



**SISTEM DETEKSI DINI KEBAKARAN MELALUI ASAP BERBASIS
*COMPUTER VISION***



Oleh:
Dithyo Danoetirto Wahid
21410200019

FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA
2025

**SISTEM DETEKSI DINI KEBAKARAN MELALUI ASAP BERBASIS
*COMPUTER VISION***

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana**



UNIVERSITAS
Dinamika

Oleh :

Nama : DITHYO DANOETIRTO WAHID
NIM : 21410200019
Program Studi : S1 Teknik Komputer

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA**

2025

TUGAS AKHIR

SISTEM DETEKSI DINI KEBAKARAN MELALUI ASAP BERBASIS COMPUTER VISION

Dipersiapkan dan disusun oleh:

Dithyo Danoetirto Wahid

NIM: 21410200019

Telah diperiksa, dibahas dan disetujui oleh Dewan Pembahas

Pada: Jumat, 7 Februari 2025

Susunan Dewan Pembahas

Pembimbing:

I. Heri Pratikno, M.T., MTCNA., MTCRE.

NIDN: 0716117302

II. Weny Indah Kusumawati, S.Kom., M.MT.

NIDN: 0721047201

Pembahas:

III. Pauladie Susanto, S.Kom., M.T.

NIDN: 0729047501



Digitally signed by Heri Pratikno, M.T.
DN: cn=Heri Pratikno, M.T.,
o=Universitas Dinamika, ou=S1 Teknik
Komputer, email=heri@dinamika.ac.id,
c=ID
Date: 2025.02.07 13:51:37 +07'00'
Adobe Acrobat version: 11.0.23



cn=Weny Indah Kusumawati,
o=Undika, ou=Prodi S1 TK - FTI,
email=weny@dinamika.ac.id, c=ID
2025.02.07 13:48:17 +07'00'



cn=Pauladie Susanto, o=Universitas
Dinamika, ou=PS S1 Teknik
Komputer,
email=pauladie@dinamika.ac.id, c=ID
2025.02.07 13:57:30 +07'00'

Tugas Akhir ini telah diterima sebagai salah satu persyaratan

Untuk memperoleh gelar Sarjana



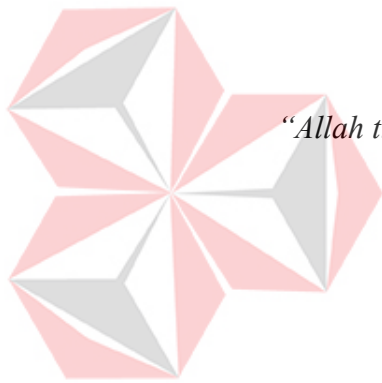
Fakultas Teknologi dan Informatika
UNIVERSITAS
Dinamika

Dr. Anjik Sukmaaji, S.Kom., M.Eng.

NIDN. 0731057301

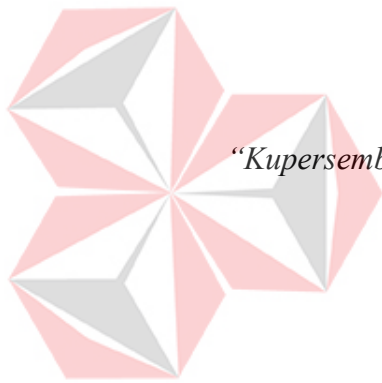
Dekan Fakultas Teknologi dan Informatika

UNIVERSITAS DINAMIKA



*“Allah tidak akan memberi cobaan di luar kemampuan hamba-Nya”
(Al-Baqarah ayat 286)*

UNIVERSITAS
Dinamika



“Kupersembahkan ke Ibu dan Ayah Tercinta, dan Semua yang mengenal dan Menyayangiku”

UNIVERSITAS
Dinamika

PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa **Universitas Dinamika**, Saya :

Nama : **Dithyo Danoetirto Wahid**
NIM : **21410200019**
Program Studi : **S1 Teknik Komputer**
Fakultas : **Fakultas Teknologi dan Informatika**
Jenis Karya : **Laporan Tugas Akhir**
Judul Karya : **SISTEM DETEKSI DINI KEBAKARAN MELALUI
ASAP BERBASIS *COMPUTER VISION***

Menyatakan dengan sesungguhnya bahwa :

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, Saya menyetujui memberikan kepada **Universitas Dinamika** Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalty Free Right*) atas seluruh isi/sebagian karya ilmiah Saya tersebut diatas untuk disimpan, dialihmediakan, dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama Saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut diatas adalah hasil karya asli Saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya, atau pendapat orang lain yang ada dalam karya ilmiah ini semata-mata hanya sebagai rujukan yang dicantumkan dalam Daftar Pustaka Saya.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiasi pada karya ilmiah ini, maka Saya bersedia untuk menerima pencabutan terhadap gelar keserjanaan yang telah diberikan kepada Saya.

Demikian surat pernyataan ini Saya buat dengan sebenar-benarnya.

Surabaya, 23 Januari 2025



Dithyo Danoetirto Wahid
NIM : 21410200019

ABSTRAK

Kebakaran merupakan bencana yang dapat menyebabkan kerugian material maupun korban jiwa. Sistem deteksi kebakaran konvensional umumnya menggunakan sensor fisik seperti sensor suhu dan asap yang membutuhkan waktu untuk mendeteksi perubahan fisik. Teknologi *computer vision* menawarkan pendekatan non-invasif untuk deteksi dini kebakaran melalui deteksi asap menggunakan kamera. Penelitian ini bertujuan untuk mengembangkan sistem deteksi dini kebakaran berbasis *computer vision* menggunakan algoritma YOLOv8, yang mampu mendeteksi asap secara *real-time*. Proses pengembangan penelitian ini meliputi pengumpulan dataset, anotasi, pelatihan, dan validasi. Hasil penelitian menunjukkan bahwa model YOLOv8 dengan 150 *epoch* mampu mendeteksi asap dengan akurasi yang bervariasi tergantung dari metode pengujian. Pengujian dengan metode panci yang dibiarkan terbuka menghasilkan rata-rata akurasi sebesar 80% untuk pengujian kertas, rata-rata akurasi pengujian plastik sebesar 48% dan rata-rata akurasi pengujian kayu sebesar 56%. Sedangkan, pengujian dengan metode panci yang ditutup lalu dibuka menghasilkan rata-rata akurasi 92% untuk pengujian kertas, rata-rata akurasi pengujian plastik sebesar 88% dan rata-rata akurasi pengujian kayu sebesar 100%. Selain itu, sistem juga menunjukkan performa stabil dengan rata-rata FPS sebesar 6.36.

Kata Kunci: *Computer Vision*, YOLOv8, Deteksi Asap, FPS.



UNIVERSITAS
Dinamika

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan Laporan Tugas Akhir yang berjudul "Sistem Deteksi Dini Kebakaran Melalui Asap Berbasis *Computer Vision*". Selama proses penyusunan Laporan Tugas Akhir ini, penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu, memberikan saran, masukan dan dukungan. Ucapan terima kasih ini ditujukan kepada:

1. Allah SWT, karena dengan Rahmat-Nya penulis dapat menyelesaikan Laporan Tugas Akhir ini.
2. Orang tua dan seluruh keluarga, yang telah memberikan dukungan dan dorongan selama proses penyelesaian Laporan Tugas Akhir ini.
3. Bapak Dr. Anjik Dukmaaji, S.Kom., M.Eng., selaku Dekan Fakultas Teknologi dan Informatika Universitas Dinamika.
4. Bapak Pauladie Susanto, S.Kom., M.T., selaku Ketua Program Studi S1 Teknik Komputer sekaligus dosen pembahas, atas masukan dan saran, sehingga penulis dapat menyelesaikan Laporan Tugas Akhir ini.
5. Bapak Heri Pratikno, M.T., MTCNA., MTCRE., selaku Dosen Pembimbing 1 yang telah memberikan masukan, wawasan dan arahan bagi penulis selama pengerjaan dan pembuatan Laporan Tugas Akhir ini.
6. Ibu Weny Indah Kusumawati, S.Kom., M.MT., selaku Dosen Pembimbing 2 yang telah memberikan bimbingan, dukungan, dan motivasi bagi penulis pengerjaan dan pembuatan Laporan Tugas Akhir ini.
7. Seluruh rekan-rekan S1 Teknik Komputer Angkatan 2021, yang telah membantu memberikan dukungan dan semangat.
8. Seluruh pihak yang tidak dapat disebutkan satu per satu, yang telah memberikan dukungan serta bantuan, baik secara langsung maupun tidak langsung dalam penyelesaian Laporan Tugas Akhir ini.

Penulis sangat menyadari bahwa masih terdapat kekurangan pada penulisan laporan ini. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun agar laporan ini menjadi lebih baik. Semoga laporan ini dapat berguna dan bermanfaat untuk menambah wawasan bagi pembaca.

DAFTAR ISI

	Halaman
ABSTRAK	vii
KATA PENGANTAR.....	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	2
1.5 Manfaat	3
BAB II LANDASAN TEORI	4
2.1 Kebakaran	4
2.2 <i>Computer Vision</i>	4
2.3 YOLO.....	5
2.4 Python	7
2.5 <i>Visual Studio Code</i>	7
2.6 <i>Google Colaboratory</i>	8
2.7 <i>Roboflow</i>	9
BAB III METODOLOGI PENELITIAN.....	10
3.1 Studi Literatur	10
3.2 Pengumpulan Dataset.....	10
3.3 Pengolahan Dataset	11
3.4 <i>Training Dataset</i>	13
3.5 <i>Flowchart</i>	14
3.5.1 <i>Flowchart</i> Tahap Persiapan Kerja	14
3.5.2 <i>Flowchart</i> Cara Kerja Alat	15
3.6 Blok Diagram	16
BAB IV HASIL DAN PEMBAHASAN.....	18

4.1 Validasi Proses <i>Training Data</i>	18
4.1.1 Prosedur Validasi <i>Training Data</i>	18
4.1.2 Hasil Validasi <i>Custom Model</i>	18
4.2 Pengujian Deteksi Asap.....	18
4.2.1 Metode Pengujian Deteksi Asap.....	19
4.2.2 Prosedur Pengujian.....	20
4.2.3 Hasil Pengujian Deteksi Asap	21
BAB V PENUTUP.....	28
5.1 Kesimpulan	28
5.2 Saran.....	29
DAFTAR PUSTAKA	30
LAMPIRAN.....	31



UNIVERSITAS
Dinamika

DAFTAR GAMBAR

	Halaman
Gambar 2.1 <i>Architecture</i> YOLO	6
Gambar 2.2 Logo Python	7
Gambar 2.3 Logo Visual Studio Code	8
Gambar 2.4 Logo Google Colaboratory.....	8
Gambar 2.5 Logo Roboflow	9
Gambar 3.1 Deteksi Asap.....	10
Gambar 3.2 Proses memberi label dan <i>class</i> pada gambar	11
Gambar 3.3 Proses <i>Train/Test Split</i>	11
Gambar 3.4 Proses <i>preprocessing</i> dataset.....	12
Gambar 3.5 Memilih model yang akan digunakan	13
Gambar 3.6 Dataset yang berhasil diimpor menggunakan <i>API Key</i>	13
Gambar 3.7 <i>Training</i> dataset 150 <i>epochs</i>	14
Gambar 3.8 <i>Flowchart</i> tahap persiapan kerja	15
Gambar 3.9 <i>Flowchart</i> cara kerja alat.....	16
Gambar 3.10 Blok Diagram	17
Gambar 4.1 Metode pengujian panci yang dibiarkan terbuka	19
Gambar 4.2 Metode panci yang ditutup	20

DAFTAR TABEL

	Halaman
Tabel 4.1 Validasi <i>Custom Model</i>	18
Tabel 4.2 Hasil pengujian kertas dengan metode panci yang dibiarkan terbuka ..	21
Tabel 4.3 Hasil pengujian plastik dengan metode panci yang dibiarkan terbuka .	22
Tabel 4.4 Hasil pengujian kayu dengan metode panci yang dibiarkan terbuka	23
Tabel 4.5 Hasil pengujian kertas dengan metode panci yang ditutup lalu dibuka	24
Tabel 4.6 Hasil pengujian plastik dengan metode panci ditutup lalu dibuka.....	25
Tabel 4.7 Hasil pengujian kayu dengan metode panci ditutup lalu dibuka.....	26



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Program Deteksi Asap Secara <i>Real-Time</i>	31
Lampiran 2 Program Deteksi Asap Melalui Gambar	32
Lampiran 3 Program Deteksi Asap Melalui Video	33
Lampiran 4 Bukti Originalitas	34
Lampiran 5 Form Bimbingan.....	39
Lampiran 6 Biodata Penulis	40



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebakaran adalah bencana yang menimbulkan banyak kerugian, baik dari segi material maupun korban jiwa. Kasus kebakaran biasanya terjadi di industri besar dan kawasan hutan. Kebakaran di daerah industri menjadi perhatian khusus, karena tidak hanya merugikan dari segi ekonomi, tetapi juga berpotensi menimbulkan korban jiwa. Maka dari itu, diperlukan deteksi dini kebakaran untuk mencegah kerugian material maupun korban jiwa.

Sebagian besar sistem pendeteksi kebakaran pada saat ini masih bergantung pada sensor fisik, seperti sensor suhu dan asap, yang bekerja dengan mendeteksi perubahan fisik di lingkungan. Meskipun sistem ini bekerja dengan baik, sistem ini memerlukan waktu untuk mendeteksi adanya kebakaran karena sensor fisik baru aktif setelah perubahan suhu atau konsentrasi asap mencapai titik tertentu. Selain itu, untuk mencakup area yang luas sensor juga harus dipasang dan dipelihara, yang tentunya menambah biaya. Seperti penelitian yang dilakukan oleh (Waworundeng, 2020) sensor asap dan api berbasis *Internet of Things* dapat mendeteksi kebakaran api dengan akurat, namun biaya instalasi dan pemeliharaan sensor di area industri yang luas menjadi masalah utamanya.

Semakin berkembangnya teknologi, *computer vision* memberikan alternatif untuk deteksi dini kebakaran. Teknologi ini tidak hanya mampu mendeteksi asap dengan akurat, tetapi juga dapat mencakup area yang luas dengan menggunakan kamera yang sudah ada di lokasi. Pada penelitian yang dilakukan oleh (Fajri & Priambodo, 2023) mampu mendeteksi asap dan api menggunakan metode mask r-cnn, namun nilai *average precision* masih tergolong rendah yaitu 0.38 dan *average recall* sebesar 0.29. Pada penelitian yang dilakukan oleh (Wirajati & Natha, 2021) menunjukkan kemampuan deteksi *computer vision* yang baik, namun fokus penelitian yang dilakukan hanya mendeteksi api saja. Selain itu, pada penelitian yang dilakukan oleh (Hayati et al., 2023) menggunakan YOLOv8 untuk mendeteksi jumlah kendaraan. Hasil dari penelitian ini menunjukkan *average precision* sebesar 89% dan *average recall* sebesar 90%.

Berdasarkan latar belakang yang telah dijelaskan diatas, maka pada Tugas Akhir ini akan mengembangkan sebuah sistem yang mampu mendeteksi asap secara dini untuk mencegah terjadinya bahaya kebakaran. Sistem deteksi dini kebakaran melalui asap ini diterapkan berbasis *computer vision* yang mampu mendeteksi asap melalui sebuah citra atau dari sebuah video yang dapat berjalan dalam waktu nyata (*real-time*). Dengan adanya penelitian ini dapat memberikan alternatif dalam deteksi dini kebakaran melalui asap secara non-invasif tanpa perlu adanya kontak fisik antara asap dan sensor karena menggunakan kamera.

1.2 Rumusan Masalah

Bagaimana merancang dan mengimplementasikan sistem deteksi dini kebakaran melalui deteksi asap berbasis *computer vision* yang berjalan secara *real-time* serta berapa besar tingkat akurasi deteksinya?

1.3 Batasan Masalah

Berdasarkan latar belakang diatas, maka dapat dirumuskan masalah sebagai berikut:

1. Sistem ini hanya mendeteksi asap sebagai tanda kebakaran, bukan peningkatan suhu atau deteksi api lainnya.
2. Sistem ini tidak dirancang untuk mendeteksi asap tipis misalnya asap rokok, terutama pada kondisi pencahayaan yang buruk, serta kinerjanya bergantung pada kualitas kamera yang digunakan.
3. Lingkup uji coba sistem ini dibatasi pada kawasan tertentu dengan ukuran area terbatas, tidak mencakup lingkungan dengan cuaca ekstrem atau area yang sangat luas.
4. Sistem ini berjalan pada pencahayaan ruang yang merata atau ideal.

1.4 Tujuan

Berdasarkan latar belakang dan rumusan masalah di atas, berikut merupakan tujuan dari penelitian ini:

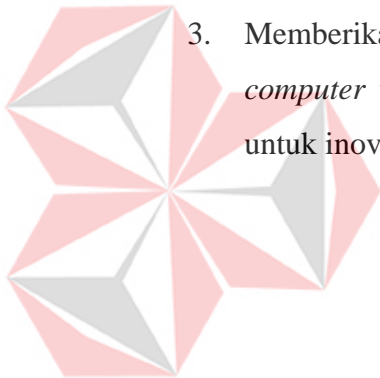
1. Mampu merancang dan mengimplementasikan sistem deteksi dini kebakaran melalui deteksi asap berbasis *computer vision* yang berjalan secara *real-time*.

2. Mampu menguji tingkat akurasi deteksi sistem dalam mendeteksi keberadaan asap.
3. Mampu mengukur kinerja sistem secara real-time dengan menggunakan parameter FPS (*Frame per Second*).

1.5 Manfaat

Adapun manfaat yang dapat diperoleh dari penelitian ini, yaitu:

1. Memberikan alternatif dalam mendeteksi potensi kebakaran secara dini, sehingga dapat mengurangi risiko kerugian akibat kebakaran yang dapat merusak aset perusahaan.
2. Memberikan alternatif untuk mendeteksi kebakaran pada tahap awal di tempat kerja, sehingga memberikan peringatan dini dan waktu yang cukup untuk evakuasi.
3. Memberikan landasan untuk penelitian lebih lanjut tentang penggunaan *computer vision* untuk mengurangi risiko kebakaran dan menjadi referensi untuk inovasi teknologi keselamatan.



UNIVERSITAS
Dinamika

BAB II

LANDASAN TEORI

2.1 Kebakaran

Kebakaran adalah keadaan di mana api tidak dapat dikendalikan, sering kali berkembang dengan cepat dan menyebar ke area yang lebih luas jika tidak segera ditangani (Ferguson et al., 2020). Kebakaran yang terjadi di lingkungan industri biasanya dipicu oleh berbagai faktor, seperti kegagalan alat-alat listrik yang mengalami korsleting, paparan bahan kimia yang mudah terbakar, serta kelalaian manusia, seperti penggunaan peralatan dengan standar keamanan yang tidak memadai. Kebakaran terjadi ketika tiga elemen penting bersatu yaitu panas, bahan bakar, dan oksigen. Hal ini dikenal sebagai segitiga api. Tanpa salah satu dari unsur ini, api tidak akan terbentuk, namun ketika ketiganya bersatu, risiko kebakaran meningkat secara signifikan.

Kebakaran di industri dikelompokkan menjadi beberapa kelas berdasarkan material yang terlibat dalam pembakaran. Kebakaran kelas A mencakup bahan padat yang mudah terbakar, seperti kayu, kain, kertas, dan plastik. Kebakaran kelas B melibatkan bahan cair atau gas yang mudah terbakar, seperti minyak, bensin, dan gas alam. Sedangkan kebakaran kelas C berhubungan dengan peralatan listrik, seperti motor, generator, dan perangkat elektronik yang mengalami kegagalan atau arus pendek. Setiap kelas kebakaran memerlukan penanganan yang berbeda, jika menggunakan metode pemadaman yang salah, seperti penggunaan air pada kebakaran kelas B atau C, dapat memperburuk situasi.

2.2 Computer Vision

Computer Vision merupakan salah satu cabang ilmu pengetahuan yang bertujuan untuk membuat suatu keputusan yang berguna mengenai objek fisik nyata dan keadaan berdasarkan sebuah gambar atau citra (Rahmadhika & Thantawi, 2021). Dalam hal ini sistem komputer dilatih untuk melihat dan memahami dunia sekitarnya dengan cara yang mirip dengan manusia. Teknologi ini memungkinkan komputer untuk menganalisis gambar atau video secara otomatis untuk mendeteksi objek, mengenali pola, dan membuat keputusan berdasarkan data visual.

Deteksi kebakaran adalah contoh lain dari penggunaan *computer vision*, di mana teknologi ini digunakan untuk memantau area secara *real-time* untuk menemukan tanda-tanda awal kebakaran, seperti asap. Sistem *computer vision* dapat mendeteksi perubahan warna dan tekstur yang tidak biasa pada gambar dengan menggunakan kamera dan software analisis gambar. Algoritma *Computer vision* ada bermacam-macam salah satunya adalah algoritma YOLO dan *Convolutional Neural Networks (CNN)* yang telah terbukti efektif dalam deteksi kebakaran dan asap dalam berbagai skenario indoor maupun outdoor (Saponara et al., 2021).

2.3 YOLO

You Only Look Once atau biasa disebut dengan YOLO adalah algoritma yang digunakan untuk mendeteksi objek berbasis deep learning. YOLO dikenal dengan kemampuannya dalam mendeteksi objek dalam gambar atau video. Metode ini tidak memerlukan langkah-langkah terpisah untuk menghasilkan proposal wilayah, mengekstraksi fitur, dan melakukan klasifikasi, YOLO melakukan semua proses tersebut hanya dalam satu fase. Cara kerja deteksi YOLO ini membagi gambar menjadi grid berukuran $S \times S$ untuk mendeteksi objek yang pusatnya berada di dalam sel tersebut, setiap sel memprediksi *bounding box* (kotak pembatas), memprediksi skor kepercayaan (*Confidence Score*) dan untuk menghindari deteksi ganda, YOLO menerapkan NMS atau (*Non-Maximum Suppression*). Pada proses NMS, *bounding box* dengan skor tertinggi akan dipertahankan dan menghilangkan *bounding box* lain yang tumpang tindih (*Overlap*).

YOLO pertama kali dirilis pada tahun 2015 dan telah mengalami berbagai perkembangan dari pertama kali dirilis hingga YOLOv8. Berikut merupakan penjelasan mengenai setiap versi YOLO:

1. YOLOv1: Memiliki kecepatan deteksi objek secara *real-time* dengan cepat dan efisien, namun pada versi ini kurang akurat terutama dalam mendeteksi objek yang kecil dalam sekelompok gambar dan kesulitan mendeteksi bentuk yang belum pernah dilihat sebelumnya.
2. YOLOv2: Memperkenalkan teknik seperti *batch normalization* untuk stabilisasi pelatihan, resolusi input yang lebih tinggi yaitu 448×448 untuk

2. *convolutional* 1x1 pertama kali diterapkan untuk mengurangi jumlah *channel*, kemudian diikuti oleh *convolutional* 3x3 untuk menghasilkan *cuboidal output*.
3. Fungsi aktivasi yang digunakan adalah ReLU, kecuali pada lapisan terakhir yang menggunakan fungsi aktivasi linear.
4. Beberapa teknik tambahan, seperti *batch normalization* dan *dropout*, digunakan untuk meregulasi model dan mencegah overfitting.

Pada penelitian ini YOLOv8 dipilih karena menawarkan peningkatan yang signifikan dalam akurasi dan efisiensi deteksi objek dibandingkan dengan versi sebelumnya. Dengan arsitektur yang ditingkatkan dan teknik ekstraksi fitur yang lebih baik, YOLOv8 mampu mendeteksi objek dengan lebih akurat dan cepat. Selain itu, kemudahan integrasi dan ukuran model yang lebih kecil memungkinkan implementasi pada perangkat dengan sumber daya terbatas.

2.4 Python

Python adalah bahasa pemrograman yang sering digunakan untuk membuat aplikasi berbasis kecerdasan buatan, termasuk computer vision. Python memiliki banyak pustaka yang mendukung analisis data dan pemrosesan gambar, seperti OpenCV, YOLO, TensorFlow, dan Keras. Bahasa pemrograman Python ditandai dengan sintaksis yang mudah dibaca dan dipahami, yang memungkinkan pemrogram untuk mengembangkan kode dengan cepat dan efisien (Kairos Abinaya Susanto et al., 2023). Bahasa python ini dipilih karena sintaksisnya yang sederhana dan dukungan pustaka yang luas. Logo *python* dapat dilihat pada Gambar 2.2

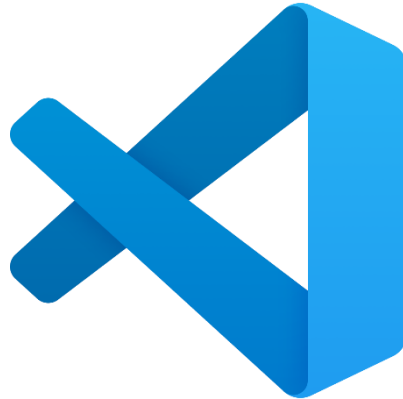


Gambar 2.2 Logo Python
(Sumber: <https://www.python.org/>)

2.5 Visual Studio Code

Visual Studio Code (VS Code) ini adalah sebuah teks editor ringan dan handal yang dibuat oleh *Microsoft*. Aplikasi ini juga tersedia untuk versi *Linux*,

Mac, dan *Windows*. Dalam proses pengembangan sistem deteksi dini kebakaran berbasis *computer vision*, *VS code* digunakan untuk *deployment* model setelah tahap evaluasi model selesai. *VS Code* menyediakan beberapa fitur yang mendukung efisiensi selama proses pengembangan, salah satunya yaitu integrasi yang mudah dengan *Git* dan *GitHub*. Logo *vs code* dapat dilihat pada Gambar 2.3



Gambar 2.3 Logo Visual Studio Code
(Sumber: <https://code.visualstudio.com/brand>)

2.6 Google Colaboratory

Google Colaboratory atau biasa disebut dengan *Google Colab* adalah *platform* berbasis cloud yang memungkinkan pengembang menulis dan menjalankan kode program langsung dari *browser* mereka. *Google Colab* menyediakan akses gratis ke *GPU* dan *TPU*, selain itu *Google Colab* juga menyediakan dukungan pustaka yang lengkap tanpa perlu instalasi tambahan pada perangkat lokal. Pada penelitian ini, *Google Colab* digunakan untuk melatih dan menguji model deteksi asap menggunakan algoritma *YOLO*. Logo *google colab* dapat dilihat pada Gambar 2.4



Gambar 2.4 Logo Google Colaboratory
(Sumber: <https://colab.google/>)

2.7 Roboflow

Roboflow adalah sebuah *platform* untuk mengelola dataset dan melakukan anotasi data dengan optimal. *Platform* ini memungkinkan pengembang untuk mengelola dataset dengan mudah. *Roboflow* dirancang untuk meningkatkan alur kerja dalam pengembangan model *computer vision*, *platform* ini menawarkan beragam fitur untuk membantu dalam pelabelan data, *pre-processing dataset*, hingga pelatihan model. Logo roboflow dapat dilihat pada Gambar 2.5



Gambar 2.5 Logo Roboflow
(Sumber: <https://github.com/roboflow>)



UNIVERSITAS
Dinamika

BAB III METODOLOGI PENELITIAN

3.1 Studi Literatur

Pada tahap ini dilakukan penelitian dan kajian literatur dari berbagai sumber ilmiah untuk mendapatkan pemahaman yang lebih mendalam mengenai topik dan judul penelitian yang akan dikerjakan. Hal ini dilakukan dengan membaca dan menganalisa berbagai jurnal, artikel ilmiah, dan buku yang relevan dengan tema penelitian.

3.2 Pengumpulan Dataset

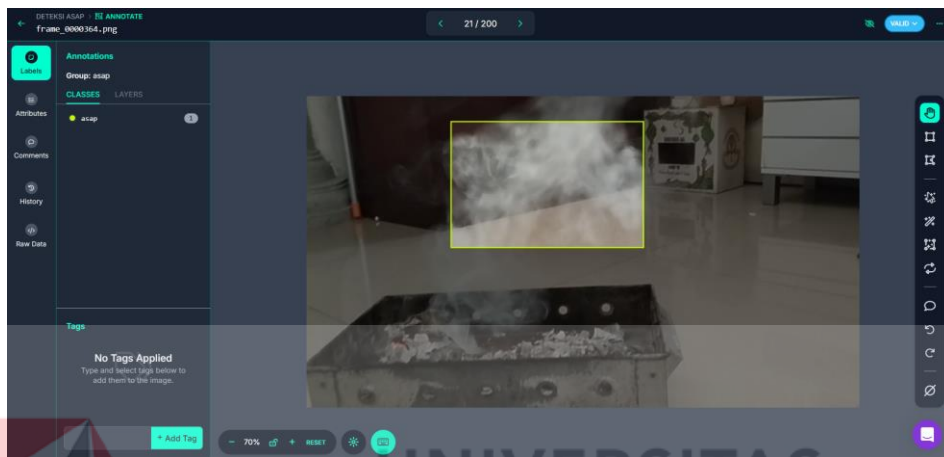
Dataset yang digunakan dalam penelitian ini terdiri dari 5080 gambar asap yang dikumpulkan secara mandiri menggunakan kamera di dalam ruangan. Proses pengumpulan dataset dilakukan dengan membakar bahan-bahan yang menghasilkan asap, yaitu 3165 gambar dari pembakaran kertas, 153 gambar dari pembakaran plastik, dan 1762 gambar dari pembakaran kayu. Gambar-gambar ini dipilih untuk meningkatkan keragaman jenis asap yang dihasilkan. Setelah data terkumpul, gambar-gambar tersebut diunggah ke *platform Roboflow* untuk diproses. Pada tahap ini area yang mengandung asap akan diberi label untuk melatih model *computer vision* agar model dapat mendeteksi asap dengan akurat. Gambar 3.1 adalah contoh dari deteksi asap secara *real-time*.



Gambar 3.1 Deteksi Asap

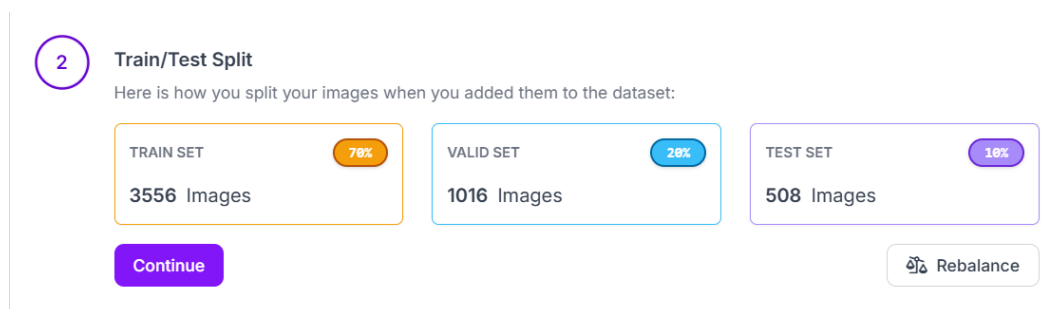
3.3 Pengolahan Dataset

Pada tahap ini dataset yang telah dikumpulkan akan di proses pada *platform Roboflow* untuk melatih model. Proses ini dimulai dengan memberi anotasi dan memberi nama *class* pada gambar asap. Anotasi ini melibatkan penandaan area asap secara manual sehingga model dapat dilatih untuk mengenali pola visual yang spesifik. Anotasi gambar asap bisa dilihat pada Gambar 3.2



Gambar 3.2 Proses memberi label dan *class* pada gambar

Setelah proses anotasi selesai, langkah selanjutnya adalah *Train/Test Split*. Pada tahap ini membagi dataset menjadi data *training*, data valid, dan data *testing*. Pada penelitian ini dataset dibagi menjadi 70% data *training*, 20% data valid, dan 10% data *testing*. Pembagian ini dilakukan agar model dapat mengenali pola asap dengan baik. Proses *Train/Test Split* dapat dilihat pada Gambar 3.3



Gambar 3.3 Proses *Train/Test Split*

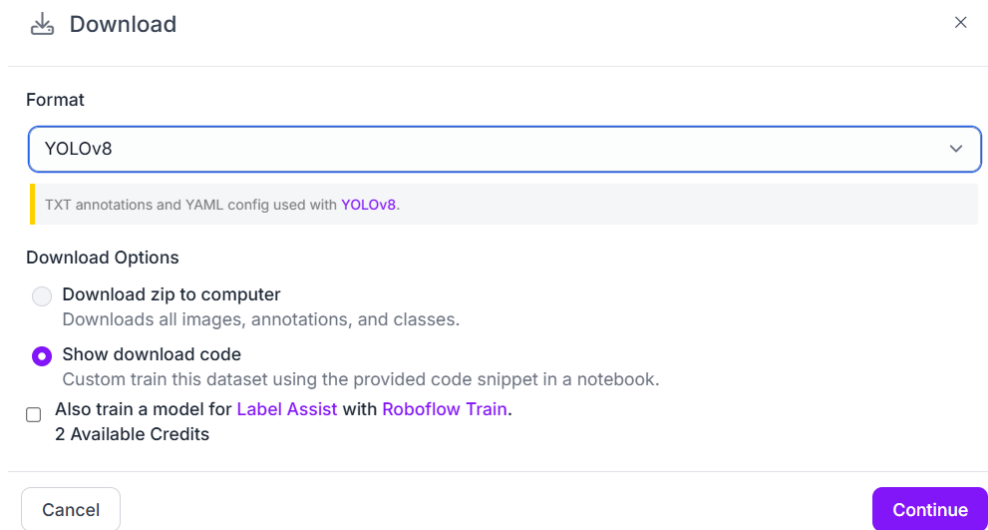
Berikutnya adalah tahap *preprocessing*. Pada tahap ini gambar yang telah di anotasi dan sudah dibagi menjadi data *training*, data valid, dan data *testing* akan di

ubah ukurannya sesuai kebutuhan. Pada penelitian ini ukuran gambar diubah menjadi 640x640 piksel untuk memberikan keseimbangan antara kecepatan inferensi dan akurasi. Setelah semua proses sudah selesai, selanjutnya *create*. Proses preprocessing dataset dapat dilihat pada Gambar 3.4



Gambar 3.4 Proses *preprocessing* dataset

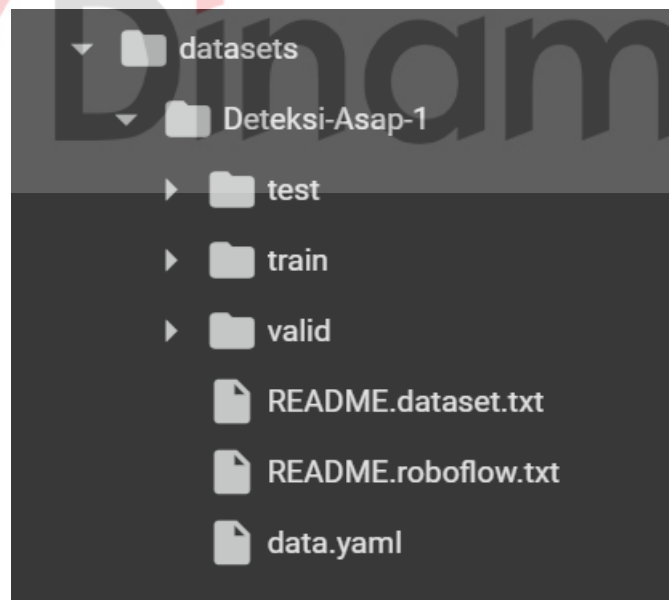
Tahap selanjutnya memilih model apa yang akan digunakan. Pada penelitian ini model yang digunakan adalah YOLOv8. Setelah itu, Roboflow akan memberi dua opsi yaitu mendownload dataset yang telah diproses ke komputer dalam bentuk zip atau menggunakan *API Key* untuk diproses lebih lanjut. Tahap pemilihan model dapat dilihat pada Gambar 3.5



Gambar 3.5 Memilih model yang akan digunakan

3.4 Training Dataset

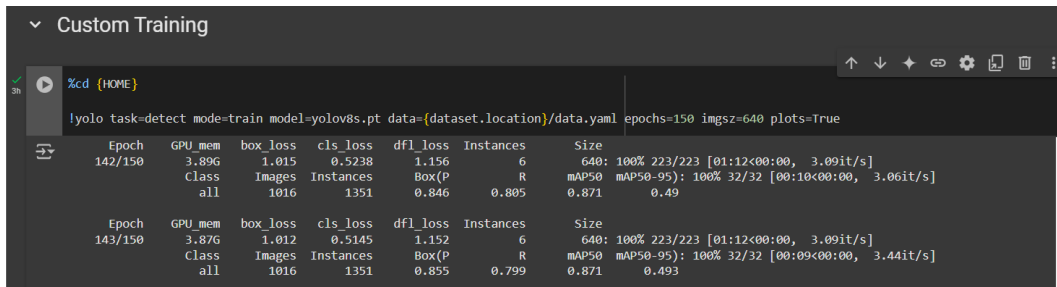
Pada tahap ini, dataset yang sudah diproses pada *platform* Roboflow dan diperoleh *API Key* akan diimpor ke dalam *Google Colab* untuk proses pelatihan. Gambar 3.6 merupakan dataset yang telah berhasil diimpor menggunakan *API Key*.



Gambar 3.6 Dataset yang berhasil diimpor menggunakan *API Key*

Model dilatih menggunakan data *training* untuk melatih model mengenali pola asap. Data valid digunakan untuk memonitor performa model selama proses pelatihan. Pelatihan dilakukan menggunakan *GPU* gratis dari *Google Colab* untuk

mempercepat komputasi. Selama pelatihan, parameter seperti *epoch* dan *imgsz* diatur agar mendapatkan hasil yang optimal.



```

Custom Training
%cd {HOME}
yolo task=detect mode=train model=yolov8s.pt data={dataset.location}/data.yaml epochs=150 imgsz=640 plots=True

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
142/150  3.89G   1.015    0.5238   1.156     6          640: 100% 223/223 [01:12<00:00, 3.09it/s]
Class   Images  Instances  Box(P)   R          mAP50  mAP50-95): 100% 32/32 [00:10<00:00, 3.06it/s]
all     1016    1351     0.846   0.805     0.871   0.49

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
143/150  3.87G   1.012    0.5145   1.152     6          640: 100% 223/223 [01:12<00:00, 3.09it/s]
Class   Images  Instances  Box(P)   R          mAP50  mAP50-95): 100% 32/32 [00:09<00:00, 3.44it/s]
all     1016    1351     0.855   0.799     0.871   0.493

```

Gambar 3.7 Training dataset 150 epochs

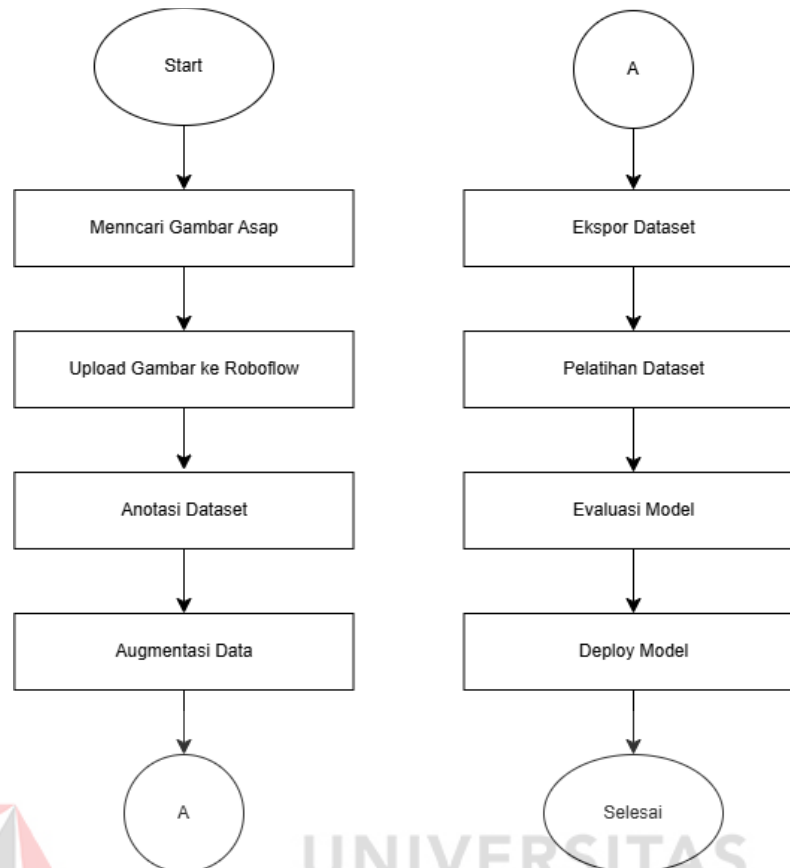
Pelatihan dilakukan 5 kali dengan *epochs* sebanyak 25, 50, 75, 100, dan 150. Hasil pelatihan berupa model yang sudah terlatih yaitu ‘best.pt’ dan dievaluasi menggunakan model terbaik untuk mengukur mengukur akurasi, *precision*, dan *recall* dalam mendeteksi asap.

3.5 Flowchart

Berikut merupakan *flowchart* dari penelitian ini:

3.5.1 Flowchart Tahap Persiapan Kerja

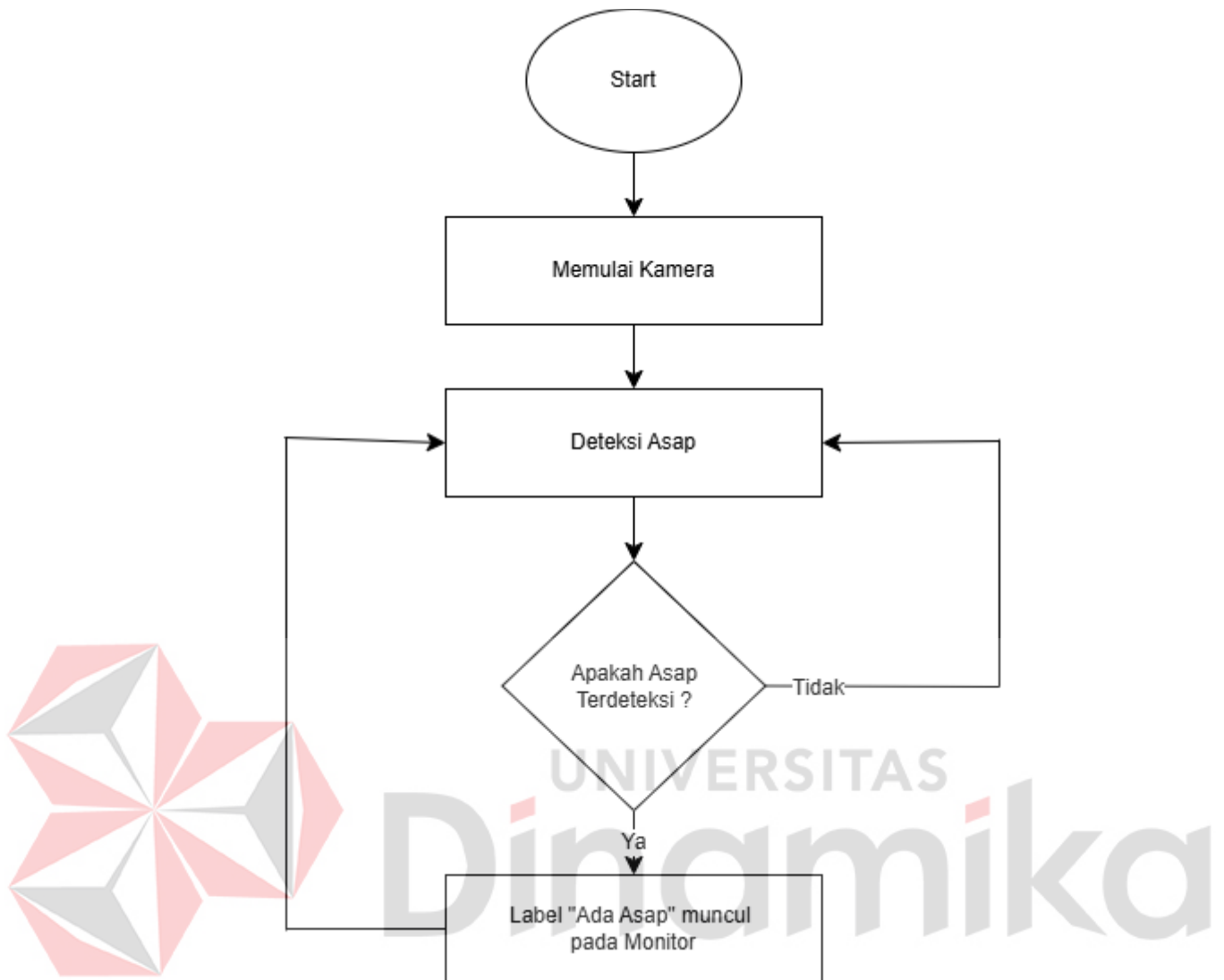
Pada proses ini dimulai dengan mengumpulkan dataset berisi gambar asap, gambar diambil secara mandiri menggunakan kamera. Dataset tersebut kemudian diproses menggunakan *Roboflow* untuk melakukan anotasi dan *preprocessing*. Dataset yang sudah lengkap kemudian dilatih menggunakan *Google Colab*. Terakhir, mengevaluasi performa model untuk memastikan akurasi, *precision*, dan *recall* sebelum implementasi sistem. *Flowchart* tahap persiapan kerja dapat dilihat pada Gambar 3.8



Gambar 3.8 *Flowchart* tahap persiapan kerja

3.5.2 *Flowchart* Cara Kerja Alat

Pada proses ini kamera akan diaktifkan dengan fungsi `cap = cv2.VideoCapture(video_source)`. Setelah kamera aktif, sistem akan mendeteksi asap secara *real-time*. Model kemudian mendeteksi keberadaan asap melalui kamera dengan fungsi `results = model.predict(source=frame, imsz=640, conf=0.5)`. Jika asap terdeteksi maka akan ada label “asap” pada monitor. Namun, jika tidak ada asap yang terdeteksi, sistem akan melanjutkan pemantauan secara *real-time*. *Flowchart* cara kerja alat dapat dilihat pada Gambar 3.9

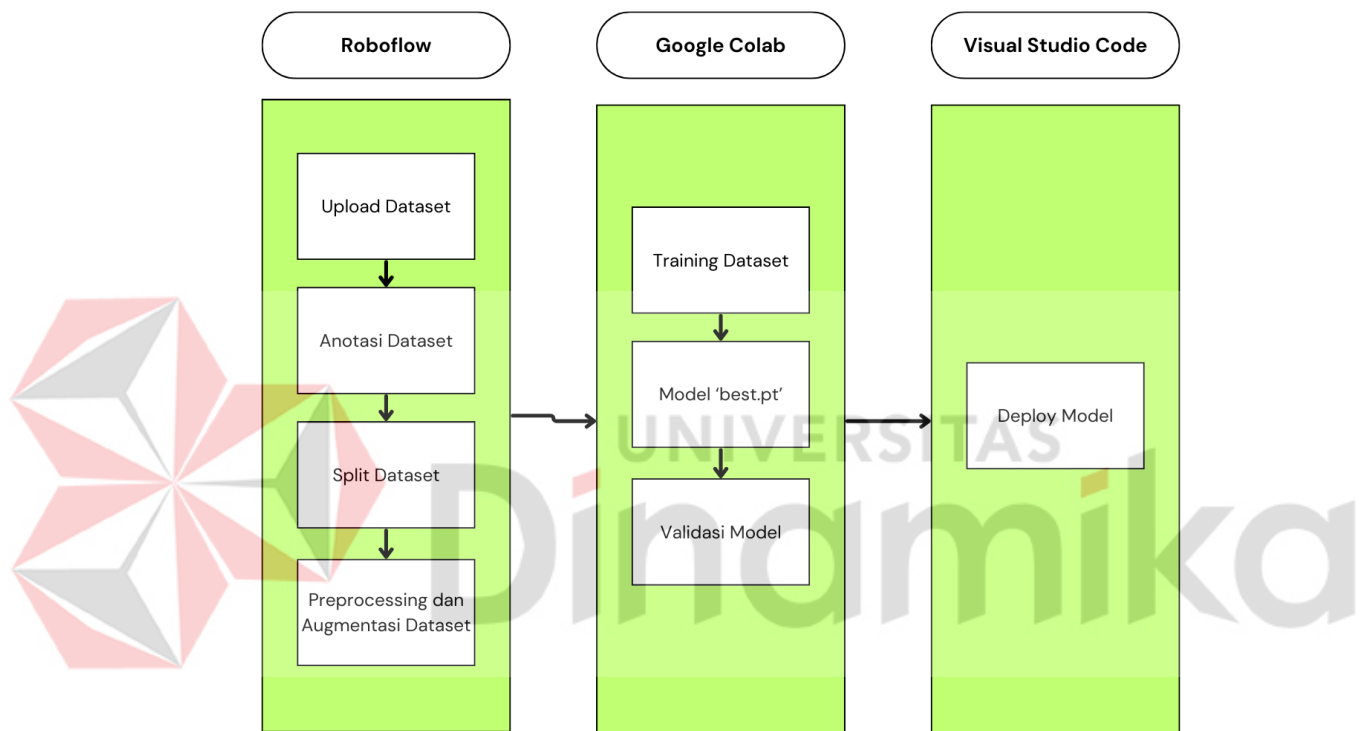


Gambar 3.9 Flowchart cara kerja alat

3.6 Blok Diagram

Blok diagram yang terdapat pada Gambar 3.10 merupakan penjelasan rincian langkah-langkah pemrosesan dari awal pengolahan dataset hingga proses *deploy model*. Pertama dataset yang telah dikumpulkan akan di *upload* ke *platform* Roboflow untuk memulai proses anotasi. Anotasi merupakan proses melabeli bagian yang terdapat asap dan memberi nama *class* untuk mengenali pola asap. Dataset yang telah di anotasi akan di bagi menjadi data *training*, data valid dan data *testing*. Selanjutnya, dataset yang telah dibagi akan diubah ukuran gambar sesuai kebutuhan dan memilih model yang akan digunakan. Kemudian Roboflow akan memberi *API Key* untuk diproses lebih lanjut.

Tahap selanjutnya, *API Key* disalin dan dimasukkan ke google colab untuk melatih model mengenali pola asap. Setelah proses *training*, selanjutnya model divalidasi dengan mengukur parameter seperti *precision* dan *recall* untuk memastikan performa terbaik sebelum *deploy* model ke Visual Studio Code. Pada Visual Studio Code, model terbaik akan diuji kemampuannya secara *real-time* menggunakan parameter seperti akurasi, FPS dan waktu deteksi model dalam mendeteksi asap.



Gambar 3.10 Blok Diagram

BAB IV HASIL DAN PEMBAHASAN

4.1 Validasi Proses *Training Data*

Pada bagian ini, proses validasi bertujuan untuk mengevaluasi kinerja model yang telah dilatih. Pada penelitian ini model yang digunakan adalah YOLOv8. Model akan dievaluasi pada setiap *epoch* untuk mengetahui seberapa baik performa model dalam memahami data.

4.1.1 Prosedur Validasi *Training Data*

Berikut merupakan prosedur dari validasi training data:

1. Login pada akun Google Colab dan Roboflow.
2. Menginstall *environment* yang diperlukan untuk *training* dataset.
3. Memasukkan API dataset yang telah diunduh pada platform Roboflow ke dalam Google Colab.
4. *Training* dataset menggunakan Google Colab dengan jumlah 25 epoch, 50 epoch, 75 epoch, 100 epoch dan 150 epoch untuk mengevaluasi kinerja model.

4.1.2 Hasil Validasi *Custom Model*

Setelah proses *training* selesai, model akan dievaluasi dan diukur performanya. Metrik yang digunakan dalam evaluasi yaitu Epoch, Precision, Recall dan mAP. Hasil validasi model dapat dilihat pada tabel 4.1

Tabel 4.1 Validasi *Custom Model*

Epoch	Precision	Recall	mAP50	mAP50-95
25	0.785	0.724	0.813	0.442
50	0.823	0.774	0.848	0.471
75	0.848	0.798	0.867	0.483
100	0.842	0.802	0.866	0.492
150	0.843	0.826	0.882	0.497

4.2 Pengujian Deteksi Asap

Pada tahap ini dilakukan pengujian terhadap kemampuan model dalam mendeteksi asap. Pengujian ini bertujuan untuk mengevaluasi performa model

berdasarkan parameter seperti *confidence score*, akurasi, FPS dan waktu deteksi dalam mendeteksi asap. Bahan yang digunakan untuk menghasilkan asap meliputi kertas, plastik, dan kayu. Pengujian ini menggunakan laptop ASUS TUF Gaming F15 dengan spesifikasi Intel Core i5 generasi ke-10, RAM sebesar 8 GB dan GPU NVIDIA GeForce GTX 1650. Kamera yang digunakan memiliki resolusi 720p dengan kecepatan 30 FPS. Untuk mendapat hasil yang optimal pengujian ini dilakukan dengan keadaan warna *background* yang bertolak belakang dengan warna asap.

4.2.1 Metode Pengujian Deteksi Asap

Pengujian deteksi asap dilakukan dengan dua metode, yaitu:

A. Panci Dibiarkan Terbuka

1. Pembakaran kertas: Mula-mula kertas dibakar untuk menghasilkan bara api. Setelah bara api terbentuk, kertas ditambahkan untuk menghasilkan asap.
2. Pembakaran plastik: Plastik dikumpulkan menjadi satu, lalu dibakar sekaligus di dalam panci untuk menghasilkan asap.
3. Pembakaran kayu: Kayu ditumpuk menyerupai api unggun, kemudian disiram menggunakan spiritus dan dibakar untuk menghasilkan asap.

Metode pengujian panci yang dibiarkan terbuka dapat dilihat pada Gambar 4.1



Gambar 4.1 Metode pengujian panci yang dibiarkan terbuka

B. Panci Ditutup

1. Pembakaran kertas: Kertas dibakar hingga api menyala, kemudian panci ditutup untuk mematikan api. Setelah itu, panci dibuka untuk menghasilkan asap.
2. Pembakaran plastik: Plastik dibakar hingga api menyala, kemudian panci ditutup untuk mematikan api. Setelah itu, panci dibuka untuk menghasilkan asap.
3. Pembakaran kayu: Kayu ditumpuk menyerupai api unggun, kemudian disiram menggunakan spiritus dan dibakar. Setelah itu, panci ditutup untuk mematikan api dan dibuka lagi untuk menghasilkan asap.

Metode pengujian panci yang ditutup dapat dilihat pada Gambar 4.2



Gambar 4.2 Metode panci yang ditutup

4.2.2 Prosedur Pengujian

Berikut merupakan langkah-langkah dalam menguji model untuk mendeteksi asap:

1. Mengambil pembacaan hasil deteksi asap.
2. Mengambil nilai *confidence score* dalam mendeteksi asap.
3. Mengambil FPS dalam mendeteksi asap.
4. Mencatat waktu yang dibutuhkan model dalam mendeteksi asap.
5. Mencatat hasil nilai akurasi dari hasil deteksi asap.

4.2.3 Hasil Pengujian Deteksi Asap

Hasil pengujian diperoleh menggunakan kamera dari laptop yang telah dilengkapi dengan model AI untuk mendeteksi asap secara *real-time*. Hasil pengujian deteksi akan menampilkan keberadaan asap, hasil nilai dari FPS dan *confidence score*. Waktu deteksi juga dicatat menggunakan *stopwatch* untuk mengukur seberapa cepat model dalam mendeteksi asap. Untuk menghitung waktu deteksi pertama-tama *stopwatch* baru akan ditekan ketika asap sudah muncul dan berhenti ketika model pertama kali berhasil mendeteksi asap. Selain itu, pengujian ini juga mengukur akurasi. Perhitungan akurasi dihitung menggunakan rumus 4.1.

$$Akurasi = \left(\frac{\text{Jumlah Asap Terdeteksi}}{\text{Total Pengujian}} \right) \times 100 \quad (4.1)$$

Jumlah asap terdeteksi dihitung ketika model berhasil mendeteksi asap, total data merupakan total keseluruhan pada percobaan yang dilakukan selama pengujian. Model AI yang digunakan pada pengujian ini adalah YOLOv8, yang telah dilatih dengan *epoch* sebanyak 150.

A. Hasil Pengujian Panci Dibiarkan Terbuka

Pengujian ini dilakukan untuk mendeteksi asap yang dihasilkan dari bahan pembakaran kertas, plastik dan kayu menggunakan model yang telah dilatih yaitu YOLOv8. Hasil pengujian menunjukkan bahwa sistem mampu menentukan asap dari bahan pembakaran kertas dengan cukup baik, namun tidak dengan bahan pembakaran plastik dan kayu. Hasil pengujian menggunakan metode panci yang dibiarkan terbuka dapat dilihat pada Tabel 4.2

Tabel 4.2 Hasil pengujian kertas dengan metode panci yang dibiarkan terbuka

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
Kertas	✓	-	0.63	6.27	15.26
	✓	-	0.71	6.23	19.63
	✓	-	0.54	6.25	34.68
	✓	-	0.66	6.31	12.65
	✓	-	0.57	6.35	14.57
	✓	-	0.55	6.19	15.00
	✓	-	0.53	6.27	13.47
	-	✓	-	-	-

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	✓	-	0.58	6.31	48.19
	✓	-	0.54	6.19	10.85
	✓	-	0.60	5.83	14.93
	✓	-	0.58	6.04	13.67
	✓	-	0.51	7.27	10.49
	✓	-	0.54	7.08	12.68
	✓	-	0.57	6.19	31.47
	-	✓	-	-	-
	✓	-	0.62	6.89	17.79
	✓	-	0.61	6.71	15.26
	✓	-	0.59	6.23	23.18
	✓	-	0.58	6.58	20.79
	✓	-	0.50	6.19	18.13
	-	✓	-	-	-
	-	✓	-	-	-
	✓	-	0.55	5.93	15.97
	-	✓	-	-	-

Pada Tabel 4.2 menunjukkan hasil pengujian 25 data dari deteksi asap dengan bahan pembakaran kertas menggunakan metode panci yang dibiarkan terbuka. Perhitungan akurasi dilakukan menggunakan rumus 4.1 sebagai berikut:

$$Akurasi = \left(\frac{20}{25}\right) \times 100 = 80\%$$

Pengujian ini menghasilkan nilai rata-rata akurasi sebesar 80% dengan 20 deteksi benar dan 5 deteksi salah. Rata-rata FPS pada pengujian kertas sebesar 6.37.

Tabel 4.3 Hasil pengujian plastik dengan metode panci yang dibiarkan terbuka

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	-	✓	-	-	-
	✓	-	0.50	6.48	15.31
	✓	-	0.53	6.71	18.01
	✓	-	0.50	6.23	13.97
	-	✓	-	-	-
	-	✓	-	-	-
	-	✓	-	-	-
	-	✓	-	-	-
	✓	-	0.52	5.78	15.31
	✓	-	0.53	6.31	14.56
	-	✓	-	-	-
	-	✓	-	-	-

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	-	✓	-	-	-
	✓	-	0.51	6.08	16.78
	-	✓	-	-	-
	-	✓	-	-	-
	✓	-	0.50	5.91	14.36
	✓	-	0.50	6.31	12.36
	✓	-	0.54	6.28	14.84
	✓	-	0.52	6.32	15.49
	-	✓	-	-	-
	-	✓	-	-	-
	-	✓	-	-	-
	✓	-	0.53	6.51	17.11
	✓	-	0.55	6.72	15.97

Pada Tabel 4.3 menunjukkan hasil pengujian 25 data dari deteksi asap dengan bahan pembakaran plastik menggunakan metode panci yang dibiarkan terbuka. Perhitungan akurasi dilakukan menggunakan rumus 4.1 sebagai berikut:

$$Akurasi = \left(\frac{12}{25}\right) \times 100 = 48\%$$

Pengujian ini menghasilkan nilai rata-rata akurasi sebesar 48% dengan 12 deteksi benar dan 13 deteksi salah. Rata-rata FPS pada pengujian plastik sebesar 6.30.

Tabel 4.4 Hasil pengujian kayu dengan metode panci yang dibiarkan terbuka

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
Kayu	✓	-	0.54	6.71	1.19.36
	✓	-	0.54	6.36	1.53.71
	✓	-	0.51	6.24	1.58.27
	✓	-	0.50	5.89	2.04.35
	-	✓	-	-	-
	-	✓	-	-	-
	-	✓	-	-	-
	✓	-	0.51	6.27	2.06.17
	-	✓	-	-	-
	✓	-	0.52	6.14	1.30.58
	✓	-	0.50	5.92	1.24.17
	✓	-	0.52	5.81	1.86.97
	-	✓	-	-	-
	-	✓	-	-	-
	✓	-	0.54	6.27	1.58.79
✓	-	0.55	6.27	1.71.26	

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	✓	-	0.50	5.84	1.54.89
	✓	-	0.50	6.09	1.21.46
	✓	-	0.53	5.98	1.40.13
	-	✓	-	-	-
	-	✓	-	-	-
	-	✓	-	-	-
	✓	-	0.51	6.29	2.10.02
	-	✓	-	-	-
	-	✓	-	-	-

Pada Tabel 4.4 menunjukkan hasil pengujian 25 data dari deteksi asap dengan bahan pembakaran kayu menggunakan metode panci yang dibiarkan terbuka. Perhitungan akurasi dilakukan menggunakan rumus 4.1 sebagai berikut:

$$Akurasi = \left(\frac{14}{25}\right) \times 100 = 56\%$$

Pengujian ini menghasilkan nilai akurasi rata-rata sebesar 56% dengan 14 deteksi benar dan 11 deteksi salah. Rata-rata FPS pada pengujian kayu sebesar 6.11.

B. Hasil Pengujian Panci Ditutup

Pengujian ini dilakukan untuk mendeteksi asap yang dihasilkan dari bahan pembakaran kertas, plastik dan kayu menggunakan model yang telah dilatih yaitu YOLOv8. Hasil pengujian menunjukkan bahwa sistem mampu menentukan asap dari bahan pembakaran kertas, plastik dan kayu dengan cukup baik. Hasil pengujian menggunakan metode panci ditutup lalu dibuka dapat dilihat pada Tabel 4.5

Tabel 4.5 Hasil pengujian kertas dengan metode panci yang ditutup lalu dibuka

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
Kertas	✓	-	0.66	6.60	3.16
	✓	-	0.73	6.65	3.27
	✓	-	0.58	6.43	4.29
	✓	-	0.54	5.71	4.91
	✓	-	0.54	5.42	2.51
	✓	-	0.59	6.43	3.12
	✓	-	0.62	7.27	3.69
	✓	-	0.60	6.37	3.62
	✓	-	0.60	6.92	3.25
	✓	-	0.50	6.32	4.06

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	✓	-	0.59	5.43	3.73
	✓	-	0.61	6.51	3.08
	✓	-	0.66	6.60	3.87
	✓	-	0.58	6.60	3.74
	✓	-	0.60	6.37	2.68
	-	✓	-	-	-
	✓	-	0.58	6.48	2.96
	✓	-	0.53	6.40	4.10
	✓	-	0.64	5.46	3.89
	✓	-	0.57	6.48	4.02
	✓	-	0.51	6.37	3.20
	✓	-	0.65	6.32	3.53
	✓	-	0.52	6.43	3.30
	-	✓	-	-	-
	✓	-	0.62	7.32	3.69

Pada Tabel 4.5 menunjukkan hasil pengujian 25 data dari deteksi asap dengan bahan pembakaran kertas menggunakan metode panci yang ditutup. Perhitungan akurasi dilakukan menggunakan rumus 4.1 sebagai berikut:

$$Akurasi = \left(\frac{23}{25}\right) \times 100 = 92\%$$

Pengujian ini menghasilkan nilai rata-rata akurasi sebesar 92% dengan 23 deteksi benar dan 2 deteksi salah. Rata-rata FPS pada pengujian kertas sebesar 6.38.

Tabel 4.6 Hasil pengujian plastik dengan metode panci ditutup lalu dibuka

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	✓	-	0.55	6.87	3.06
	✓	-	0.56	6.11	2.77
	✓	-	0.59	6.37	3.93
	✓	-	0.59	6.32	3.64
	✓	-	0.52	6.54	3.90
	-	✓	-	-	-
Plastik	✓	-	0.56	6.43	4.31
	✓	-	0.52	6.87	2.80
	✓	-	0.51	6.43	2.80
	✓	-	0.57	6.73	3.05
	✓	-	0.54	6.06	2.38
	✓	-	0.54	6.16	2.48
	✓	-	0.60	6.32	2.67
	✓	-	0.55	7.35	2.51

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	✓	-	0.56	6.48	3.52
	✓	-	0.56	6.43	2.44
	✓	-	0.55	6.32	3.67
	✓	-	0.55	7.37	2.65
	✓	-	0.60	5.37	2.64
	✓	-	0.60	6.32	2.45
	-	✓	-	-	-
	✓	-	0.56	6.37	2.77
	✓	-	0.55	5.32	3.11
	-	✓	-	-	-
	✓	-	0.54	5.21	2.99

Pada Tabel 4.6 menunjukkan hasil pengujian 25 data dari deteksi asap dengan bahan pembakaran plastik menggunakan metode panci yang ditutup. Perhitungan akurasi dilakukan menggunakan rumus 4.1 sebagai berikut:

$$Akurasi = \left(\frac{22}{25}\right) \times 100 = 88\%$$

Pengujian ini menghasilkan nilai rata-rata akurasi sebesar 88% dengan 22 deteksi benar dan 3 deteksi salah. Rata-rata FPS pada pengujian plastik sebesar 6.33.

Tabel 4.7 Hasil pengujian kayu dengan metode panci ditutup lalu dibuka

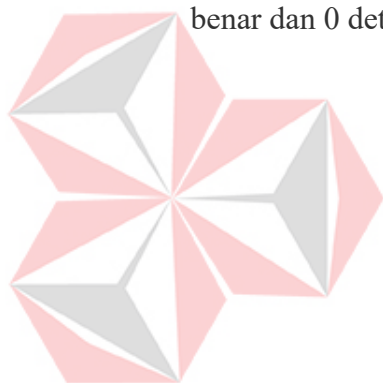
Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
Kayu	✓	-	0.72	6.59	1.89
	✓	-	0.50	6.82	2.70
	✓	-	0.56	6.65	2.19
	✓	-	0.52	7.42	2.17
	✓	-	0.51	6.65	1.85
	✓	-	0.55	6.65	2.35
	✓	-	0.50	6.22	2.03
	✓	-	0.70	6.88	1.96
	✓	-	0.55	6.53	1.85
	✓	-	0.56	6.65	1.93
	✓	-	0.63	6.59	1.75
	✓	-	0.66	6.45	1.90
	✓	-	0.66	6.42	2.26
	✓	-	0.69	6.65	2.51
	✓	-	0.63	6.48	2.23
	✓	-	0.51	5.48	2.07
	✓	-	0.64	7.12	2.22
✓	-	0.65	6.22	2.17	

Jenis Asal/ Bahan Asap	Terdeteksi	Tidak Terdeteksi	Confidence Score	FPS	Waktu Deteksi
	✓	-	0.75	6.32	2.06
	✓	-	0.52	6.97	2.01
	✓	-	0.50	6.45	2.17
	✓	-	0.55	6.48	1.78
	✓	-	0.59	7.42	2.66
	✓	-	0.54	6.65	2.67
	✓	-	0.53	7.09	2.20

Pada Tabel 4.7 menunjukkan hasil pengujian 25 data dari deteksi asap dengan bahan pembakaran kayu menggunakan metode panci yang ditutup. Perhitungan akurasi dilakukan menggunakan rumus 4.1 sebagai berikut:

$$Akurasi = \left(\frac{25}{25}\right) \times 100 = 100\%$$

Pengujian ini menghasilkan nilai rata-rata akurasi sebesar 100% dengan 25 deteksi benar dan 0 deteksi salah. Rata-rata FPS pada pengujian kayu sebesar 6.64.



UNIVERSITAS
Dinamika

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian yang telah dilakukan, berikut merupakan kesimpulan yang dapat diambil:

1. Teknologi *Computer Vision* dapat menjadi solusi alternatif untuk mendeteksi asap. Berdasarkan hasil pengujian, sistem dapat mendeteksi keberadaan asap secara *real-time* dan menampilkan nilai *Frame per Second* (FPS).
2. Hasil rata-rata nilai FPS dari metode panci yang dibiarkan terbuka dan panci yang tertutup adalah sebesar 6.36 yang menunjukkan kemampuan deteksi asap dengan cukup stabil.
3. Pengujian dilakukan dengan dua metode, yaitu dengan metode panci yang dibiarkan terbuka dan metode panci yang ditutup lalu dibuka. Pada bahan pembakaran kertas, plastik dan kayu dilakukan 25 percobaan. Hasil rata-rata akurasi dari kedua metode menunjukkan perbedaan yang signifikan. Hasil rata-rata akurasi menggunakan metode panci yang dibiarkan terbuka yaitu sebesar 61.33% dengan pembakaran kertas, plastik dan kayu. Sedangkan, rata-rata akurasi menggunakan metode panci yang ditutup lalu dibuka yaitu sebesar 93.33% dengan pembakaran kertas, plastik dan kayu. Perbedaan akurasi ini disebabkan oleh perbedaan asap yang dihasilkan. Pada metode panci yang dibiarkan dibuka, api akan terus menyala sehingga asap yang dihasilkan akan lebih sedikit. Sedangkan pada metode panci yang ditutup lalu dibuka, api akan lebih cepat padam dan menghasilkan asap yang lebih banyak.
4. Hasil pengujian waktu deteksi dengan metode panci yang dibiarkan terbuka relatif lama karena menunggu api padam terlebih dahulu, baru asap yang muncul akan lebih banyak. Sedangkan, metode panci yang ditutup lalu dibuka menghasilkan lebih banyak asap dan waktu deteksi yang cepat karena api padam dengan cepat.

5.2 Saran

Dari hasil pengujian dan untuk pengembangan lebih lanjut, berikut beberapa saran yang dapat dipertimbangkan:

1. Disarankan untuk mencari dataset asap yang lebih banyak dan bervariasi, mencakup berbagai kondisi pencahayaan, lingkungan dan berbagai jenis asap agar model dapat lebih optimal.
2. Pada saat implementasi disarankan untuk meletakkan kamera di sudut ruangan agar mencakup area yang luas.
3. Disarankan untuk mengeksplorasi algoritma lain guna membandingkan akurasi dan kinerja sistem untuk deteksi asap.
4. Dapat dikembangkan dengan menambahkan fitur notifikasi atau alarm untuk memberikan peringatan dini kepada pengguna.
5. Dapat dikembangkan dengan menambahkan fitur deteksi api untuk meningkatkan kemampuan sistem dalam mendeteksi potensi kebakaran dengan lebih optimal.



UNIVERSITAS
Dinamika

DAFTAR PUSTAKA

- Fajri, F. N., & Priambodo, W. G. (2023). *Jurnal Kecerdasan Buatan , Komputasi dan Teknologi Informasi Fire and Smoke Object Detection Using Mask R-CNN*. 4(2), 97–102.
- Ferguson, F., Agung, T., Prasetya, E., Nawainetu, E. D., Ilmu, S., Masyarakat, K., & Airlangga, U. (2020). *RISK ASSESSMENT KEBAKARAN DAN PELEDAKAN DI PT XYZ SURABAYA RISK ASSESSMENT OF FIRE AND EXPLOSION Abstract kasus ledakan di pabrik gas PT Aneka Gas Industri (AGI) di Medan Deli . Diduga telah banyak metode risk assessment yang Analysis (FTA), Event Tre*. 4(2), 42–53.
- Hayati, N. J., Singasatia, D., & Muttaqin, M. R. (2023). Object Tracking Menggunakan Algoritma You Only Look Once (YOLO)v8 untuk Menghitung Kendaraan. *Komputa : Jurnal Ilmiah Komputer Dan Informatika*, 12(2), 91–99. <https://doi.org/10.34010/komputa.v12i2.10654>
- Kairos Abinaya Susanto, Darrien Rafael Wijaya, Matthew Owen, Tertius Raya Prasetya, George Maximillian Theodore, Jevant Russell, & Rahmi Yulia Ningsih. (2023). Implementasi Bahasa Python Dalam Menganalisis Pengaruh Rokok Terhadap Risiko Pasien Terkena Penyakit Stroke. *Jurnal Publikasi Teknik Informatika*, 2(2), 48–58. <https://doi.org/10.55606/juhti.v2i2.1722>
- Rahmadhika, M. K., & Thantawi, A. M. (2021). Rancang Bangun Aplikasi Face Recognition Pada Pendekatan CRM Menggunakan Opencv Dan Algoritma Haarcascade. *IKRA-ITH INFORMATIKA: Jurnal Komputer Dan Informatika*, 5(1), 109–118.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-Decem*, 779–788. <https://doi.org/10.1109/CVPR.2016.91>
- Saponara, S., Elhanashi, A., & Gagliardi, A. (2021). Real-time video fire/smoke detection based on CNN in antifire surveillance systems. *Journal of Real-Time Image Processing*, 18(3), 889–900. <https://doi.org/10.1007/s11554-020-01044-0>
- Waworundeng, J. M. S. (2020). Desain Sistem Deteksi Asap dan Api Berbasis Sensor, Mikrokontroler dan IoT. *CogITO Smart Journal*, 6(1), 117–127. <https://doi.org/10.31154/cogito.v6i1.239.117-127>
- Wirajati, & Natha. (2021). Deteksi Api Kebakaran Berbasis Computer Vision Dengan Algoritma YOLO. *Journal of Applied Mechanical ...*, 2, 114–118. <https://scholar.archive.org/work/ahowr3etkrf4rbwj4aedmd6awm/access/wayback/http://ojs.pnb.ac.id/index.php/JAMETECH/article/download/2461/1675>