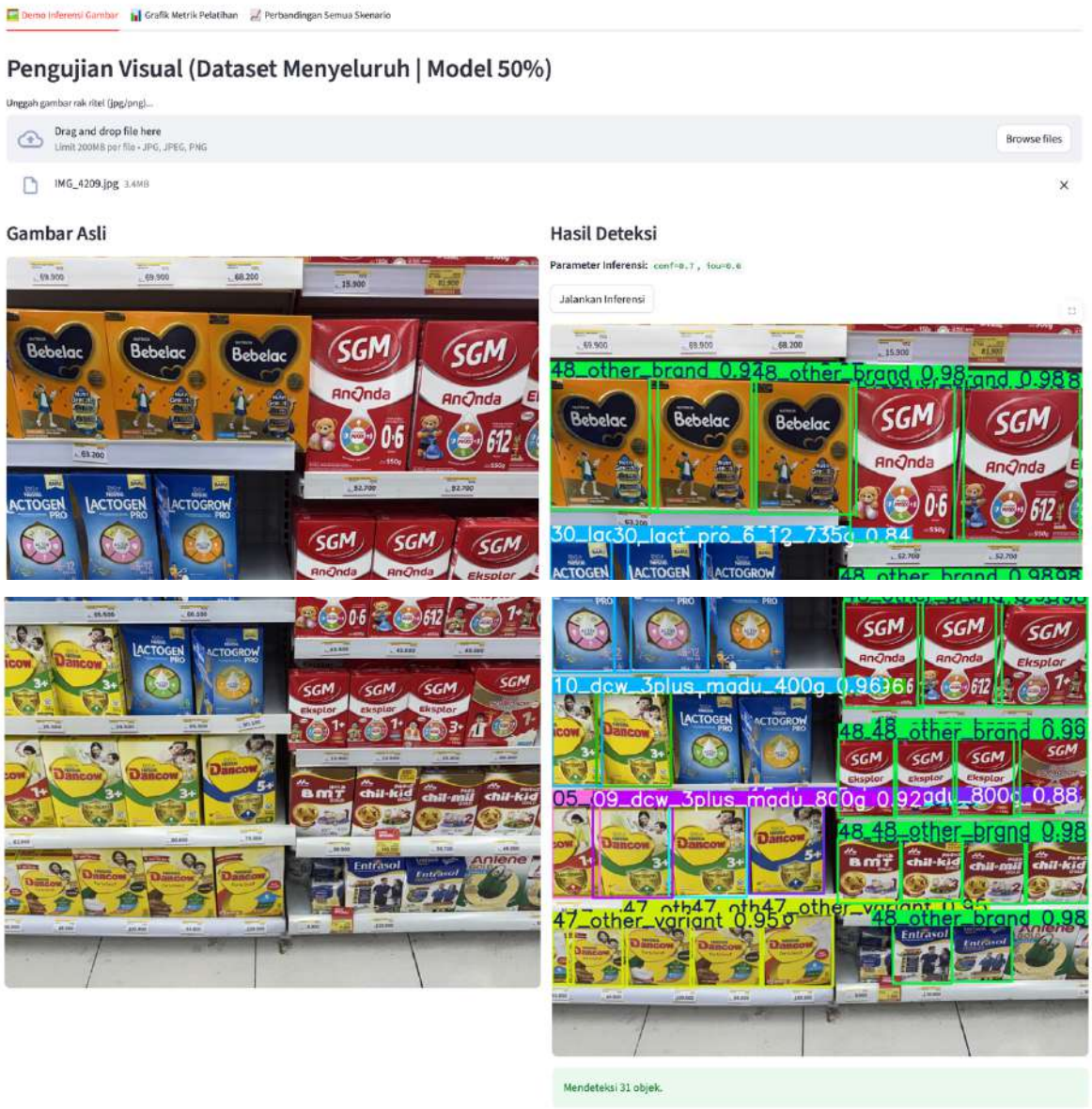


Gambar 4.10 Tampilan Halaman Utama Aplikasi

4.5.1 Demo Inferensi Gambar

Tab pertama pada aplikasi demo adalah "Demo Inferensi Gambar", yang dirancang khusus untuk memfasilitasi pengujian visual secara langsung (*real-time*) terhadap model YOLOv12 yang telah dilatih. Tampilan antarmuka awal pengujian visual ini disajikan pada Gambar 4.11. Pada halaman ini, pengguna disediakan area pengunggahan berkas (*file uploader*) yang mendukung format citra standar seperti JPG, JPEG, dan PNG. Setelah pengguna mengunggah citra rak produk ritel, gambar tersebut bisa dilihat langsung pada kolom "Gambar Asli". Tepat di sebelahnya, terdapat tombol "Jalankan Inferensi" yang siap dieksekusi untuk memulai proses deteksi. Di atas tombol "Jalankan Inferensi" terdapat parameter inferensi berupa *confidence threshold* sebesar 0,7 dan juga IoU *threshold* sebesar 0,6. Ketika pengguna menekan tombol "Jalankan Inferensi", sistem secara dinamis akan memuat bobot model YOLOv12 yang bersesuaian dengan pengaturan "Sumber Dataset" dan "Rasio Data Berlabel" yang dipilih di kontrol utama (sebagai contoh, menggunakan model Dataset Menyeluruh pada rasio 50%). Sistem kemudian memproses citra tersebut dan memunculkan *output*-nya pada kolom "Hasil Deteksi".

Gambar 4.11 memperlihatkan hasil visualisasi yang menampilkan gambar produk di mana objek yang dikenali telah ditandai dengan kotak pembatas (*bounding box*). Setiap *bounding box* dilengkapi dengan informasi detail berupa label nama kelas produk beserta skor kepercayaannya (*confidence score*). Sebagai tambahan, aplikasi juga memberikan notifikasi ringkas di bagian bawah berupa kotak berwarna hijau yang merangkum total objek yang berhasil dikenali oleh model pada citra tersebut.

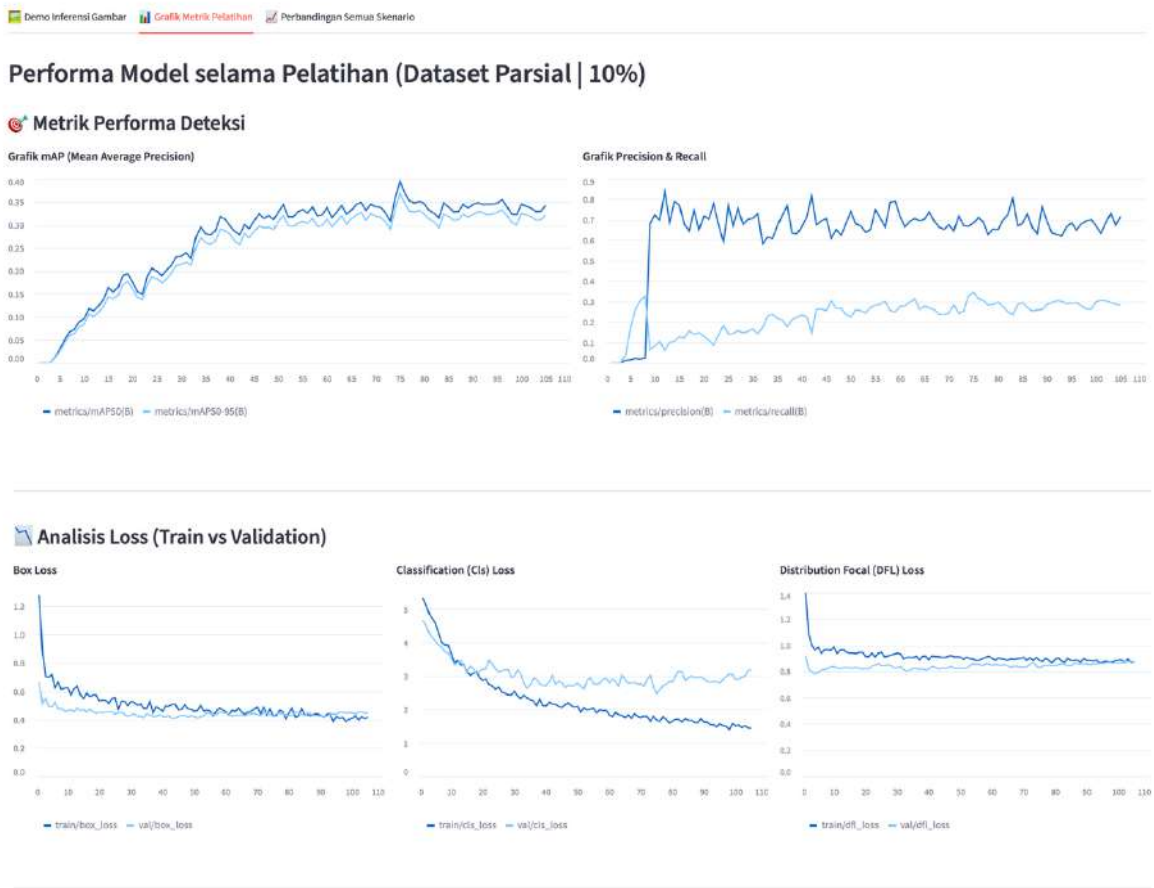


Gambar 4.11 Tampilan Halaman *Tab* Pertama Demo Inferensi Gambar

4.5.2 Grafik Metrik Pelatihan

Pada *tab* kedua, yaitu "Grafik Metrik Pelatihan", aplikasi demo menyediakan fasilitas bagi pengguna untuk meninjau historis pembelajaran model secara mendetail. Halaman ini dirancang untuk membedah evaluasi metrik performa selama proses pelatihan model berlangsung. Berbeda dengan *tab* sebelumnya yang berfokus pada hasil akhir inferensi, tab ini memvisualisasikan bagaimana model belajar dan berkembang, yang mencakup kurva *mean Average Precision* (mAP), *Precision*, *Recall*, serta grafik penurunan berbagai jenis fungsi kerugian (*Loss*).

Tampilan antarmuka untuk modul metrik performa dan analisis *loss* ini disajikan pada Gambar 4.12. Antarmuka halaman secara dinamis menyesuaikan judul laporannya berdasarkan parameter yang dipilih oleh pengguna di halaman utama (sebagai contoh, menampilkan data "Dataset Parsial | 10%"). Pada bagian "Metrik Performa Deteksi", aplikasi menyajikan dua grafik utama. Grafik pertama memantau pergerakan nilai mAP@50 dan mAP@50-95, sementara grafik kedua melacak fluktuasi *Precision* dan *Recall* di sepanjang *epoch* pelatihan. Selanjutnya, aplikasi juga menyajikan modul "Analisis Loss (Train vs Validation)" yang menampilkan tiga grafik fungsi kerugian khas arsitektur YOLOv12, yaitu *Box Loss*, *Classification (Cls) Loss*, dan *Distribution Focal (DFL) Loss*. Grafik ini secara langsung mengomparasikan nilai *loss* pada fase pelatihan (*train*) dan fase validasi (*val*), yang berfungsi sebagai instrumen diagnostik visual untuk mendeteksi gejala *overfitting* atau ketidakstabilan pembelajaran.



Gambar 4.12 Tampilan Halaman *Tab* Kedua Grafik Metrik Pelatihan

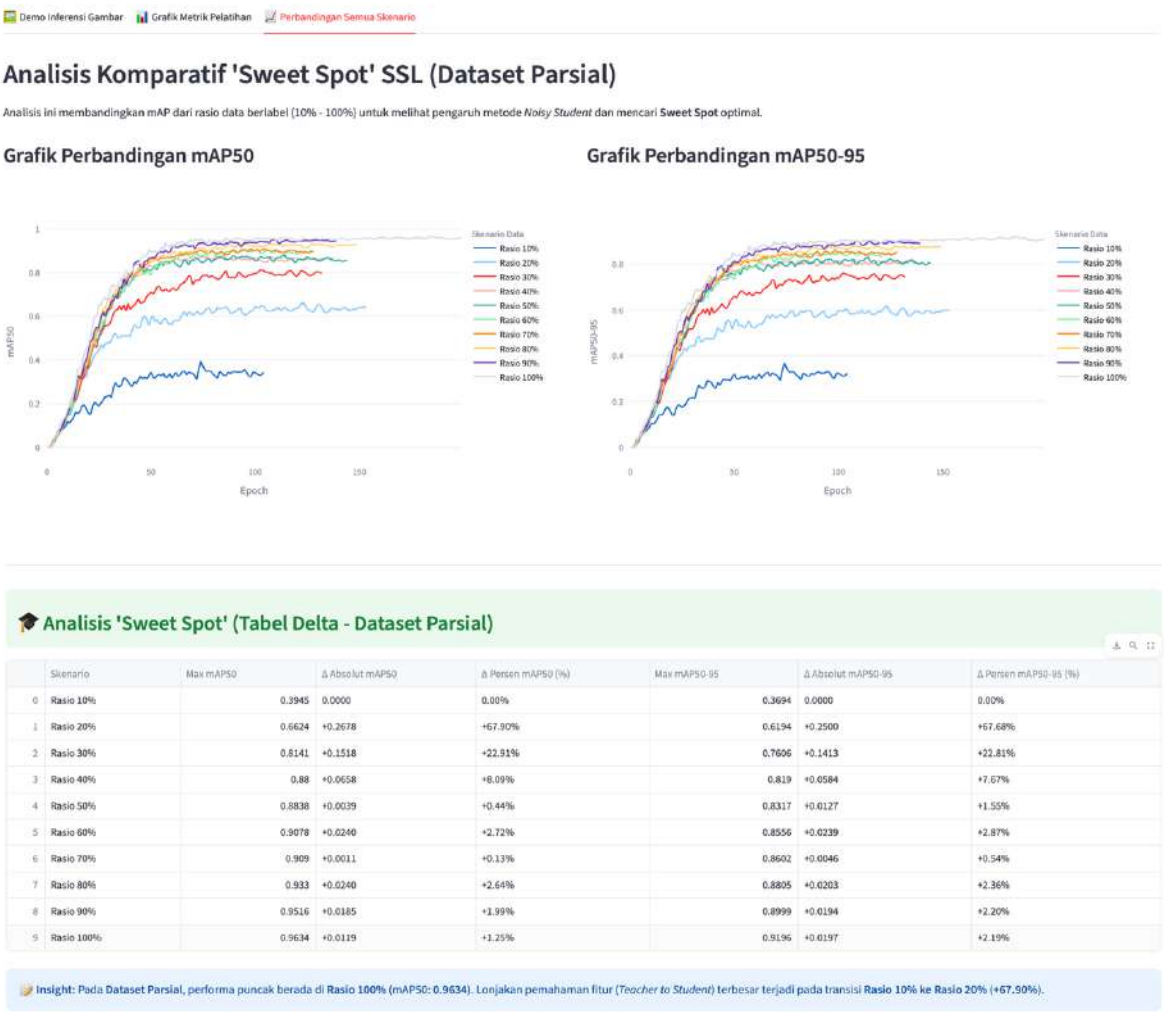
4.5.3 Perbandingan Semua Skenario

Tab ketiga pada aplikasi demo adalah "Perbandingan Semua Skenario". Halaman ini dirancang sebagai pusat analisis komprehensif yang merangkum keseluruhan hasil eksperimen *Semi-Supervised Learning* (SSL). Tampilan halaman *tab* ketiga ini dapat dilihat pada Gambar 4.13. Jika pada tab sebelumnya pengguna hanya dapat melihat performa satu skenario secara individual, *tab* ini membuat pengguna untuk melakukan komparasi performa dari seluruh rentang rasio data berlabel (mulai dari 10% hingga 100%) secara bersamaan. Tujuan utama dari halaman ini adalah untuk mempermudah identifikasi titik *sweet spot* yang paling optimal secara visual dan matematis.

Berdasarkan Gambar 4.13, aplikasi menyajikan visualisasi berupa grafik *multiline* untuk "Grafik Perbandingan mAP50" dan "Grafik Perbandingan mAP50-95". Dengan menampilkan seluruh kurva dalam satu grafik, pengguna dapat mengamati perkembangan performa model seiring meningkatnya proporsi data berlabel. Visualisasi ini juga memudahkan identifikasi perbedaan performa antar skenario, terutama pada rasio pelabelan rendah seperti 10% dan 20%

yang masih menunjukkan kesenjangan cukup besar. Seiring bertambahnya jumlah data berlabel, jarak antar kurva semakin mengecil hingga akhirnya cenderung konvergen pada rasio menengah dan tinggi.

Selain menyajikan visualisasi grafik, aplikasi juga menyediakan modul "Analisis Sweet Spot (Tabel Delta)". Sistem secara otomatis mengekstraksi nilai maksimum mAP@50 dan mAP@50-95 dari masing-masing skenario, lalu menghitung selisih (delta) absolut dan persentase peningkatannya terhadap skenario sebelumnya. Aplikasi juga menyediakan kotak "Insight" yang menyimpulkan temuan utama dari tabel tersebut. Sebagai contoh, sistem secara otomatis mendeteksi dan melaporkan bahwa lonjakan pemahaman fitur (*Teacher to Student*) terbesar terjadi pada transisi rasio 10% ke 20% dengan peningkatan sebesar +67,90%. Fitur ini membantu mempercepat proses analisis hasil eksperimen sekaligus memudahkan pengguna dalam mengidentifikasi rasio pelabelan yang memberikan peningkatan performa paling signifikan.



	Skenario	Max mAP50	Δ Absolut mAP50	Δ Persen mAP50 (%)	Max mAP50-95	Δ Absolut mAP50-95	Δ Persen mAP50-95 (%)
0	Rasio 10%	0.3945	0.0000	0.00%	0.3694	0.0000	0.00%
1	Rasio 20%	0.6624	+0.2678	+67.90%	0.6194	+0.2500	+67.68%
2	Rasio 30%	0.8141	+0.1518	+22.91%	0.7606	+0.1413	+22.81%
3	Rasio 40%	0.88	+0.0658	+8.09%	0.819	+0.0584	+7.67%
4	Rasio 50%	0.8838	+0.0039	+0.44%	0.8317	+0.0127	+1.55%
5	Rasio 60%	0.9078	+0.0240	+2.72%	0.8556	+0.0239	+2.87%
6	Rasio 70%	0.909	+0.0011	+0.13%	0.8602	+0.0046	+0.54%
7	Rasio 80%	0.933	+0.0240	+2.64%	0.8805	+0.0203	+2.36%
8	Rasio 90%	0.9516	+0.0185	+1.99%	0.8999	+0.0194	+2.20%
9	Rasio 100%	0.9634	+0.0119	+1.25%	0.9196	+0.0197	+2.19%

Gambar 4.13 Tampilan Halaman *Tab* Ketiga Perbandingan Semua Skenario

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan seluruh rangkaian penelitian, analisis performa model, dan pembahasan mengenai penerapan metode *Semi-Supervised Learning* (SSL) menggunakan algoritma *Noisy Student* pada arsitektur YOLOv12 untuk deteksi produk ritel, dapat ditarik beberapa kesimpulan sebagai berikut:

a. Penelitian ini berhasil menerapkan skema pelatihan *Noisy Student* dengan model dasar YOLOv12-Nano untuk mendeteksi 47-49 kelas produk susu bubuk kemasan kardus. *Pipeline* SSL yang dirancang mencakup tahap pelatihan model *Teacher* secara *supervised*, proses *pseudo-labeling* menggunakan *confidence threshold* 0,70 dan IoU *threshold* 0,60, serta pelatihan model *Student* dengan augmentasi agresif sebagai *noise* terbukti berjalan secara fungsional dan menghasilkan model deteksi yang valid pada seluruh sepuluh skenario rasio yang diujikan.

b. Skenario L50 (50% data berlabel : 50% data tidak berlabel) secara konsisten teridentifikasi sebagai titik *sweet spot* atau rasio paling optimal pada kedua varian dataset. Pada skenario ini, model *Student* mampu mencapai nilai mAP@50 sebesar 0,8612 pada Dataset Menyeluruh dan 0,8544 pada Dataset Parsial. Pencapaian tersebut setara dengan 97,2% dari performa *baseline fully supervised* L100 (mAP@50 = 0,8862) pada Dataset Menyeluruh, dan 89,9% dari performa L100 (mAP@50 = 0,9501) pada Dataset Parsial. Temuan ini membuktikan bahwa pendekatan SSL mampu memangkas setengah beban anotasi manual tanpa mengorbankan reliabilitas performa deteksi secara signifikan. Secara praktis, jika diasumsikan proses anotasi manual membutuhkan waktu rata-rata satu menit/citra, maka pada skala penelitian ini yang memuat 328 citra latih, penerapan rasio 50% berhasil menghemat waktu pelabelan manual sebanyak 164 menit atau sekitar 2,7 jam. Lebih jauh, jika diilustrasikan pada konteks implementasi industri ritel dengan volume data yang besar, misalnya 10.000 citra, penerapan efisiensi 50% ini akan memungkinkan perusahaan untuk memangkas waktu anotasi sebanyak 5.000 menit, atau menghemat sekitar 10 hari kerja operasional. Hal ini menegaskan bahwa titik *sweet spot* L50 tidak hanya mempertahankan kekuatan akurasi model, tetapi juga memberikan dampak finansial dan efisiensi waktu yang signifikan di dunia nyata.

c. Analisis delta mAP@50 antar-skenario mengungkap tiga pola utama dalam kurva efisiensi model. Pertama, terjadi lonjakan performa yang sangat dramatis pada transisi L10 ke L20. Dataset Parsial mencatat kenaikan +64,2% dan Dataset Menyeluruh +147,3% yang disebabkan oleh kondisi *Teacher* yang hampir tidak dapat belajar akibat rata-rata kurang dari satu gambar per kelas pada skenario L10. Kedua, terdapat *tipping point* sekunder pada transisi L40 ke L50, di mana Dataset Menyeluruh mencatat lonjakan mAP@50 sebesar +15,7%, menandakan bahwa 40% data berlabel merupakan ambang kecukupan representasi visual bagi model *Teacher*. Ketiga, terbentuk fase *plateau* (landai) yang konsisten setelah L60, di mana setiap penambahan 10% data berlabel hanya memberikan peningkatan mAP marginal di bawah 3%, mempertegas berlakunya hukum *diminishing returns* dalam proses anotasi data.

d. Kualitas anotasi dataset terbukti memberikan pengaruh yang signifikan terhadap performa SSL, khususnya pada skenario data berlabel yang sangat terbatas. Dataset Menyeluruh yang memiliki total 5665 label (vs. 2614 label pada Dataset Parsial) mampu menghasilkan *pseudo-label* yang jauh lebih padat dan representatif per gambar. Meskipun demikian, pada skenario L10 hingga L40, Dataset Parsial secara paradoks menghasilkan mAP yang lebih tinggi karena gaya *bounding box* yang lebih longgar sehingga lebih mudah melampaui IoU *threshold* saat evaluasi. Fenomena ini menunjukkan bahwa standar anotasi berinteraksi secara kompleks dengan konfigurasi *threshold* pada *pipeline* SSL.

e. Dari sudut pandang praktis dan ekonomi, penelitian ini memberikan rekomendasi konkret bagi industri ritel yaitu investasi awal dalam pelabelan setidaknya 50% dari total data yang direncanakan merupakan langkah paling *cost-effective*. Strategi ini memungkinkan pengelola ritel untuk memanfaatkan melimpahnya citra produk tanpa label guna membangun sistem deteksi objek berkinerja tinggi, tanpa harus menanggung keseluruhan beban biaya dan waktu anotasi manual yang selama ini menjadi *bottleneck* utama dalam pengembangan sistem *Computer Vision* berbasis *Deep Learning*.

5.2 Limitasi

Meskipun penelitian ini menghasilkan beberapa temuan yang relevan, terdapat beberapa keterbatasan yang perlu diakui dan menjadi catatan penting dalam menginterpretasikan hasil penelitian, antara lain:

a. Cakupan dataset dalam penelitian ini terbatas hanya pada 408 citra produk susu bubuk kemasan kardus dengan 47-49 kelas produk. Oleh karena itu, temuan mengenai rasio

pelabelan optimal (*sweet spot*) pada skenario L50 belum tentu dapat digeneralisasikan secara langsung ke kategori produk *Fast-Moving Consumer Goods* (FMCG) lainnya. Perbedaan karakteristik visual, seperti produk dengan kemasan fleksibel, produk minuman botol, atau produk dengan jumlah kelas yang jauh lebih banyak atau lebih sedikit, berpotensi menghasilkan karakteristik pembelajaran dan rasio pelabelan optimal yang berbeda.

b. Proses pelabelan dataset *ground truth* pada Dataset Menyeluruh dilakukan secara manual pada 408 citra ritel yang mencakup 49 kelas objek. Meskipun proses pelabelan telah dilakukan secara bertahap, penelitian ini memiliki limitasi berupa potensi label *noise* atau ketidakakuratan akibat *human error* (seperti kesalahan visual saat penarikan *bounding box* pada objek yang saling bertumpuk/terpotong atau kesalahan pelabelan produk yang ada di dalam citra).
c. Penelitian ini hanya mengimplementasikan satu iterasi *Noisy Student*. Implementasi multi-iterasi, di mana model *Student* dari satu siklus digunakan sebagai *Teacher* baru pada siklus berikutnya, berpotensi meningkatkan performa secara signifikan pada skenario data berlabel rendah, namun tidak dieksplorasi dalam penelitian ini karena keterbatasan waktu dan sumber daya komputasi.
d. Penelitian dilakukan menggunakan satu varian arsitektur YOLOv12 saja, yaitu YOLOv12-Nano. Penggunaan varian yang lebih besar seperti YOLOv12-Small atau YOLOv12-Medium sebagai model *Teacher* dan *Student* berpotensi menghasilkan *pseudo-label* yang lebih berkualitas, terutama pada skenario data berlabel rendah, namun konsekuensinya adalah peningkatan kebutuhan memori GPU dan waktu komputasi yang signifikan di luar kapasitas perangkat keras yang tersedia.
e. Penelitian ini tidak mengeksplorasi variasi konfigurasi *confidence threshold* dan IoU *threshold* pada proses *pseudo-labeling*. Nilai tetap yang digunakan (0,70 dan 0,60) merupakan nilai konservatif yang dipilih untuk memastikan kualitas *pseudo-label*. Variasi nilai *threshold* tersebut berpotensi memengaruhi rasio *sweet spot* secara berbeda dan belum menjadi cakupan dalam penelitian ini.

5.3 Saran

Berdasarkan temuan penelitian dan limitasi yang telah diidentifikasi, berikut adalah beberapa saran yang dapat menjadi acuan bagi pengembangan penelitian selanjutnya:

a. Untuk penelitian selanjutnya, disarankan untuk menerapkan skema verifikasi ganda (*Cross-Validation Annotation*) pada proses pelabelan data (*Data Annotation*) melalui skema *multi-annotator* atau pengecekan silang oleh lebih dari satu orang. Pendekatan ini dapat meminimalisir potensi *human error* dan meningkatkan konsistensi *bounding box* yang dihasilkan.
b. Eksplorasi algoritma *Semi-Supervised Learning* selain *Noisy Student*. Penelitian lanjutan dapat membandingkan atau menggunakan algoritma SSL lain untuk mendapatkan rasio pelabelan yang optimal menggunakan dataset dan juga rasio yang identik.
c. Penelitian lanjutan dapat mengeksplorasi penerapan *Noisy Student* secara iteratif untuk mengatasi keterbatasan yang muncul dari penggunaan satu siklus *Teacher-Student* saja. Dalam pendekatan ini, model *Student* yang sudah terlatih dari siklus pertama digunakan kembali sebagai *Teacher* baru untuk siklus berikutnya. Proses yang berulang ini diharapkan mampu secara bertahap memperbaiki kualitas *pseudo-label*, sehingga performa yang setara mungkin dapat dicapai bahkan dengan proporsi data berlabel yang lebih kecil dari 50%.
d. Penelitian selanjutnya dapat menggunakan arsitektur YOLOv12 yang lebih besar (Small/Medium) untuk menghasilkan *pseudo-label* yang lebih berkualitas lagi. Kapasitas ekstraksi fitur yang lebih tinggi pada arsitektur yang lebih besar berpotensi menghasilkan *pseudo-label* dengan tingkat akurasi dan kepercayaan (*confidence score*) yang lebih presisi pada *unlabeled* data, sehingga dapat meminimalisir penumpukan kesalahan (*error propagation*) selama proses *Semi-Supervised Learning*.
e. Mengingat keterbatasan perangkat keras yang dialami, penelitian selanjutnya juga disarankan untuk memanfaatkan fasilitas komputasi berbasis *cloud* yang menyediakan akses ke GPU dengan kapasitas lebih besar. Infrastruktur yang lebih memadai akan membuka kemungkinan untuk menguji konfigurasi *Teacher* yang lebih bertenaga, misalnya menggunakan varian YOLOv12 yang lebih besar, sehingga kualitas *pseudo-label* pada skenario data terbatas dapat ditingkatkan tanpa terhambat oleh keterbatasan memori.
f. Untuk mendapatkan kualitas *pseudo-label* yang optimal pada skenario data berlabel rendah, disarankan agar penelitian selanjutnya melakukan eksplorasi dan optimasi sistematis terhadap nilai *confidence threshold* dan IoU *threshold* pada proses *pseudo-labeling*. Pendekatan *adaptive thresholding* di mana nilai *threshold* disesuaikan secara dinamis berdasarkan distribusi skor kepercayaan *Teacher* per kelas dapat menjadi strategi

yang menjanjikan untuk meningkatkan kualitas *pseudo-label* pada skenario yang memiliki ketidakseimbangan kelas yang parah seperti L10.

g. Bagi praktisi industri ritel yang ingin mengadopsi temuan penelitian ini, sangat disarankan untuk memulai dengan melakukan anotasi manual berkualitas tinggi pada 50% dari total dataset yang direncanakan dengan memperhatikan konsistensi gaya *bounding box* yang presisi dan inklusif (mencakup seluruh objek yang relevan dalam citra). Standar anotasi yang konsisten terbukti dalam penelitian ini menjadi faktor penentu yang krusial dalam memaksimalkan efektivitas *pipeline Noisy Student*, khususnya pada skenario dengan data berlabel menengah (L50 ke atas).

DAFTAR PUSTAKA

Buleu, B., Robu, R., & Filip, I. (2025). A Deep Learning-Based System for Automatic License Plate Recognition Using YOLOv12 and PaddleOCR. *Applied Sciences*, *15*(14), 7833. https://doi.org/10.3390/app15147833

C A Padmanabha Reddy, Y., Viswanath, P., & Eswara Reddy, B. (2018). Semi-supervised learning: A brief review. *International Journal of Engineering & Technology*, *7*(1.8), 81. https://doi.org/10.14419/ijet.v7i1.8.9977

Chauhan, J., Varadarajan, S., & Srivastava, M. M. (2021). *Semi-supervised Learning for Dense Object Detection in Retail Scenes* (arXiv:2107.02114). arXiv. https://doi.org/10.48550/arXiv.2107.02114

Hou, P., & Huang, S. (2025). BCSM-YOLO: An Improved Product Package Recognition Algorithm for Automated Retail Stores Based on YOLOv11. *IEEE Access*, *13*, 139665–139679. https://doi.org/10.1109/ACCESS.2025.3595175

Jun, Y. Y., Poh, T. R., Wong, K., Mooi, L. S., & Eun, H. K. (2025a). Optimizing Online Retail: Uncovering Opportunities with Unsupervised Learning Techniques. *JOIV : International Journal on Informatics Visualization*, *9*(5), 2220–2230. https://doi.org/10.62527/joiv.9.5.3082

Lachhab, W. (2023). *Deep Learning for Efficient Retail Shelf Stock Monitoring and Analysis*. https://aaltodoc.aalto.fi/handle/123456789/124070

Putra, G. A. A., & Wowor, A. D. (2025). Analisis Perbandingan Performa YOLO v11 Dan v12 menggunakan model N dan S. *Innovative: Journal Of Social Science Research*, *5*(4), 8593–8601. https://doi.org/10.31004/innovative.v5i4.21004

Sapkota, R., Meng, Z., Churuvija, M., Du, X., Ma, Z., & Karkee, M. (2025). *Comprehensive Performance Evaluation of YOLOv12, YOLO11, YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments* (arXiv:2407.12040). arXiv. https://doi.org/10.48550/arXiv.2407.12040

Sohn, K., Zhang, Z., Li, C.-L., Zhang, H., Lee, C.-Y., & Pfister, T. (2020). *A Simple Semi-Supervised Learning Framework for Object Detection* (arXiv:2005.04757). arXiv. https://doi.org/10.48550/arXiv.2005.04757

Song, J., Park, D., Jeong, D., & Kang, B. (2024). *Weakly Supervised Learning-Based Product Detection for Offline Retail Store* (SSRN Scholarly Paper No. 4719904). Social Science Research Network. https://doi.org/10.2139/ssrn.4719904

Tian, Y., Ye, Q., & Doermann, D. (2025). *YOLOv12: Attention-Centric Real-Time Object Detectors* (arXiv:2502.12524). arXiv. https://doi.org/10.48550/arXiv.2502.12524

Varadarajan, S., & Srivastava, M. M. (2020). Weakly Supervised Object Localization on grocery shelves using simple FCN and Synthetic Dataset. *Proceedings of the 11th Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP '18*, 1–7. https://doi.org/10.1145/3293353.3293367

Wang, X., Girdhar, R., Yu, S. X., & Misra, I. (2023). *Cut and Learn for Unsupervised Object Detection and Instance Segmentation*. 3124–3134. https://openaccess.thecvf.com/content/CVPR2023/html/Wang_Cut_and_Learn_for_Unsupervised_Object_Detection_and_Instance_Segmentation_CVPR_2023_paper.html

Wibowo, S. A., & Corputty, F. (2024). *Retail Product Recognition: A Combination of Yolov8 and Modified Self-Supervised Learning* (SSRN Scholarly Paper No. 4890491). Social Science Research Network. https://doi.org/10.2139/ssrn.4890491

Xie, E., Ding, J., Wang, W., Zhan, X., Xu, H., Sun, P., Li, Z., & Luo, P. (2021). *DetCo: Unsupervised Contrastive Learning for Object Detection*. 8392–8401. https://openaccess.thecvf.com/content/ICCV2021/html/Xie_DetCo_Unsupervised_Contrastive_Learning_for_Object_Detection_ICCV_2021_paper.html

Yazdanjouei, H., Mansouri, A., & Shokouhifar, M. (2025). *A Co-Training Semi-Supervised Framework Using Faster R-CNN and YOLO Networks for Object Detection in Densely Packed Retail Images* (arXiv:2509.09750). arXiv. https://doi.org/10.48550/arXiv.2509.09750

Zangana, H. (2025). Object Detection and Visual Intelligence in Retail Environments: A Deep Learning Approach for Inventory and Behavior Analytics. *Intelligent Methods In Engineering Sciences*, *4*(2), 22–28. https://doi.org/10.58190/imiens.2025.124

LAMPIRAN