

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Tabel 2.1 Penelitian Terdahulu

Penelitian	Metode	Gambar /Data	Tipe Penelitian	Focus Utama	Keunggulan	Keterbatasan/ Kelemahan
Nofryanti dkk. (2023)	YOLO v5	694 citra digital	Deteksi objek	Mendeteksi penggunaan <i>safety helmet</i>	Memperoleh precision 0,95 dan recall 0,93 serta mampu mengenali penggunaan helm dengan baik	Masih berfokus pada citra digital dan belum menerapkan prototype <i>real-time</i>
Widodo dkk. (2021)	CNN dan YOLO	Citra pekerja konstruksi	Deteksi objek	Mendeteksi pemakaian helm proyek	Mampu menerapkan metode <i>deep learning</i> untuk mengenali objek helm	pengujian masih menggunakan citra statis dan belum diterapkan secara <i>real-time</i>
Reswara dkk. (2025)	YOLO v10	Ratusan citra	Deteksi objek	Mendeteksi penggunaan helm	Menggunakan YOLO versi terbaru dan menghasilkan performa deteksi yang baik berdasarkan precision dan mAP	Belum menekankan implementasi prototype sederhana secara <i>real-time</i>
Harahap (2024)	YOLO	Citra digital	Deteksi objek	Mendeteksi objek menggunakan model YOLO	Memiliki kecepatan dan akurasi deteksi yang baik	Masih berfokus pada pengolahan citra digital dan belum diterapkan

Penelitian	Metode	Gambar /Data	Tipe Penelitian	Focus Utama	Keunggulan	Keterbatasan/ Kelemahan
						secara langsung di lapangan
Nasution dkk. (2025)	YOLO v5 dan <i>Deep Learning</i>	Citra digital	Perbandingan metode	Membandingkan metode deteksi objek	Menunjukkan bahwa <i>deep learning</i> memiliki performa baik dalam akurasi dan kecepatan deteksi	Belum secara khusus mengembangkan sistem deteksi helm dan belum menerapkan deteksi <i>real-time</i>

2.2 Pengolahan Citra Digital

1. Computer vision

Computer Vision merupakan salah satu cabang ilmu kecerdasan buatan (Artificial Intelligence/AI) yang bertujuan untuk memberikan kemampuan kepada komputer agar dapat melihat, mengenali, memahami, dan menganalisis informasi yang terdapat pada citra maupun video, sebagaimana kemampuan penglihatan manusia. Melalui teknologi ini, komputer dapat mengekstraksi informasi penting dari suatu objek sehingga mampu melakukan berbagai tugas, seperti klasifikasi, deteksi objek, segmentasi citra, pelacakan objek (object tracking), hingga pengenalan wajah (face recognition).

Pada dasarnya, Computer Vision bekerja dengan mengolah data visual yang diperoleh dari kamera atau perangkat pencitraan

lainnya. Data tersebut kemudian diproses menggunakan algoritma pengolahan citra (image processing) dan metode machine learning maupun deep learning untuk mengenali pola atau karakteristik tertentu dari objek yang terdapat pada gambar atau video. Perkembangan teknologi deep learning, khususnya penggunaan Convolutional Neural Network (CNN), telah meningkatkan kemampuan Computer Vision dalam menghasilkan deteksi objek yang lebih cepat dan akurat dibandingkan metode konvensional.

Dalam penerapannya, Computer Vision memiliki berbagai tahapan, yaitu proses akuisisi citra, pre-processing, ekstraksi fitur, analisis menggunakan model kecerdasan buatan, dan penyajian hasil deteksi. Tahapan pertama adalah akuisisi citra, yaitu proses memperoleh gambar atau video dari kamera. Selanjutnya dilakukan pre-processing untuk meningkatkan kualitas citra, seperti pengaturan ukuran gambar, pengurangan noise, atau normalisasi warna. Setelah itu, sistem melakukan ekstraksi fitur dan analisis menggunakan model deep learning sehingga objek dapat dikenali dan diklasifikasikan sesuai dengan kategori yang telah dipelajari sebelumnya.

Computer Vision telah banyak diterapkan pada berbagai bidang, seperti sistem keamanan, kendaraan otonom (autonomous vehicle), kesehatan, industri manufaktur, pertanian, pengawasan lalu lintas, hingga sistem Keselamatan dan Kesehatan Kerja (K3). Pada

bidang konstruksi, teknologi ini dimanfaatkan untuk mendeteksi penggunaan alat pelindung diri (APD), seperti helm proyek (safety helmet), rompi keselamatan (safety vest), maupun perlengkapan keselamatan lainnya secara otomatis dan real-time. Dengan adanya sistem tersebut, proses pengawasan terhadap kepatuhan pekerja menjadi lebih efektif dan efisien dibandingkan dengan pengawasan secara manual. (Putra & Mulyana, 2024)

Pada penelitian ini, Computer Vision digunakan sebagai teknologi utama untuk mendeteksi penggunaan helm proyek (safety helmet) menggunakan algoritma YOLOv8. Kamera laptop berfungsi sebagai perangkat untuk menangkap citra secara real-time, kemudian setiap frame video diproses oleh model YOLOv8 yang telah dilatih sebelumnya. Sistem akan mengidentifikasi apakah pekerja menggunakan helm atau tidak menggunakan helm, kemudian menampilkan hasil deteksi dalam bentuk bounding box, label objek, dan nilai confidence. Dengan demikian, Computer Vision berperan sebagai dasar dalam membangun sistem pengawasan K3 yang mampu bekerja secara otomatis, cepat, dan akurat. (Talib et al., 2024)

2. Deteksi otomatis

Deteksi otomatis (automatic detection) merupakan suatu teknik yang memungkinkan sistem komputer mengenali, menemukan, dan mengidentifikasi objek atau kondisi tertentu secara

otomatis tanpa memerlukan campur tangan manusia. Teknologi ini memanfaatkan algoritma pengolahan citra (image processing), machine learning, dan deep learning untuk menganalisis data visual yang diperoleh dari kamera atau perangkat pencitraan lainnya. Tujuan utama deteksi otomatis adalah meningkatkan kecepatan, ketepatan, dan efisiensi proses identifikasi objek dibandingkan dengan metode pengamatan secara manual.

Dalam bidang Computer Vision, deteksi otomatis bekerja dengan cara menganalisis setiap citra atau frame video yang masuk ke dalam sistem. Selanjutnya, model yang telah dilatih (trained model) akan mencari pola atau karakteristik tertentu yang sesuai dengan objek target. Setelah objek berhasil dikenali, sistem akan memberikan informasi mengenai lokasi objek dalam bentuk bounding box, nama kelas (class label), dan tingkat kepercayaan (confidence score). Dengan demikian, sistem mampu melakukan identifikasi objek secara cepat dan konsisten meskipun jumlah data yang diproses sangat banyak. (Muhammad Azizul Mustofa & Bagus Satrio Waluyo Poetro, 2025)

Perkembangan teknologi deep learning, khususnya penggunaan Convolutional Neural Network (CNN) dan algoritma You Only Look Once (YOLO), telah meningkatkan kemampuan deteksi otomatis secara signifikan. Algoritma YOLO mampu melakukan proses deteksi dan klasifikasi objek hanya dalam satu

tahap (single-stage detector), sehingga memiliki kecepatan yang tinggi dan sangat sesuai untuk aplikasi real-time. Kemampuan tersebut menjadikan YOLO banyak digunakan dalam berbagai bidang, seperti sistem keamanan, pengawasan lalu lintas, industri manufaktur, kesehatan, hingga pengawasan Keselamatan dan Kesehatan Kerja (K3). (Triyanto et al., 2024)

Pada bidang konstruksi, deteksi otomatis dimanfaatkan untuk mengawasi kepatuhan pekerja dalam menggunakan Alat Pelindung Diri (APD), khususnya helm proyek (safety helmet). Sistem secara otomatis mendeteksi apakah pekerja menggunakan helm atau tidak menggunakan helm berdasarkan citra yang diperoleh dari kamera. Hasil deteksi kemudian ditampilkan secara langsung dalam bentuk kotak pembatas (bounding box), label objek, dan nilai confidence. Dengan adanya sistem ini, proses pengawasan menjadi lebih efektif karena tidak memerlukan pemeriksaan secara terus-menerus oleh petugas, sehingga dapat membantu meminimalkan risiko kecelakaan kerja akibat kelalaian penggunaan APD.

Pada penelitian ini, deteksi otomatis diterapkan menggunakan algoritma YOLOv8 yang telah dilatih dengan dataset penggunaan helm proyek. Kamera laptop digunakan sebagai media untuk memperoleh video secara real-time. Setiap frame video diproses oleh model YOLOv8 untuk mendeteksi keberadaan helm

pada pekerja konstruksi. Apabila objek helm terdeteksi, sistem akan memberikan label "Helmet", sedangkan apabila pekerja tidak menggunakan helm akan diberikan label "No Helmet". Hasil deteksi tersebut ditampilkan secara langsung pada layar sehingga dapat digunakan sebagai media pendukung pengawasan Keselamatan dan Kesehatan Kerja (K3) di area konstruksi.(Muhammad Azizul Mustofa & Bagus Satrio Waluyo Poetro, 2025)

2.3 Python

Python merupakan bahasa pemrograman tingkat tinggi (high-level programming language) yang dikembangkan oleh Guido van Rossum dan pertama kali dirilis pada tahun 1991. Python dirancang dengan tujuan untuk menghasilkan bahasa pemrograman yang sederhana, mudah dipelajari, dan memiliki sintaks yang mudah dibaca oleh manusia. Karena memiliki struktur kode yang ringkas dan jelas, Python menjadi salah satu bahasa pemrograman yang paling banyak digunakan dalam berbagai bidang, seperti pengembangan perangkat lunak, pengembangan web, analisis data, kecerdasan buatan (Artificial Intelligence), Machine Learning, Deep Learning, Computer Vision, Internet of Things (IoT), serta otomasi sistem.(Astuti, 2024)

Python bersifat open source, sehingga dapat digunakan, dimodifikasi, dan didistribusikan secara bebas. Selain itu, Python juga mendukung berbagai sistem operasi, seperti Windows, Linux, dan macOS. Salah satu keunggulan utama Python adalah tersedianya banyak pustaka

(library) dan kerangka kerja (framework) yang mempermudah pengembang dalam membangun berbagai jenis aplikasi tanpa harus membuat seluruh fungsi dari awal.

Pada bidang Artificial Intelligence dan Computer Vision, Python menjadi bahasa pemrograman yang paling populer karena memiliki dukungan pustaka yang sangat lengkap, seperti NumPy, Pandas, OpenCV, TensorFlow, PyTorch, Scikit-Learn, Keras, dan Ultralytics. Dengan bantuan pustaka tersebut, proses pengembangan sistem berbasis kecerdasan buatan dapat dilakukan dengan lebih cepat, mudah, dan efisien. (Ismail Setiawan, 2022)

2.3.1 Sejarah Python

Python diciptakan oleh Guido van Rossum pada akhir tahun 1980-an di Centrum Wiskunde & Informatica (CWI), Belanda. Pengembangan Python dimulai sebagai proyek pribadi yang bertujuan menciptakan bahasa pemrograman yang sederhana namun tetap memiliki kemampuan yang tinggi.

Nama Python tidak berasal dari nama ular, melainkan terinspirasi dari acara komedi Inggris berjudul Monty Python's Flying Circus. Guido van Rossum memilih nama tersebut karena menginginkan bahasa pemrograman yang terdengar unik, mudah diingat, dan tidak terlalu formal.

Sejak pertama kali dirilis pada tahun 1991, Python terus mengalami perkembangan. Saat ini Python telah menjadi salah satu

bahasa pemrograman paling populer di dunia dan banyak digunakan dalam dunia pendidikan, penelitian, maupun industri.

2.3.2 Karakteristik Python

Python memiliki beberapa karakteristik utama, yaitu:

1. High-Level Language

Python merupakan bahasa pemrograman tingkat tinggi sehingga sintaksnya mudah dipahami oleh manusia tanpa harus berhubungan langsung dengan bahasa mesin.

2. Interpreted Language

Python dijalankan menggunakan interpreter sehingga kode program dapat dieksekusi secara langsung tanpa melalui proses kompilasi seperti bahasa C atau C++.

3. Object-Oriented Programming (OOP)

Python mendukung konsep pemrograman berorientasi objek sehingga memudahkan pengembangan program yang bersifat modular dan terstruktur.

4. Cross Platform

Program Python dapat dijalankan pada berbagai sistem operasi tanpa perlu melakukan banyak perubahan pada kode program.

5. Open Source

Python dapat digunakan secara gratis dan memiliki komunitas pengembang yang sangat besar sehingga banyak tersedia dokumentasi, tutorial, dan pustaka pendukung.

2.3.3 Kelebihan Python

Python memiliki berbagai kelebihan yang menjadikannya banyak digunakan dalam penelitian dan pengembangan aplikasi, antara lain:

1. Sintaks sederhana sehingga mudah dipelajari.
2. Memiliki banyak pustaka (library) yang mendukung berbagai kebutuhan.
3. Bersifat gratis (free) dan open source.
4. Dapat dijalankan pada berbagai sistem operasi.
5. Memiliki komunitas pengguna yang sangat besar.
6. Mendukung pengembangan aplikasi berbasis Artificial Intelligence, Machine Learning, dan Deep Learning.
7. Mudah diintegrasikan dengan perangkat keras maupun perangkat lunak lainnya.
8. Cocok digunakan untuk membangun aplikasi real-time.

2.3.4 Kekurangan Python

Meskipun memiliki banyak kelebihan, Python juga memiliki beberapa keterbatasan, yaitu:

1. Kecepatan eksekusi relatif lebih lambat dibandingkan bahasa pemrograman yang dikompilasi seperti C atau C++.
2. Konsumsi memori cenderung lebih besar.
3. Kurang sesuai untuk pengembangan aplikasi yang membutuhkan performa sangat tinggi pada tingkat sistem operasi.
4. Ketergantungan pada pustaka (library) tertentu dapat menimbulkan konflik versi apabila tidak dikelola dengan baik.

2.3.5 Kesimpulan

Python merupakan bahasa pemrograman tingkat tinggi yang sederhana, fleksibel, dan memiliki dukungan pustaka yang sangat lengkap. Kemampuannya dalam mengembangkan aplikasi berbasis Artificial Intelligence, Machine Learning, Deep Learning, dan Computer Vision menjadikan Python sebagai salah satu bahasa pemrograman yang paling banyak digunakan dalam penelitian maupun industri.

Dalam penelitian ini, Python digunakan sebagai bahasa pemrograman utama untuk membangun aplikasi deteksi penggunaan helm proyek berbasis algoritma YOLOv8. Dengan dukungan pustaka seperti OpenCV, Ultralytics, PyTorch, dan Pathlib, Python mampu mengintegrasikan proses pengambilan video, deteksi objek, serta penampilan hasil secara real-time. Oleh

karena itu, Python menjadi komponen penting dalam keberhasilan implementasi sistem deteksi penggunaan helm untuk mendukung pengawasan Keselamatan dan Kesehatan Kerja (K3) di area konstruksi.(Reswara et al., 2025).

2.4 YOLOv8

YOLOv8 (You Only Look Once Version 8) merupakan salah satu algoritma deep learning untuk deteksi objek (object detection) yang dikembangkan oleh Ultralytics sebagai penerus dari versi sebelumnya, seperti YOLOv5 dan YOLOv7. Algoritma ini dirancang untuk mendeteksi dan mengklasifikasikan objek dalam gambar maupun video secara cepat dan akurat. YOLOv8 termasuk dalam kategori single-stage object detector, yaitu melakukan proses identifikasi lokasi objek (localization) dan klasifikasi objek (classification) dalam satu kali proses, sehingga memiliki kecepatan yang sangat tinggi dan sangat sesuai untuk aplikasi real-time.(Widodo et al., 2021)

Berbeda dengan metode deteksi objek dua tahap (two-stage detector), seperti Faster R-CNN yang memerlukan proses pencarian wilayah objek (region proposal) terlebih dahulu, YOLOv8 secara langsung memproses seluruh gambar dalam satu jaringan saraf (neural network). Pendekatan ini membuat YOLOv8 mampu menghasilkan waktu inferensi yang lebih cepat tanpa mengurangi tingkat akurasi secara signifikan. Oleh karena itu, algoritma ini banyak diterapkan pada berbagai bidang, seperti pengawasan keamanan, kendaraan otonom, industri manufaktur, kesehatan,

pertanian, serta sistem Keselamatan dan Kesehatan Kerja (K3). (Harahap, 2024)

YOLOv8 menggunakan arsitektur jaringan saraf konvolusional (Convolutional Neural Network/CNN) yang telah dioptimalkan untuk meningkatkan performa deteksi objek. Algoritma ini terdiri dari tiga bagian utama, yaitu Backbone, Neck, dan Head. Backbone berfungsi mengekstraksi fitur-fitur penting dari gambar masukan, seperti bentuk, warna, dan tekstur objek. Neck berfungsi menggabungkan informasi fitur dari berbagai skala sehingga objek berukuran kecil maupun besar dapat dikenali dengan lebih baik. Selanjutnya, Head bertugas menghasilkan keluaran berupa koordinat bounding box, label kelas objek, dan nilai confidence score sebagai tingkat keyakinan model terhadap hasil deteksi. (Stiki et al., 2026)

Salah satu keunggulan utama YOLOv8 adalah penggunaan pendekatan anchor-free detection, yaitu metode yang tidak lagi bergantung pada anchor box seperti pada beberapa versi YOLO sebelumnya. Pendekatan ini membuat proses pelatihan model menjadi lebih sederhana, mengurangi kompleksitas perhitungan, serta meningkatkan kemampuan model dalam mendeteksi objek dengan berbagai ukuran dan posisi. Selain itu, YOLOv8 juga mendukung berbagai tugas computer vision, seperti deteksi objek (object detection), segmentasi objek (instance segmentation), klasifikasi citra (image classification), estimasi pose (pose estimation), dan pelacakan objek (object tracking). (Nasution et al., 2025)

Pada penelitian ini, YOLOv8 digunakan untuk mendeteksi penggunaan helm proyek (safety helmet) pada pekerja konstruksi secara real-time. Dataset yang telah dikumpulkan dan dianotasi digunakan untuk melatih model YOLOv8 sehingga mampu membedakan objek helmet dan no helmet. Setelah proses pelatihan selesai, model terbaik (best.pt) diintegrasikan ke dalam aplikasi berbasis Python menggunakan library Ultralytics dan OpenCV. Kamera laptop digunakan sebagai media input untuk menangkap video secara langsung. Setiap frame video diproses oleh model YOLOv8, kemudian sistem menampilkan hasil deteksi berupa bounding box, label objek, serta nilai confidence pada layar. Dengan kemampuan tersebut, YOLOv8 mampu mendukung proses pengawasan penggunaan helm secara otomatis, cepat, dan akurat sehingga dapat membantu meningkatkan efektivitas penerapan Keselamatan dan Kesehatan Kerja (K3) di area konstruksi.

2.4.1 Arsitektur YOLOv8

1. Backbone

Backbone merupakan bagian pertama dari jaringan YOLOv8 yang bertugas melakukan ekstraksi fitur (feature extraction) dari citra masukan. Pada tahap ini sistem mempelajari berbagai karakteristik visual, seperti tepi objek (edges), bentuk, tekstur, warna, dan pola tertentu yang terdapat pada gambar.

Semakin dalam lapisan jaringan, semakin kompleks fitur yang dapat dikenali. Misalnya pada penelitian deteksi helm proyek, Backbone akan mempelajari karakteristik bentuk helm, kepala manusia, tubuh pekerja, hingga lingkungan konstruksi sehingga objek dapat dibedakan dengan baik.

Kelebihan YOLOv8

2. Neck

Bagian Neck berfungsi menggabungkan informasi fitur yang diperoleh dari Backbone. YOLOv8 menggunakan konsep Feature Pyramid Network (FPN) dan Path Aggregation Network (PAN) sehingga informasi dari berbagai ukuran objek dapat diproses secara bersamaan.

Tahap ini sangat penting karena pada kondisi nyata ukuran objek tidak selalu sama. Ada pekerja yang berada dekat kamera sehingga ukuran helm terlihat besar, ada pula yang berada jauh sehingga ukuran helm menjadi kecil. Neck membantu sistem agar tetap mampu mendeteksi objek pada berbagai skala tersebut.

3. Head

Head merupakan bagian akhir dari jaringan YOLOv8 yang menghasilkan keluaran (output). Pada tahap ini sistem menentukan:

1. lokasi objek (bounding box);
2. kelas objek (helmet atau no helmet);
3. nilai confidence score;
4. probabilitas setiap kelas.

Output tersebut kemudian ditampilkan pada layar dalam bentuk kotak pembatas (bounding box) beserta label dan tingkat kepercayaan sistem terhadap hasil deteksi.

2.4.2 Cara Kerja YOLOv8

Secara umum, proses kerja YOLOv8 terdiri atas beberapa tahapan sebagai berikut.

1. Akuisisi Data

Kamera laptop menangkap gambar atau video secara real-time. Setiap gambar yang diterima disebut frame.

2. Pre-processing

Frame yang diperoleh dipersiapkan sebelum diproses oleh model, misalnya melalui perubahan ukuran gambar (resize) menjadi 640×640 piksel, normalisasi nilai piksel, dan penyesuaian format data agar sesuai dengan kebutuhan model YOLOv8.

3. Feature Extraction

Frame diproses oleh Backbone untuk memperoleh informasi penting mengenai karakteristik objek.

4. Feature Fusion

Informasi fitur dari berbagai tingkat digabungkan oleh Neck agar model mampu mendeteksi objek dengan ukuran kecil maupun besar.

5. Object Detection

Bagian Head memprediksi:

1. posisi objek;
2. jenis objek;
3. nilai confidence.

6. Non-Maximum Suppression (NMS)

Sering kali satu objek menghasilkan beberapa bounding box. Oleh karena itu YOLOv8 menggunakan proses Non-Maximum Suppression (NMS) untuk memilih kotak dengan nilai confidence tertinggi dan menghapus kotak lain yang saling tumpang tindih.

7. Menampilkan Hasil

Hasil akhir berupa:

1. Bounding Box;
2. Label objek;
3. Nilai Confidence.

Semuanya ditampilkan secara langsung pada video.

2.4.3 Kelebihan Dan Kekurangan YOLOv8

Kelebihan YOLOv8:

YOLOv8 memiliki beberapa keunggulan dibandingkan metode deteksi objek lainnya, yaitu:

1. Mampu mendeteksi objek secara real-time dengan kecepatan tinggi.
2. Memiliki tingkat akurasi yang lebih baik dibandingkan beberapa versi YOLO sebelumnya.
3. Menggunakan metode anchor-free detection sehingga proses pelatihan lebih sederhana.
4. Mampu mendeteksi objek kecil maupun besar dengan lebih baik.
5. Mendukung berbagai tugas computer vision, seperti:
 1. Object Detection;
 2. Image Classification;
 3. Instance Segmentation;
 4. Pose Estimation;
 5. Object Tracking.
6. Mudah digunakan karena telah terintegrasi dengan framework Ultralytics.
7. Dapat dijalankan pada CPU maupun GPU.
8. Mendukung implementasi menggunakan bahasa pemrograman Python.

Kekurangan YOLOv8:

Selain memiliki berbagai kelebihan, YOLOv8 juga memiliki beberapa keterbatasan, antara lain:

1. Membutuhkan dataset yang cukup banyak agar menghasilkan model dengan akurasi tinggi.
2. Proses pelatihan menjadi lebih lama apabila jumlah data sangat besar.
3. Untuk memperoleh performa optimal diperlukan GPU dengan spesifikasi yang memadai.
4. Hasil deteksi dapat dipengaruhi oleh kondisi pencahayaan, sudut pengambilan gambar, dan kualitas kamera.
5. Apabila objek saling menutupi (occlusion), tingkat akurasi deteksi dapat menurun.

2.4.4 Implementasi YOLOv8 Pada Penelitian

Pada penelitian ini, algoritma YOLOv8 digunakan untuk mendeteksi penggunaan helm proyek (safety helmet) pada pekerja konstruksi secara real-time. Dataset berupa gambar pekerja yang menggunakan helm dan tidak menggunakan helm dikumpulkan dari berbagai sumber, kemudian dilakukan proses anotasi menggunakan format YOLO. Dataset tersebut digunakan untuk melatih model sehingga mampu mengenali dua kelas objek, yaitu Helmet dan No Helmet. (Munip et al., 2025)

Model hasil pelatihan disimpan dalam file best.pt dan diintegrasikan ke dalam aplikasi menggunakan bahasa pemrograman Python, library Ultralytics, dan OpenCV. Kamera laptop digunakan sebagai media input untuk menangkap video secara langsung. Setiap frame video diproses oleh model YOLOv8 untuk mendeteksi keberadaan helm. Apabila sistem berhasil mengenali objek, maka akan ditampilkan bounding box, label objek, dan nilai confidence pada layar. Dengan demikian, sistem dapat membantu proses pengawasan penggunaan helm secara otomatis sebagai upaya meningkatkan penerapan Keselamatan dan Kesehatan Kerja (K3) di area konstruksi.

2.4.5 Kesimpulan

YOLOv8 merupakan algoritma deteksi objek berbasis Deep Learning yang memiliki kemampuan mendeteksi objek secara cepat, akurat, dan real-time. Dengan arsitektur Backbone, Neck, dan Head, serta penggunaan metode anchor-free detection, YOLOv8 mampu memberikan performa yang tinggi pada berbagai aplikasi computer vision. Dalam penelitian ini, YOLOv8 digunakan sebagai inti dari sistem deteksi penggunaan helm proyek. Model yang telah dilatih diintegrasikan dengan Python, OpenCV, dan kamera laptop sehingga mampu mengidentifikasi pekerja yang menggunakan maupun tidak menggunakan helm secara otomatis. Implementasi tersebut diharapkan dapat mendukung pengawasan Keselamatan dan

Kesehatan Kerja (K3) di area konstruksi secara lebih efektif, efisien, dan aplikatif.

2.5 Visual Studio Code

Visual Studio Code (VS Code) merupakan perangkat lunak penyunting kode sumber (source code editor) yang dikembangkan oleh Microsoft dan pertama kali dirilis pada tahun 2015. VS Code dirancang sebagai editor kode yang ringan, cepat, dan dapat digunakan secara gratis (free) serta bersifat open source. Perangkat lunak ini mendukung berbagai sistem operasi, seperti Windows, Linux, dan macOS, sehingga dapat digunakan oleh berbagai kalangan, mulai dari mahasiswa, peneliti, hingga pengembang perangkat lunak profesional. (Mailoa & Santoso, 2022)

Visual Studio Code dibangun menggunakan teknologi Electron, yaitu framework yang memungkinkan aplikasi desktop dibuat menggunakan HTML, CSS, dan JavaScript. Meskipun dikembangkan menggunakan teknologi web, VS Code tetap memiliki performa yang tinggi sehingga mampu menjalankan berbagai proyek pemrograman dengan baik.

VS Code mendukung berbagai bahasa pemrograman, seperti Python, Java, JavaScript, PHP, C, C++, C#, HTML, CSS, R, Go, Kotlin, dan masih banyak lagi. Dukungan terhadap banyak bahasa pemrograman tersebut diperoleh melalui fitur Extension Marketplace, sehingga pengguna dapat menambahkan berbagai fungsi sesuai kebutuhan.

Pada bidang kecerdasan buatan (Artificial Intelligence), Machine Learning, dan Computer Vision, Visual Studio Code menjadi salah satu

perangkat lunak yang paling banyak digunakan karena memiliki integrasi yang sangat baik dengan bahasa pemrograman Python serta berbagai pustaka (library) seperti OpenCV, TensorFlow, PyTorch, Keras, dan Ultralytics.

2.5.1 Sejarah Visual Studio Code

Visual Studio Code pertama kali diperkenalkan oleh Microsoft pada acara Build Developer Conference pada tanggal 29 April 2015. Tujuan utama pengembangan VS Code adalah menyediakan editor kode yang ringan tetapi memiliki kemampuan menyerupai Integrated Development Environment (IDE).

Berbeda dengan Microsoft Visual Studio yang memiliki ukuran instalasi besar dan ditujukan untuk pengembangan aplikasi skala besar, Visual Studio Code dirancang agar lebih sederhana, ringan, namun tetap memiliki fitur yang lengkap melalui sistem ekstensi (extensions).

Seiring perkembangan teknologi, VS Code menjadi salah satu editor kode yang paling populer di dunia. Popularitas tersebut disebabkan oleh antarmuka yang sederhana, performa yang cepat, serta dukungan komunitas yang sangat besar.

2.5.2 Fitur-Fitur Visual Studio Code

Visual Studio Code memiliki berbagai fitur yang mendukung proses pengembangan perangkat lunak, antara lain:

1. IntelliSense (Auto Completion)

IntelliSense merupakan fitur yang memberikan saran penulisan kode secara otomatis (auto-complete). Fitur ini membantu pengguna dalam menulis sintaks program dengan lebih cepat, mengurangi kesalahan penulisan (syntax error), serta menampilkan informasi mengenai fungsi atau metode yang digunakan.

2. Debugging

VS Code menyediakan fasilitas debugging yang memungkinkan pengguna menemukan dan memperbaiki kesalahan program. Dengan fitur ini, pengguna dapat menjalankan program langkah demi langkah (step by step), memantau nilai variabel, serta mengidentifikasi letak kesalahan pada kode.

3. Integrated Terminal

VS Code memiliki terminal yang terintegrasi sehingga pengguna tidak perlu membuka Command Prompt atau PowerShell secara terpisah. Terminal ini digunakan untuk menjalankan berbagai perintah, seperti instalasi pustaka Python menggunakan pip, menjalankan program Python, maupun melakukan pelatihan model YOLOv8.

4. Extension Marketplace

Salah satu keunggulan VS Code adalah tersedianya ribuan ekstensi yang dapat dipasang sesuai kebutuhan. Pada penelitian ini, ekstensi yang digunakan antara lain:

1. Python
2. Pylance
3. Jupyter
4. GitHub Copilot (opsional)

Ekstensi tersebut membantu proses penulisan, analisis, serta eksekusi program Python.

5. Syntax Highlighting

Fitur ini memberikan pewarnaan pada sintaks program sehingga kode menjadi lebih mudah dibaca dan dipahami.

6. Git Integration

VS Code telah terintegrasi dengan Git sehingga memudahkan pengguna dalam mengelola versi kode program (version control).

2.5.3 Kelebihan Dan Keurangan Visual Studio Code

Kelebihan :

Visual Studio Code memiliki beberapa keunggulan, antara lain:

1. Bersifat gratis (free) dan open source.
2. Ringan sehingga dapat berjalan pada komputer dengan spesifikasi menengah.
3. Mendukung banyak bahasa pemrograman.
4. Memiliki ribuan ekstensi yang dapat disesuaikan dengan kebutuhan pengguna.
5. Terintegrasi dengan terminal dan Git.
6. Mendukung proses debugging secara langsung.
7. Memiliki antarmuka yang sederhana dan mudah digunakan.

Kekurangan:

Meskipun memiliki banyak kelebihan, Visual Studio Code juga memiliki beberapa keterbatasan, yaitu:

1. Membutuhkan pemasangan ekstensi tambahan untuk memperoleh fitur yang lebih lengkap.
2. Penggunaan banyak ekstensi dapat meningkatkan konsumsi memori (RAM).
3. Tidak semua fitur tersedia secara bawaan seperti pada IDE yang lebih lengkap.

2.5.4Kesimpulan

Visual Studio Code merupakan editor kode sumber yang ringan, fleksibel, dan memiliki berbagai fitur pendukung pengembangan perangkat lunak. Dukungan terhadap bahasa

pemrograman Python, terminal terintegrasi, ekstensi yang lengkap, serta kemampuan debugging menjadikan VS Code sangat sesuai digunakan dalam pengembangan aplikasi berbasis Computer Vision. Pada penelitian ini, Visual Studio Code digunakan sebagai media utama untuk membangun, menjalankan, dan menguji aplikasi deteksi penggunaan helm proyek berbasis algoritma YOLOv8 secara real-time, sehingga mendukung proses pengembangan sistem secara lebih efektif dan efisien.

