ANALISIS PERBANDINGAN KINERJA ALGORITMA YOLOV8 HINGGA YOLOV12 UNTUK DETEKSI EMOSI PADA EKSPRESI WAJAH

Disusun Oleh:

N a m a : Rizky Prayogi Reksomulyo
NIM : 20523047

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2025

HALAMAN PENGESAHAN DOSEN PEMBIMBING

ANALISIS PERBANDINGAN KINERJA ALGORITMA YOLOV8 HINGGA YOLOV12 UNTUK DETEKSI EMOSI PADA EKSPRESI WAJAH

TUGAS AKHIR

Disusun Oleh:

N a m a : Rizky Prayogi Reksomulyo

NIM : 20523047

Yogyakarta, 4 Agusuts 2025

Pembimbing,

(Arrie Kurniawardhani, S.Si., M.Kom.)

HALAMAN PENGESAHAN DOSEN PENGUJI

DETEKSI EMOSI MENGGUNAKAN YOLO

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, 4 Agustus 2025

Tim Penguji

Arrie Kurniawardhani, S.Si., M.Kom. _____

Anggota 1

Dr. Raden Teduh Dirgahayu, S.T., M.Sc. _____

Anggota 2

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

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 : Rizky Prayogi Reksomulyo

NIM : 20523047

Tugas akhir dengan judul:

ANALISIS PERBANDINGAN KINERJA ALGORITMA YOLOV8 HINGGA YOLOV12 UNTUK DETEKSI EMOSI PADA EKSPRESI WAJAH

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, 4 Agustus 2025

(Rizky Prayogi Reksomulyo)

HALAMAN PERSEMBAHAN

Bismillahirahmanirrahim,

Syukur alhamdulillah atas izin Allah Subhanahu wa Ta'ala akhirnya saya dapat menyelesaikan tugas akhir ini dengan baik. Terimakasih yang tak terhingga saya ucapkan kepada kedua orang tua yang selalu mendoakan saya dan selalu memberi dukungan kepada saya selama pengerjaan tugas akhir. Dengan laporan inilah salah bentuk wujud terimakasih saya dan berharap semoga mampu membuat bahagia dan bangga. Penulis juga mengucapkan terimakasih sebanyak-banyaknya kepada semua pihak yang telah membantu, mendo'akan, dan mendukung saya selama pengerjaan tugas akhir ini. Semoga Allah Subhanahu wa Ta'ala selalu memberi keberkahan, kesehatan dan rezeki kepada kita semua, Aamiin.

HALAMAN MOTO

“Pain and suffering are always inevitable for a large intelligence and a deep heart.”

(Fyodor Dostoevsky)

“All things that are truly great are at first thought impossible.”

(Friedrich Nietzsche)

KATA PENGANTAR

Assalamu'alaikum Wr.Wb.

Alhamdulillahirabbil'aalamiin, segala puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah memberikan rahmat, hidayah, berkah dan karunia-Nya yang takkan terhitung. Sholawat serta salam senantiasa tercurahkan kepada Nabi Muhammad SAW yang senantiasa mendoakan keselamatan umatnya dan telah membawa umatnya dari zaman jahilliyah menuju zaman islam yang terang benderang dan penuh pengetahuan serta teknologi seperti sekarang, sehingga sehingga saya dapat menyelesaikan tugas akhir ini yang berjudul "ANALISIS PERBANDINGAN KINERJA ALGORITMA YOLOV8 HINGGA YOLOV12 UNTUK DETEKSI EMOSI PADA EKSPRESI WAJAH".

Tugas akhir ini saya susun guna memenuhi syarat untuk memperoleh gelar sarjana di Program Studi Informatika Universitas Islam Indonesia. Dalam penyusunan tugas akhir ini, saya sepenuhnya menyadari bahwa tugas akhir ini tidak dapat terselesaikan tanpa adanya bimbingan, dorongan dan bantuan dari berbagai pihak yang bersifat langsung maupun tidak langsung. Dalam kesempatan ini izinkan saya mengucapkan terima kasih kepada:

1. Allah SWT, berkat limpahan rahmat, karunia dan hidayah-Nya sehingga saya mampu menyusun tugas akhir ini dengan lancar.
2. Prof. Fathul Wahid, S.T., M.Sc., Ph.D. selaku Rektor Universitas Islam Indonesia
3. Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku Ketua Jurusan Informatika.
4. Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. selaku Ketua Program Studi Informatika Program Sarjana
5. Arrie Kurniawardhani, S.Si., M.Kom. selaku dosen pembimbing yang telah memberikan banyak waktunya untuk membimbing, memberi motivasi serta mengarahkan dalam penyusunan skripsi ini dengan penuh semangat dan kesabaran.
6. Seluruh dosen Program Studi Informatika Universitas Islam Indonesia yang telah banyak memberikan ilmu, motivasi dan semangatnya dalam proses beajar di perkuliahan.
7. Kedua orang tua yang saya cintai yang selalu memberikan bantuan motivasi dan doa yang terus mengalir hingga saat ini.
8. Saudara serta kerabat yang selalu mendukung dan memberi nasihat dalam penyusunan skripsi.
9. sahabat-sahabatku yang telah memberi banyak dukungan dalam proses penulisan skripsi ini.

10. Seluruh pihak yang tidak dapat saya sebutkan satu persatu yang telah banyak membantu saya dalam penyusunan skripsi.

Saya menyadari bahwa dalam penyusunan tugas akhir ini masih banyak kekurangan dan kesalahan yang disengaja maupun tidak disengaja maka, izinkan saya mohon maaf sebesar-besarnya atas tindakan tersebut. Semoga dalam kekurangan atau pun kesalahan bisa menjadikan inspirasi bagi generasi-generasi berikutnya, sekian kata pengantar ini dibuat. Bagi siapa pun yang membaca tugas akhir ini semoga mendapatkan manfaat bagi semua pihak.

Wassalamu'alaikum Wr.Wb.

Yogyakarta, 4 Agustus 2025

(Rizky Prayogi Reksomulyo)

INTISARI

Pengenalan emosi semakin menjadi fokus penelitian yang populer karena meningkatnya kesadaran akan peran penting emosi dalam dinamika sosial dan dampaknya di berbagai sektor, sehingga relevan untuk aplikasi seperti pemasaran produk dan dukungan kesehatan mental. Algoritma YOLO (You Only Look Once) dikenal karena kemampuannya dalam mendeteksi objek secara *real-time* dengan cepat dan efektif, dengan studi awal yang menunjukkan potensi YOLO untuk menawarkan akurasi yang lebih baik dibandingkan dengan metode alternatif seperti CNN dan Faster R-CNN. Penelitian ini diarahkan untuk mengimplementasikan dan menguj kinerja serangkaian versi algoritma YOLO mulai dari YOLOv8n hingga YOLOv12n dalam pendeteksian emosi wajah, dengan tujuan utama untuk mengidentifikasi model yang mampu mencapai *mean Average Precision* (mAP) yang paling optimal. Pendekatan metodologis dalam penelitian ini mengandalkan dataset KDEF, yang terdiri dari 1.468 gambar wajah yang mewakili tujuh ekspresi emosional dasar: marah, jijik, takut, senang, netral, sedih, dan terkejut. Dataset ini kemudian dipartisi dengan rasio 80% untuk tujuan pelatihan (1.175 gambar) dan 20% untuk validasi (293 gambar). Proses anotasi atau pelabelan data dibantu oleh *platform* Roboflow. Selanjutnya, model YOLO menjalani pelatihan dengan konfigurasi *hyperparameter* tertentu, yaitu ukuran *batch* 32, *epoch* 100, dan resolusi gambar 640x640. Alur kerja penelitian meliputi beberapa tahap: akuisisi data, pelabelan data, partisi data, konfigurasi *hyperparameter*, pelatihan model, hingga pengujian model, yang juga mencakup uji inferensi menggunakan gambar dari *video* dataset RAVDESS dan gambar yang diambil secara langsung. Temuan dari pengujian yang dilakukan mengkonfirmasi bahwa model YOLOv12n secara konsisten menunjukkan kinerja yang unggul dibandingkan dengan model lainnya. YOLOv12n berhasil mencatat nilai *peak recall* sebesar 0,986 dan mAP50 tertinggi mencapai 0,993, dengan tetap mempertahankan waktu inferensi sebesar 2,9 milidetik. Selain itu, model ini menunjukkan keseimbangan nilai loss antara data pelatihan dan validasi, yang mencerminkan kemampuan generalisasi dan risiko *overfitting* yang rendah. Dalam pengujian inferensi menggunakan video dari RAVDESS, YOLOv12n mampu mengklasifikasikan semua emosi dengan satu kesalahan. Demikian pula, ketika diuji pada gambar khusus yang menampilkan banyak wajah, model ini juga menunjukkan tingkat deteksi yang tepat, meskipun ada satu contoh kesalahan klasifikasi emosi.

Kata kunci: Deteksi Emosi, YOLO, KDEF, mAP, Ekspresi Wajah.

GLOSARIUM

Glosarium memuat daftar kata tertentu yang digunakan dalam laporan dan membutuhkan penjelasan, misalnya kata serapan yang belum lazim digunakan. Urutkan sesuai abjad. Contoh penulisannya seperti di bawah ini:

Batch size	Jumlah sampel data yang diproses oleh model dalam satu kali iterasi selama proses pelatihan.
Epoch	Suatu siklus penuh di mana model machine learning atau deep learning memproses seluruh data pelatihan satu kali secara keseluruhan.
Hyperparameter	Pengaturan awal yang harus ditentukan sebelum melatih model dan dapat memengaruhi cara model belajar dan seberapa baik kinerjanya.
Inference	Proses di mana model yang telah dilatih digunakan untuk melakukan prediksi terhadap data baru yang belum pernah dilihat sebelumnya.
Loss	Nilai yang menunjukkan seberapa jauh hasil prediksi model berbeda dari jawaban yang benar dalam data pelatihan.

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
INTISARI 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 2
1.3 Tujuan Penelitian 3
1.4 Manfaat Penelitian 3
1.4.1 Bagi Penulis 3
1.4.2 Bagi Universitas 3
1.4.3 Bagi Masyarakat 3
1.5 Batasan Penelitian 3
1.6 Metodologi Penelitian 4
1.7 Sistematika penulisan 4
BAB II TINJAUAN PUSTAKA 6
2.1 Emosi 6
2.2 Deteksi Wajah 7
2.3 CNN (Convolutional Neural Networks) 7
2.3.1 Convolution Layer 8
2.3.2 Pooling Layer 9
2.3.3 Fully Connected Layer 9
2.3.4 Activation Function 10
2.4 YOLO 10
2.4.1 YOLOv8 10
2.4.2 YOLOv9 12
2.4.3 YOLOv10 12
2.4.4 YOLOv11 12
2.4.5 YOLOv12 15
2.4.6 Perbandingan YOLOv8 sampai YOLOv12 15
2.5 Confusion Matrix 17
2.6 MaP (*Mean Average Precision*) 18
2.7 Box Loss 18
2.8 Classification Loss 19
2.9 Distribution Focal Loss 19
2.10 Penelitian Sebelumnya 19
BAB III METODOLOGI PENELITIAN 23
3.1 Tahap Pengumpulan Data 23

3.2 Tahap Pelabelan Dataset 25
3.3 Tahap Pembagian Dataset 26
3.4 Tahap Mengatur *Hyperparameter* 26
3.5 Tahap Melatih Model 27
3.6 Tahap Pengujian Model 28
3.7 Tahap Inference Model 29
BAB IV PENELITIAN 30
4.1 Implementasi 30
4.2 Tahap Mengumpulkan data 30
4.3 Tahap Pelabelan Data 34
4.4 Tahap Pembagian dataset 36
4.5 Tahap Mengatur *Hyperparameter* 36
4.6 Tahap Pelatihan 37
4.6.1 Hasil Pelatihan 38
4.6.2 Perbandingan hasil loss 44
4.7 Tahap Pengujian 45
4.7.1 Hasil Pengujian 45
4.7.2 Perbandingan Hasil Pengujian 58
4.8 Tahap Inference 60
4.8.1 Hasil Inference pada Video RAVDESS 60
4.8.2 Hasil Inference pada Image 66
4.8.3 Perbandingan Hasil Inference Pada Satu Wajah dan Banyak Wajah 72
BAB V KESIMPULAN DAN SARAN 76
5.1 Kesimpulan 76
5.2 Saran 76
DAFTAR PUSTAKA 77

DAFTAR TABEL

Tabel 2.1 Novelty Setiap Model YOLO 15
Tabel 2.2 Kajian Literatur 22
Tabel 3.1 Label Emosi 25
Tabel 4.1 Pembagian Dataset 36
Tabel 4.2 Perbandingan Hasil Loss 44
Tabel 4.3 Perbandingan Model 58
Tabel 4.4 Hasil Klasifikasi Ravdess 73
Tabel 4.5 Hasil Klasifikasi pada Multi-Wajah 74

DAFTAR GAMBAR

Gambar 2.1 Contoh Ragam Emosi 6
Gambar 2.2 Arsitektur Dasar CNN 7
Gambar 2.3 Ilustrasi Cara Kerja Convolution Layer 8
Gambar 2.4 Ilustrasi Pooling Layer 9
Gambar 2.5 YOLOv8 Architecture 11
Gambar 2.6 YOLOv11 Architecture 14
Gambar 3.1 Flowchart 23
Gambar 3.2 Tujuh Ekspresi pada KDEF 24
Gambar 3.3 Tujuh Ekspresi pada KDEF 24
Gambar 3.4 Proses Pelabelan dengan Roboflow 26
Gambar 3.5 Data Pelatihan 28
Gambar 3.6 Data Validasi 28
Gambar 4.1 Kode Pengambilan Data 33
Gambar 4.2 Hasil dari Proses Pengumpulan data 34
Gambar 4.3 Data Sebelum Dilabeli 34
Gambar 4.4 Data Setelah Dilabeli 35
Gambar 4.5 Hasil dari Proses Pelabelan dengan Roboflow 35
Gambar 4.6 Kode Mengatur Hyperparameter 36
Gambar 4.7 Kode File data.yaml 37
Gambar 4.8 Kode Pelatihan Model YOLOv11n 37
Gambar 4.9 Grafik Loss YOLOv12n 38
Gambar 4.10 Grafik Loss YOLOv11n 39
Gambar 4.11 Grafik Loss YOLOv10n 41
Gambar 4.12 Grafik Loss YOLOv9t 42
Gambar 4.13 Grafik Loss YOLOv8n 43
Gambar 4.14 Hasil Pengujian YOLOv12n 45
Gambar 4.15 Confusion Matrix pada YOLOv12n 46
Gambar 4.16 Contoh Kesalahan Prediksi pada YOLOv12n 47
Gambar 4.17 Hasil Pengujian YOLOv11n 47
Gambar 4.18 Confusion Matrix pada YOLOv11n 49
Gambar 4.19 Contoh Kesalahan Prediksi pada YOLOv11n 50
Gambar 4.20 Hasil Pengujian YOLOv10n 50
Gambar 4.21 Confusion Matrix pada YOLOv10n 51

Gambar 4.22 Contoh Kesalahan Prediksi pada YOLOv10n 52
Gambar 4.23 Hasil Pengujian YOLOv9t 52
Gambar 4.24 Confusion Matrix pada YOLOv9t 53
Gambar 4.25 Contoh Kesalahan Prediksi pada YOLOv9t 55
Gambar 4.26 Hasil Pengujian YOLOv8n 55
Gambar 4.27 Confusion Matrix pada YOLOv8n 57
Gambar 4.28 Contoh Kesalahan Prediksi pada YOLOv8n 58
Gambar 4.29 Hasil Inference YOLOv12n pada Gambar Video Ravdess 60
Gambar 4.30 Hasil Inference YOLOv11n pada Gambar Video Ravdess 61
Gambar 4.31 Hasil Inference YOLOv10n pada Gambar Video Ravdess 62
Gambar 4.32 Hasil Inference YOLOv9t pada Gambar Video Ravdess 63
Gambar 4.33 Hasil Inference YOLOv8n pada Gambar Video Ravdess 64
Gambar 4.34 Hasil Inference YOLOv12n pada Image 66
Gambar 4.35 Hasil Inference YOLOv11n pada Image 67
Gambar 4.36 Hasil Inference YOLOv10n pada Image 68
Gambar 4.37 Hasil Inference YOLOv9t pada Image 69
Gambar 4.38 Hasil *Inference* YOLOv8n pada *Image* 70

BAB I
PENDAHULUAN

1.1 Latar Belakang

Dalam beberapa dekade terakhir, deteksi emosi telah menjadi bidang penelitian yang berkembang pesat dan menarik. Emosi memainkan peran penting dalam interaksi manusia sehari-hari dan dapat memengaruhi keputusan, perilaku, serta komunikasi antar individu. Emosi merupakan keadaan kognitif dan fisiologis yang dinamis yang berkembang sebagai respons terhadap stimulus seperti pengalaman, pikiran, atau interaksi dengan orang lain. Hal ini mencakup pengalaman subjektif, proses kognitif, pengaruh perilaku, respons fisiologis, dan komunikasi. Oleh karena itu, pengenalan emosi sangat penting dalam bidang aplikasi seperti pemasaran, interaksi manusia-komputer, kesehatan, pemantauan kesehatan mental, dan keamanan (KAMBLE & Sengupta, 2023).

Aplikasi pemasaran dapat menggunakan pengenalan emosi wajah manusia untuk melacak reaksi konsumen terhadap produk atau iklan. Hal ini dapat membantu perusahaan memahami preferensi konsumen dengan lebih baik dan menyesuaikan upaya pemasaran mereka sesuai dengan itu. Aplikasi kesehatan dapat menggunakan pengenalan emosi wajah manusia untuk memantau pasien terhadap tanda-tanda masalah atau melacak kemajuan terapi. Hal ini dapat membantu tenaga medis memberikan perawatan yang lebih personal dan efektif. Secara keseluruhan, deteksi emosi wajah manusia adalah teknologi yang sangat canggih yang dapat meningkatkan berbagai aspek kehidupan kita, termasuk pemasaran, kesehatan, dan lainnya (Hasan, Lazem, Alkhafaji, & Taher, 2023).

Deteksi emosi wajah adalah analisis perubahan signifikan pada fitur wajah, termasuk bibir, alis, dan mata. Perubahan ini disebabkan oleh kontraksi otot wajah sebagai respons terhadap kondisi emosional. Dengan mengidentifikasi variasi pada fitur wajah, seseorang dapat menyimpulkan suasana hati yang tergambar dalam gambar wajah (Sharma, Singh, Rana, & Kumar, 2023).

YOLO (You Only Look Once) adalah algoritma deteksi objek real-time yang cepat dan efisien, yang memproses gambar dalam satu kali proses, berbeda dengan metode tradisional yang memindai gambar berulang kali (Redmon, Divvala, Girshick, & Farhadi, 2015). YOLO menganggap deteksi objek sebagai masalah regresi, memprediksi koordinat *bounding box* dan probabilitas kelas secara bersamaan. Gambar dibagi menjadi *grid*, di mana setiap sel grid

memprediksi beberapa kotak batas dan memberikan *confident score* yang menunjukkan kemungkinan adanya objek. Untuk memperbaiki deteksi, YOLO menerapkan *Non-Maximum Suppression* (NMS) untuk menghapus *bounding box* yang menumpul dan mempertahankan *confident score* tertinggi(Rajasekhar, 2024).

Berdasarkan penelitian sebelumnya, YOLOv8 telah diterapkan untuk pengenalan ekspresi wajah. Berbeda dengan model klasifikasi tradisional, YOLOv8 menawarkan pendekatan deteksi objek secara real-time yang disesuaikan untuk mengklasifikasikan ekspresi wajah ke dalam dua kategori: “antusias” dan “tidak antusias.” Menggunakan dataset FER2013 yang dimodifikasi, studi ini membandingkan empat varian YOLOv8: YOLOv8n, YOLOv8s, YOLOv8m, dan YOLOv8l. Evaluasi menunjukkan bahwa YOLOv8n mencapai akurasi tertinggi sebesar 95,3% dalam mendeteksi ekspresi antusias, sementara YOLOv8m berkinerja terbaik untuk ekspresi tidak antusias dengan akurasi 94,1% (Salma & Hidayat, 2024).

Meskipun berbagai metode telah diterapkan dalam deteksi emosi wajah, sebagian besar studi sebelumnya mengandalkan model konvensional atau membandingkan kinerja menggunakan hanya satu atau dua varian YOLO. Masih terdapat kekurangan studi komprehensif yang mengevaluasi kinerja model YOLO terbaru, khususnya dari YOLOv8 hingga YOLOv12, untuk tugas deteksi dan klasifikasi emosi wajah. Setiap versi YOLO memperkenalkan pembaruan arsitektur dan optimasi yang dapat secara signifikan memengaruhi akurasi deteksi, kecepatan inferensi, dan efisiensi komputasi, terutama dalam kondisi yang melibatkan variasi pencahayaan, pose, dan jumlah wajah yang ada. Oleh karena itu, penelitian ini bertujuan untuk membandingkan kinerja YOLOv8, YOLOv9, YOLOv10, YOLOv11, dan YOLOv12 dalam deteksi emosi wajah, dengan tujuan mengidentifikasi model yang menawarkan keseimbangan terbaik antara akurasi deteksi, kecepatan, dan efisiensi sumber daya untuk implementasi *real-time*.

1.2 Rumusan Masalah

Dari latar belakang, diketahui bahwa deteksi emosi adalah bidang penting dan algoritma YOLO menunjukkan performa yang unggul dalam tugas deteksi objek secara real-time. Penelitian sebelumnya telah membandingkan berbagai versi YOLO untuk tugas spesifik dan menunjukkan adanya perbedaan kinerja dalam hal akurasi, efisiensi model dan kecepatan. Oleh karena itu, muncul pertanyaan tentang bagaimana perbandingan kinerja YOLOv8 sampai YOLOv12 jika diterapkan pada deteksi emosi wajah.

1.3 Tujuan Penelitian

Membandingkan kinerja serangkaian model deteksi objek menggunakan algoritma YOLOv8, YOLOv9, YOLOv10, YOLOv11, dan YOLOv12 untuk mengenali berbagai ekspresi emosi pada wajah manusia.

1.4 Manfaat Penelitian

1.4.1 Bagi Penulis

a. Penulis dapat menerapkan ilmu pengetahuan yang diperoleh selama perkuliahan dalam penelitian ini.
b. Menambah pengetahuan dalam bidang penelitian.
c. Sebagai syarat dalam menyelesaikan gelar Strata satu pada jurusan Informatika Fakultas Teknologi Industri di Universitas Islam Indonesia.

1.4.2 Bagi Universitas

a. Memperbanyak Penelitian dari jurusan Informatika Fakultas Teknologi Informasi dalam bidang Sains Data.
b. Menjadi tolak ukur bagi universitas dalam menentukan kelulusan dan kemampuan penulis dalam menerapkan ilmu pengetahuan yang sudah didapat selama masa Pendidikan di universitas.

1.4.3 Bagi Masyarakat

a. Berguna dalam hal keilmuan karena dapat memberikan pengetahuan kepada masyarakat umum.

1.5 Batasan Penelitian

a. Menggunakan algoritma YOLO model *nano*
b. Data yang diolah berupa gambar.
c. Dataset yang digunakan diambil dari *website* dengan nama " https://www.kdef.se/".
d. Hanya menggunakan dataset dengan *angle straight, left half,* dan *right half.*
e. *Hyperparameter* yang diatur hanya *batch, epoch* dan *image sized.*
f. *Inference* menggunakan video dari RAVDESS dan image hasil dari foto menggunakan *handphone.*

1.6 Metodologi Penelitian

Langkah-langkah yang diterapkan dalam penelitian ini agar mencapai tujuan yang diinginkan adalah sebagai berikut:

a. Tahap Pengumpulan Data

Pengumpulan data diperlukan untuk mendukung pelaksanaan penelitian. Data ini yang nantinya akan dijadikan sebagai bahan untuk pengembangan program pendeteksian emosi. Data tersebut akan digunakan untuk pelatihan dan validasi.

b. Tahap Pelabelan Dataset

Tahapan yang memberikan sebuah informasi berupa nama class pada dataset. Tahap ini memiliki tujuan untuk memberikan sebuah tanda pada tiap objek yang telah di tetapkan sebelum dataset melalui tahap pembagian dataset.

c. Tahap Pembagian Dataset

Tahapan untuk mengatur komposisi dataset menjadi data pelatihan dan validasi sebelum melalui tahap pelatihan.

d. Tahap Mengatur *Hyperparameter*

Tahapan yang dilakukan untuk meningkatkan kinerja model. Pengembang dapat menemukan kombinasi nilai yang terbaik dengan mengatur *hyperparameter* secara optimal.

e. Melatih Model

Tahapan dimana model dilatih sesuai dengan *hyperparameter* yang telah di atur sebelumnya. Dalam proses ini, model belajar mengenali pola dan hubungan dalam data yang diberikan untuk meningkatkan kemampuan untuk menyelesaikan tugas.

f. Menguji Model

Tahapan ini bertujuan untuk memonitoring kinerja model apakah sudah mencapai tujuan penelitian. Yaitu mendeteksi emosi pada sebuah *image*.

1.7 Sistematika penulisan

Sistematika penulisan berguna untuk mengetahui isi dan mempermudah pembaca dalam memahami penelitian ini, adapun sistematika penulisan adalah sebagai berikut:

BAB I PENDAHULUAN

Bagian ini terdiri dari latar belakang masalah yang menjadi dasar penelitian tentang pentingnya deteksi emosi pada wajah dan Solusi untuk menyelesaikan masalah tersebut.

Berdasarkan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Pada Bab ini berisi tentang review dari penelitian terkait, teori-teori dan konsep yang berhubungan dengan penelitian yang akan dilakukan untuk mendukung dalam pemecahan masalah.

BAB III METODOLOGI

Pada bagian ini akan menjelaskan Langkah-langkah tentang yang digunakan untuk mendapatkan data dan metode algoritma yang digunakan untuk pengembangan model yang ingin dibuat.

BAB IV HASIL DAN PEMBAHASAN

Pada Bab ini berisi tentang pembahasan terhadap hasil pengumpulan, pengolahan dan pengujain yang telah dilakukan.

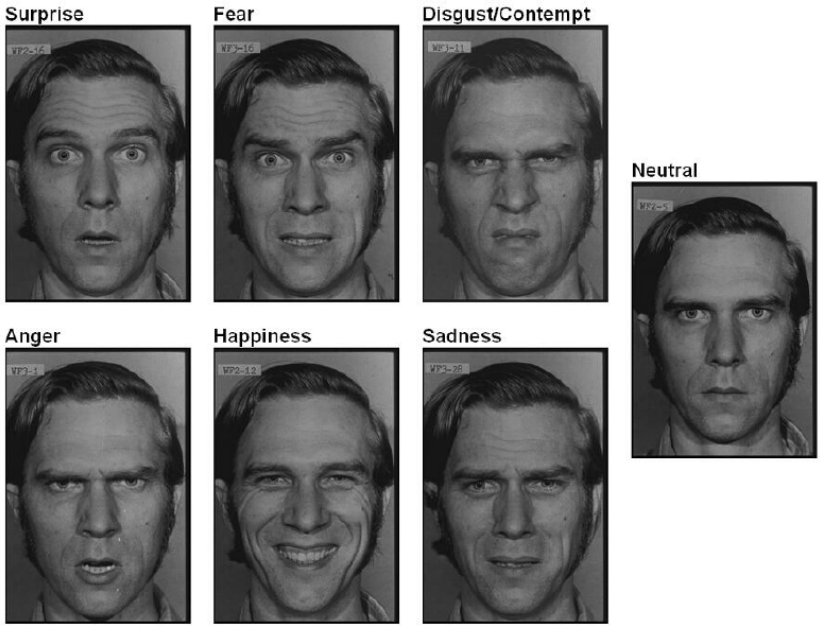
BAB V KESIMPULAN DAN SARAN

Pada Bab ini berisi tentang kesimpulan yang diperoleh dari hasil penelitian yang telah dilakukan serta saran berdasarkan dari hasil penelitian tersebut sehingga dapat menjadi sebuah masukan untuk penelitianpenelitian kedepannya.

BAB II
TINJAUAN PUSTAKA

2.1 Emosi

Emosi adalah kondisi kognitif dan fisiologis yang dinamis yang berkembang sebagai reaksi terhadap masukan, seperti pengalaman, pikiran, atau interaksi dengan orang lain. Hal ini mencakup pengalaman subjektif, proses kognitif, pengaruh perilaku, respons fisiologis, dan komunikasi (KAMBLE & Sengupta, 2023). Emosi dapat bersifat positif atau negatif. ada beberapa emosi dasar yang secara umum diakui di seluruh budaya seperti bahagia, sedih, marah, takut, jijik, terkejut beserta wajah neutral (Keightley, Chiew, Winocur, & Grady, 2007). Emosi dapat dilihat melalui ekspresi wajah dengan beberapa Fitur wajah yang digunakan seperti bibir, alis, dan mata dapat berubah disebabkan oleh emosi dalam aktivitas otot wajah (Sharma, Singh, Rana, & Kumar, 2023).Contoh emosi dapat dilihat pada Gambar 2.1.



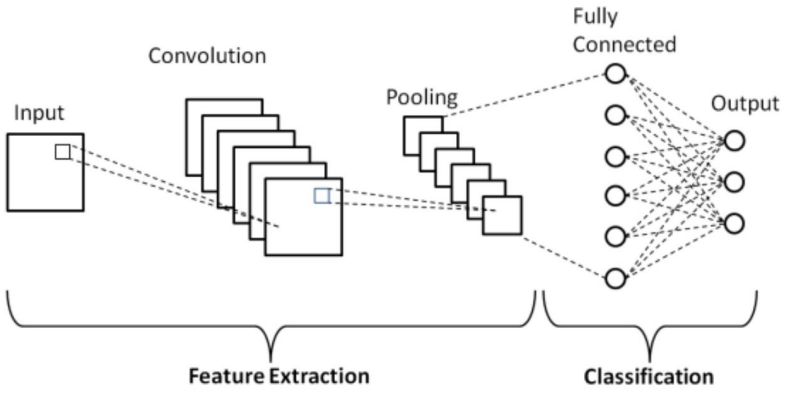
Gambar 2.1 Contoh Ragam Emosi
(Fabri, 2014)

2.2 Deteksi Wajah

Deteksi wajah adalah subdomain dari deteksi objek yang mengidentifikasi wajah manusia dalam bentuk gambar. Deteksi wajah berfungsi sebagai dasar bagi banyak algoritma analisis wajah. Ini termasuk penyelarasan wajah, pelacakan pose kepala, dan bahkan pengenalan jenis kelamin dan usia. Pada dasarnya, hal ini memungkinkan komputer untuk melakukan tugas-tugas yang dapat dilakukan manusia dengan mudah, seperti mengenali wajah dalam berbagai pose, kondisi pencahayaan, dan ekspresi. Mendeteksi wajah dalam gambar bisa menjadi tantangan karena adanya variasi dalam pose, pencahayaan, dan ekspresi. Tujuannya adalah untuk menentukan apakah ada wajah dalam gambar yang berubah-ubah, yang membutuhkan algoritma canggih untuk menangani variasi ini secara efektif. Metodologi yang digunakan dalam pendeteksian wajah telah berkembang secara signifikan, memanfaatkan pembelajaran mendalam dan pembelajaran mesin untuk meningkatkan akurasi dan efisiensi (Chaitra, M., & Dr. M.).

2.3 CNN (Convolutional Neural Networks)

CNN (*Convolutional Neural Networks*) adalah jenis jaringan saraf tiruan yang terinspirasi oleh proses biologis otak manusia yang merupakan jenis arsitektur *deep learning* yang banyak digunakan untuk pengenalan dan pemrosesan gambar pada tingkat piksel, sehingga ideal untuk deteksi objek penting dan tugas-tugas visi komputer (R. & P., 2023). Struktur dasar CNN mencakup lapisan konvolusi, lapisan *pooling*, dan lapisan *fully connected* yang ditumpuk bersama untuk membentuk arsitektur jaringan seperti pada Gambar 2.2



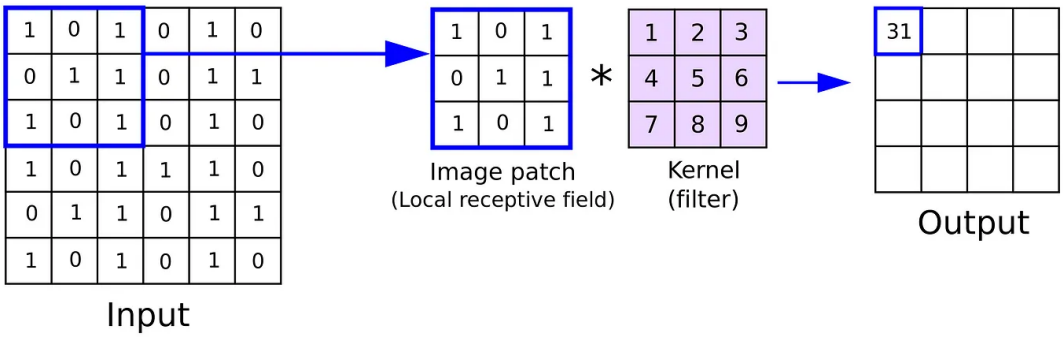
Gambar 2.2 Arsitektur Dasar CNN

Sumber: (Phung & Rhee, 2019)

CNN menggunakan teknik seperti konvolusi dan *downsampling* untuk mengubah data input lapisan demi lapisan, menghasilkan skor kelas untuk tugas-tugas seperti regresi dan klasifikasi (O'Shea & Nash, 2015).

2.3.1 Convolution Layer

Lapisan konvolusi sangat penting dalam CNN (*Convolutional Neural Networks*) untuk ekstraksi fitur dari data *input*. Lapisan ini menerapkan *filter* yang dapat dipelajari di seluruh *input*, melakukan operasi konvolusi untuk menghasilkan peta fitur yang menyoroti berbagai aspek data seperti tepi dan tekstur. Beberapa filter menghasilkan peta fitur yang berbeda, yang ditumpuk bersama untuk membentuk *volume output*.



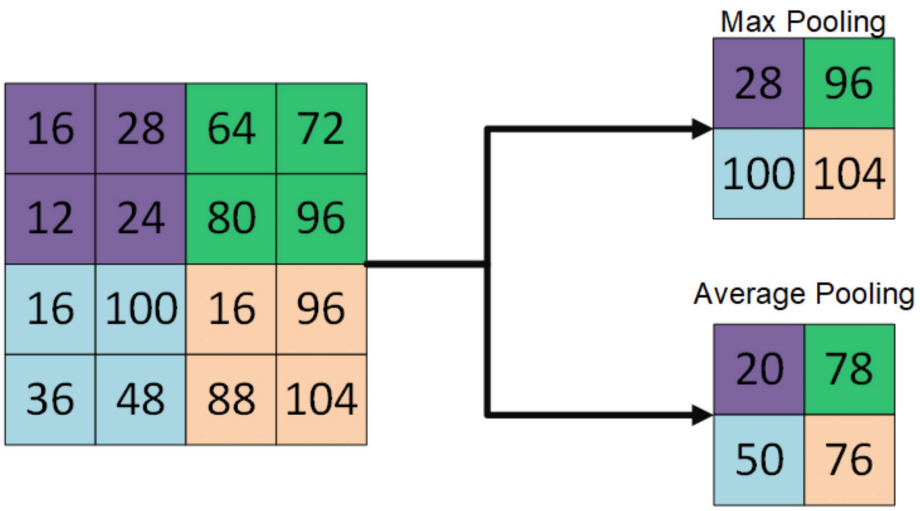
Gambar 2.3 Ilustrasi Cara Kerja Convolution Layer

Sumber: Medium.com

Pada Gambar 2.3 dari *grid* input dipilih *patch* yang lebih kecil. *Patch* ini dikonvolusi dengan *filter* dengan mengalikan nilai yang sesuai dan menjumlahkan hasilnya. Jumlah ini menjadi nilai tunggal dalam jaringan *output*. *Filter* bergerak melintasi *grid input*, mengulangi proses ini untuk mengisi *grid output*, sehingga menciptakan peta fitur. Proses ini memungkinkan jaringan untuk secara otomatis mempelajari dan mendeteksi pola, membuat lapisan konvolusi sangat efektif untuk pemrosesan gambar dan *video* (O'Shea & Nash, 2015)

2.3.2 Pooling Layer

Pooling layer dalam CNN (*Convolutional Neural Networks*) bertujuan untuk mengurangi dimensi representasi data, yang membantu mengurangi jumlah parameter dan kompleksitas komputasi model. Lapisan penyatuan beroperasi pada setiap peta aktivasi dalam input menggunakan fungsi seperti *max-pooling* dan *average-pooling*, di mana *max-pooling kernel* bergeser di atas input dan memilih nilai maksimum di setiap area dan *average-pooling kernel* bergeser di atas *input* dan memilih nilai rata-rata di setiap area (O'Shea & Nash, 2015) seperti pada Gambar 2.4



Gambar 2.4 Ilustrasi Pooling Layer

Sumber: (Debelee, Kebede, Schwenker, & Matewos, 2020)

2.3.3 Fully Connected Layer

Fully connected layer dalam CNN (*Convolutional Neural Networks*) terdiri dari neuron yang terhubung langsung dengan neuron di lapisan yang berdekatan, mirip dengan jaringan saraf tiruan tradisional. Neuron pada lapisan ini menerima input dari semua neuron pada lapisan sebelumnya, sehingga memungkinkan hubungan yang kompleks untuk dipelajari di antara fitur-fiturnya. Ini membantu dalam menangkap fitur tingkat tinggi dengan mempertimbangkan interaksi antara semua neuron di lapisan sebelumnya.

Lapisan yang terhubung sepenuhnya sangat penting untuk membuat prediksi akhir berdasarkan fitur yang dipelajari yang diekstraksi oleh lapisan *convolutional* dan *pooling*. Lapisan ini memainkan peran penting dalam keseluruhan arsitektur CNN, memungkinkan

jaringan untuk belajar dan membuat keputusan berdasarkan fitur yang diekstraksi dari data *input* (O'Shea & Nash, 2015).

2.3.4 Activation Function

Fungsi aktivasi adalah komponen penting dalam jaringan syaraf tiruan yang memperkenalkan non-linearitas pada model. Fungsi ini membantu jaringan mempelajari pola-pola kompleks dalam data dengan mengubah sinyal input menjadi sinyal *output* (Liu & Hui, 2023). Fungsi aktivasi yang umum termasuk SiLU(*Sigmoid Linear Unit*), ReLU(*Rectified Linear Unit*), *Sigmoid*, *Tanh* (*Hyperbolic Tangent*), dan *Softmax*.

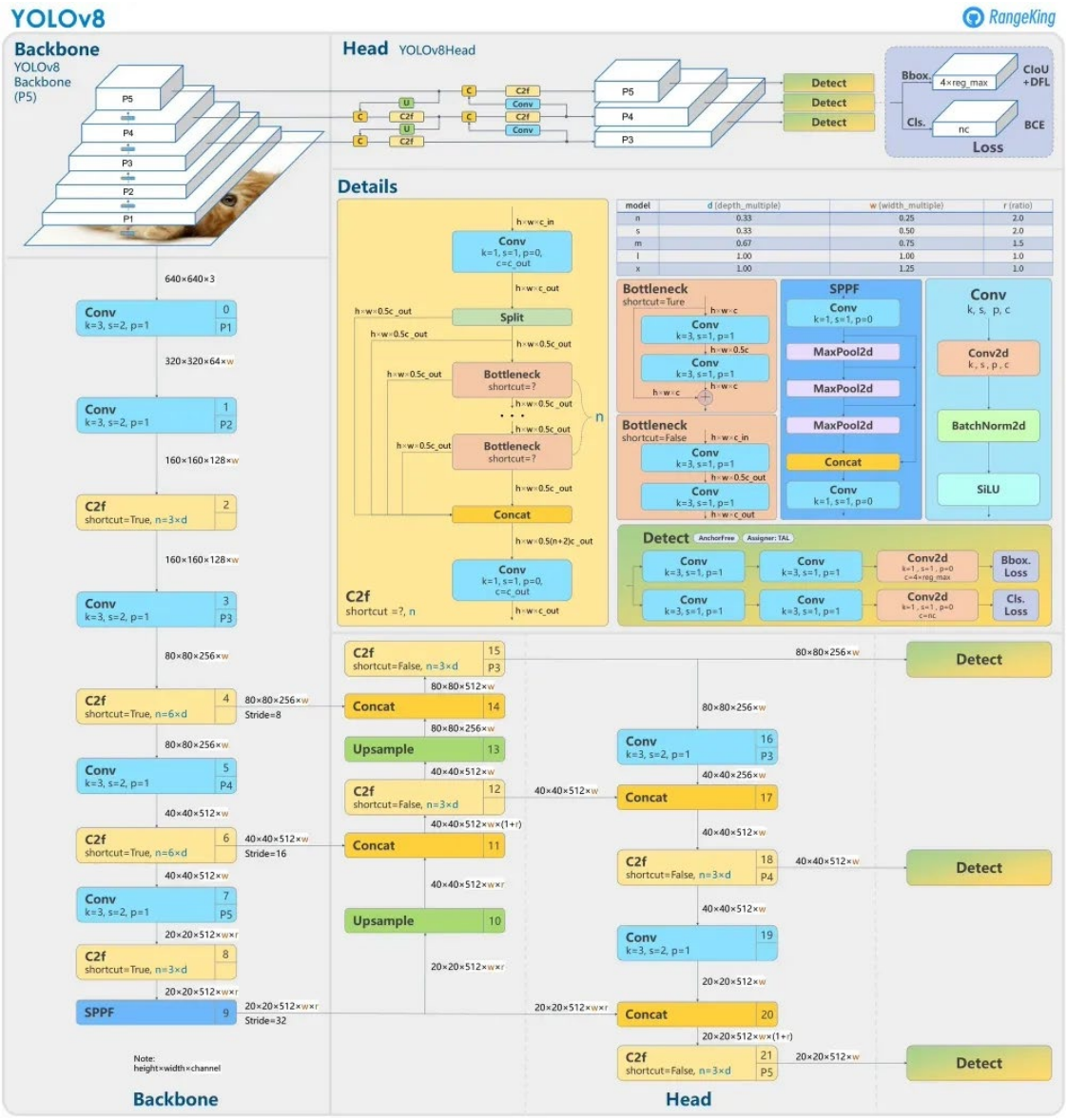
2.4 YOLO

YOLO(*You Only Look Once*) merupakan metode deteksi objek *real-time* yang dirancang untuk mengidentifikasi objek dengan cepat dan akurat dalam gambar atau *video*. YOLO bekerja dengan memproses seluruh gambar menggunakan jaringan CNN(*Convolutional Neural Networks*). Pendekatan ini memungkinkannya menghasilkan beberapa *bounding box* di sekitar objek yang terdeteksi, bersama dengan label kelas dan *confident score* untuk setiap kotak. Ini berarti daripada memindai gambar beberapa kali, YOLO hanya melihat gambar sekali saja, sehingga lebih cepat daripada metode tradisional lainnya (Redmon, Divvala, Girshick, & Farhadi, 2015).

2.4.1 YOLOv8

YOLOv8 diluncurkan oleh Ultralytics pada tanggal 10 Januari 2023, menawarkan metode deteksi *anchor-free* untuk menangani tugas deteksi objek, klasifikasi, dan regresi secara terpisah. Struktur ini memungkinkan setiap cabang untuk berkonsentrasi pada fungsinya masing-masing yang berkontribusi pada kinerja model yang lebih baik secara keseluruhan dan memperbaiki kinerja pada objek kecil dengan menyederhanakan arsitektur model. YOLOv8 tidak hanya dapat mendeteksi objek, tetapi juga dapat menggunakan berbagai fitur. Ini termasuk klasifikasi gambar, estimasi pose, *segmentation instance*, dan OOB(*Oriented Bounding Boxes*) (Jocher, Chaurasia, & Qiu, Ultralytics YOLOv8, 2023). Salah satu fitur utamanya adalah *backbone* yang mirip dengan YOLOv5 dan penyesuaian di CSPLayer. Sekarang dikenal sebagai modul C2f(*Cross-stage partial bottleneck with two convolutions*), yang menggabungkan fitur tingkat tinggi dengan informasi kontekstual untuk meningkatkan akurasi deteksi (Jegham, Abdelatti, & Koh, 2025). SPPF(*Spatial Pyramid Pooling Fast*) pada

bagian *neck*, berfungsi untuk memberikan representasi multi-skala dari peta fitur. SPPF memungkinkan model untuk menangkap fitur pada berbagai tingkat abstraksi dengan menggabungkan pada skala yang berbeda. Untuk melihat keseluruhan aritekturnya dapat dilihat pada Gambar 2.5. (Hidayatullah, Syakranib, Sholahuddin, Gelar, & Tubagus, 2025).



Gambar 2.5 YOLOv8 Architecture

(Surwade, Gundlapalli, Hussain, & Das, 2023)

2.4.2 YOLOv9

YOLOv9 dikembangkan oleh Chien-Yao Wang, I-Hau Yeh, dan Hong-Yuan Mark Liao yang menggunakan *Information Bottleneck Principle* dan *Reversible Functions* untuk meningkatkan deteksi objek, memastikan penyebaran gradien yang efisien, dan konvergensi model yang lebih baik. Fungsi yang dapat dibalik memungkinkan penyimpanan informasi tanpa kehilangan, yang menguntungkan model ringan yang rentan terhadap *underparameterization*. PGI (*Programable Gradient Information)* adalah kemajuan besar karena dapat menyesuaikan aliran gradien secara dinamis selama pelatihan dan memprioritaskan gradien informatif untuk menghindari kehilangan fitur. Selain itu, YOLOv9 mengintegrasikan GELAN(*Gradient Enhanced Lightweight Architecture Network*), yang mengoptimalkan jalur komputasi untuk memungkinkan parameter yang lebih baik, inferensi berkecepatan tinggi, dan deteksi yang lebih akurat (Wang, Yeh, & Liao, 2024).

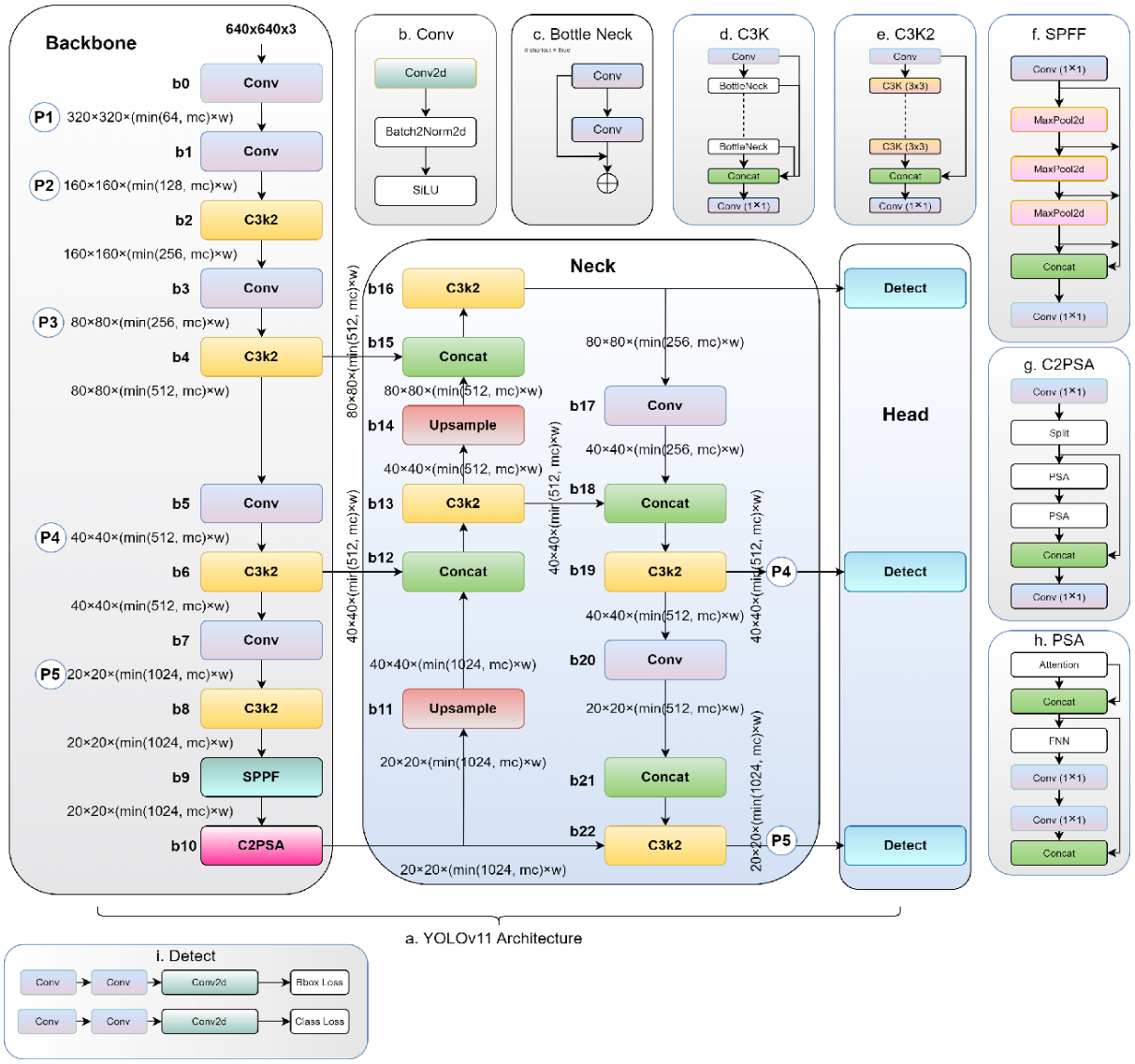
2.4.3 YOLOv10

YOLOv10 dikembangkan oleh peneliti dari Tsinghua University, membawa banyak peningkatan besar, termasuk peningkatan aliran gradien dan efisiensi komputasi melalui pengembangan *backbone* CSPNet(*Cross Stage Parsial Network*). Pendeteksian objek yang lebih baik pada berbagai ukuran dimungkinkan oleh PAN(*Path Aggregation Network*), yang meningkatkan fusi fitur multi-skala. Selama pelatihan, *one-to-many head* menghasilkan banyak prediksi per objek, meningkatkan akurasi pembelajaran. Untuk inferensi, YOLOv10 menggantikan NMS (*Non-Maximum Suppression*) dengan *one-to-one head*, yang menghasilkan prediksi terbaik per objek, mengurangi latensi dan waktu pasca-pemrosesan. Selain itu, Pelatihan NMS-Free mengoptimalkan kecepatan dan akurasi dengan memanfaatkan penugasan ganda. *head* klasifikasi yang ringan, SCDown (*Spatial-Channel Downsampling*), *large kernel convolution*, dan modul *parsial self-attention* adalah peningkatan efisiensi yang signifikan tanpa meningkatkan biaya komputasi. YOLOv10 tersedia dalam lima skala, dari *nano* hingga *extra large*, untuk memenuhi berbagai tujuan (Wang, et al., 2024).

2.4.4 YOLOv11

YOLOv11 dikembangkan oleh ultarlytics pada tahun 2024 yang memperkenalkan peningkatan signifikan dalam hal akurasi, kecepatan, dan efisiensi komputasi. Berlandaskan pada kemajuan signifikan dari versi YOLO sebelumnya, YOLO11 menghadirkan perbaikan signifikan dalam arsitektur dan metode pelatihan, menjadikannya pilihan yang serbaguna untuk

berbagai tugas pengolahan citra komputer (Jocher & Qiu, Ultralytics YOLO11, 2024). YOLOv11 menggabungkan teknik ekstraksi fitur dan *attention mechanisms*, sehingga sangat efektif untuk berbagai tugas visi komputer seperti deteksi objek, segmentasi contoh, estimasi pose, dan deteksi objek berorientasi (Khanam & Hussain, 2024). YOLOv11 mempertahankan struktur tiga komponen tradisional yang terdiri dari *backbone*, *neck*, dan *head*. *Backbone* bertanggung jawab untuk ekstraksi fitur dan menggunakan blok C3k2(*Cross Stage Partial with kernel size 2*), pengganti yang dioptimalkan untuk blok C2f(*Cross-stage partial bottleneck with two convolutions*) di YOLOv8, memberikan pemrosesan yang lebih cepat dan lebih efisien secara parameter. Selain itu, modul SPFF(*Spatial Pyramid Pooling - Fast*) dipertahankan untuk meningkatkan ekstraksi fitur multi-skala, sedangkan C2PSA(*Convolutional block with Parallel Spatial Attention*) yang baru diperkenalkan meningkatkan *attention mechanisms*, yang memungkinkan model untuk fokus pada area gambar yang penting. Komponen *neck* menyempurnakan dan menggabungkan fitur pada skala yang berbeda melalui lapisan *upsampling* dan penggabungan, mengintegrasikan fitur multi-skala sekaligus meningkatkan akurasi deteksi. Komponen *head*, yang bertanggung jawab untuk prediksi akhir, menggabungkan blok C3k2 (*Cross Stage Partial with kernel size 2*) dan lapisan CBS (Convolution-BatchNorm-SiLU), lebih jelasnya dapat dilihat pada Gambar 2.6.



Gambar 2.6 YOLOv11 Architecture

Sumber: (Rasheed & Zarkoosh, 2024)

YOLOv11 juga menawarkan versi khusus yang dioptimalkan untuk mendeteksi objek dengan ukuran berbeda. Dibandingkan dengan model YOLO sebelumnya, model ini mencapai mAP(*mean Average Precision*) yang lebih tinggi sekaligus mengurangi parameter hingga 22%, menjadikannya solusi yang kuat untuk aplikasi deteksi secara *real-time* (Rasheed & Zarkoosh, 2024).

2.4.5 YOLOv12

YOLOv12 merupakan seri yolo yang dibuat pada tahun 2025 oleh Tian et. al yang menggunakan attention mechanism yang meningkatkan kecepatan dan akurasi. YOLOv12 melanjutkan tren dengan menawarkan lima model yang dapat diskalakan (*Nano* hingga *Extra Large*), yang memungkinkan untuk disesuaikan untuk berbagai aplikasi seperti deteksi objek, segmentasi contoh, dan OBB. Modul A2(*Area Attention*), yang mempertahankan bidang reseptif yang besar sekaligus mengurangi kompleksitas komputasi secara signifikan, memungkinkan model untuk meningkatkan kecepatan tanpa mengorbankan bidang reseptif. Selain itu, YOLOv12 memiliki fitur R-ELAN(*Residual Efficient Layer Aggregation Networks*), yang, melalui desain residu tingkat blok dan agregasi fitur yang dioptimalkan, meningkatkan stabilitas pelatihan dan konvergensi model. Selain itu, YOLOv12 menggunakan *FlashAttention* untuk mengurangi *overhead* akses memori dan menutup perbedaan kecepatan dengan CNN. Ini meningkatkan efisiensi *runtime*, menghilangkan pengkodean posisi untuk model yang lebih bersih dan cepat tanpa kehilangan akurasi deteksi. Selain itu, YOLOv12 mengurangi kompleksitas dengan menggunakan satu blok R-ELAN pada tahap terakhir *backbone* daripada menumpuk tiga blok perhatian/CNN. Ini memanfaatkan lapisan konvolusional dan *batch normalization* untuk menggantikan lapisan *linier*, yang meningkatkan efisiensi komputasi dan mencapai pertukaran akurasi latensi yang canggih (Tian, Ye, & Doermann, 2025).

2.4.6 Perbandingan YOLOv8 sampai YOLOv12

Tabel 2.1 Novelty Setiap Model YOLO

Model	YOLOv8	YOLOv9	YOLOv10	YOLOv11	YOLOv12
novelty	Anchor-Free C2f	PGI GELAN	NMS-Free SCDown PSA *Large-kernel convolution*	C3k2 C2PSA DWConv	*Area Attention* R-ELAN

Berdasarkan Tabel 2.1 setiap versi YOLO memperkenalkan inovasi arsitektur yang berperan penting dalam meningkatkan kinerja deteksi objek. YOLOv8 mengadopsi pendekatan *anchor-free* yang memisahkan proses deteksi, klasifikasi, dan regresi, memungkinkan masing-masing dioptimalkan secara independen. Selain itu, modul C2f (*Cross-stage partial bottleneck*

with two convolutions) meningkatkan efisiensi ekstraksi fitur dengan mempertahankan informasi kontekstual di lapisan jaringan yang lebih dalam, yang berdampak positif pada akurasi deteksi, terutama untuk objek kecil (Jocher, Chaurasia, & Qiu, Ultralytics YOLOv8, 2023).

YOLOv9 memperkenalkan PGI (*Programmable Gradient Information*), yang memungkinkan kontrol dinamis aliran gradien untuk memprioritaskan informasi penting selama pelatihan, serta GELAN (*Gradient Enhanced Lightweight Architecture Network*), yang mengoptimalkan jalur komputasi untuk model yang ringan namun akurat, sehingga meningkatkan konvergensi dan kinerja pada berbagai skala objek (Wang, Yeh, & Liao, 2024).

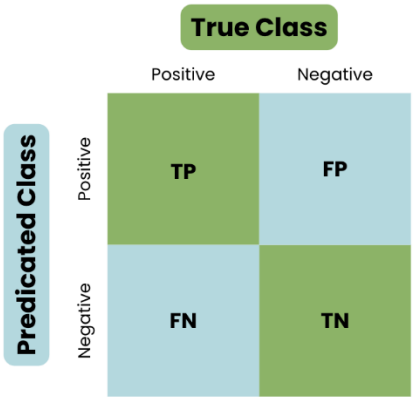
YOLOv10 menerapkan pendekatan NMS-Free yang menggantikan NMS dengan mekanisme *one-to-one assignment* saat inferensi, sehingga model langsung memilih satu prediksi terbaik untuk setiap objek tanpa perlu tahap pasca-pemrosesan yang memakan waktu. Sementara itu, pada tahap pelatihan menggunakan *one-to-many assignment*, di mana satu objek dapat memiliki beberapa prediksi positif. Cara ini membantu model memperoleh lebih banyak variasi contoh selama belajar, sehingga kemampuan deteksinya menjadi lebih akurat. Peningkatan tambahan meliputi SCDown (*Spatial-Channel Downsampling*) untuk efisiensi komputasi, PSA (*Partial Self-Attention*) untuk meningkatkan fokus model pada area penting, dan *large kernel convolution* untuk menangkap informasi dan konteks fitur pada area yang lebih luas, sehingga meningkatkan kemampuan model dalam mengenali pola global pada gambar (Wang, et al., 2024).

YOLOv11 menampilkan C3k2 (*Cross Stage Partial dengan ukuran kernel 2*), pengganti yang lebih efisien untuk C2f, C2PSA (Blok *Konvolusi dengan Perhatian Spasial Paralel*) untuk meningkatkan *attention mechanism*, dan DWConv (Konvolusi Depthwise) untuk mengurangi kompleksitas komputasi tanpa mengorbankan kinerja (Jocher & Qiu, Ultralytics YOLO11, 2024).

Akhirnya, YOLOv12 memperkenalkan *area attention*, yang mempertahankan bidang penerimaan yang luas sambil meminimalkan beban komputasi, dan R-ELAN (*Residual Efficient Layer Aggregation Networks*), yang mengoptimalkan agregasi fitur dan stabilitas pelatihan (Tian, Ye, & Doermann, 2025). Secara keseluruhan, perkembangan dari YOLOv8 hingga YOLOv12 menunjukkan tren konsisten menuju peningkatan efisiensi komputasi, optimasi arsitektur *backbone* dan *neck*, serta integrasi *attention mechanism* yang semakin canggih untuk mencapai keseimbangan antara kecepatan inferensi dan akurasi deteksi.

2.5 Confusion Matrix

Confusion matrix adalah alat untuk mengevaluasi akurasi berbagai model, termasuk model penilaian kredit dan pembelajaran mesin. Alat ini membandingkan label kelas yang diprediksi dengan label yang sebenarnya, membantu dalam memahami kinerja model dan mengidentifikasi kebingungan yang tersembunyi (Görtler, et al., 2022).



Gambar 2.7 Struktur Dasar Confusion Matrix

Sumber: datacamp.com

Berdasarkan Gambar 2.7 diatas terdapat beberapa nilai pada metrik, yaitu: TP (*True Positives*), TN (*True negatives*), FP (*False positives*) dan FN (*False negatives*), nilai tersebut bisa digunakan untuk menghitung nilai Presisi dengan persamaan

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

Presisi digunakan untuk mengukur ketepatan yang dicapai dalam prediksi yang benar. Selain akurasi dan presisi, ada *recall* untuk mengukur kepekaan dengan persamaan.

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

Recall digunakan untuk pengamatan aktual yang diprediksi benar, yaitu berapa banyak observasi kelas positif yang benar-benar diprediksi positif. Hal ini juga dikenal sebagai

Kepekaan. Untuk menghitung ketidakseimbangan kelas dalam data bisa menggunakan persamaan.

2.6 MaP (*Mean Average Precision*)

MaP (*Mean Average Precision*) adalah metrik evaluasi yang banyak digunakan dalam tugas deteksi objek dan pengenalan wajah, memberikan pengukuran komprehensif atas kinerja model dalam mengidentifikasi objek atau entitas tertentu (Pratama & Anggraeny, 2023). Metrik ini menghitung presisi rata-rata untuk setiap kelas dalam dataset dan kemudian mengambil rata-rata di semua kelas, menawarkan penilaian keseluruhan dari kinerja deteksi sistem. Presisi rata-rata untuk satu kelas ditentukan dengan menghitung area di bawah kurva *precision-recall* (PR), yang memplot presisi terhadap *recall* pada berbagai ambang batas *confident score* . Salah satu nilai yang umum direferensikan adalah mAP50 yang mengindikasikan bahwa metrik dihitung menggunakan *Thresshold* sebesar 0,5. Dalam konteks ini, prediksi dianggap benar jika IoU(*Intersection over Union*) antara *bounding box* yang diprediksi dan kotak *ground truth* memenuhi atau melebihi 0,5. Nilai mAP yang lebih tinggi mencerminkan kemampuan model yang lebih kuat untuk mendeteksi objek secara akurat dan konsisten di berbagai kategori (Wang B. , 2022). Perhitungan mAP dapat ditulis seperti persamaan (2.3).

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^{N} APi \qquad (2.3)$$

2.7 Box Loss

Box loss merujuk pada ukuran seberapa baik kotak batas yang diprediksi oleh model sesuai dengan objek sebenarnya dalam gambar. Hal ini sangat penting untuk memastikan deteksi objek yang akurat dalam tugas-tugas seperti mobil otonom atau kamera keamanan. Nilai *box loss* yang lebih rendah menunjukkan bahwa prediksi model lebih akurat, karena mencerminkan ketepatan kotak batas relatif terhadap posisi objek sebenarnya. Metrik ini membantu model menyesuaikan prediksinya selama pelatihan, sehingga meningkatkan kinerjanya secara keseluruhan dalam tugas deteksi objek (Jocher, Chaurasia, & Qiu, Ultralytics YOLOv8, 2023) (yolov8.org, n.d.).

2.8 Classification Loss

Classification Loss berperan penting dalam memastikan setiap objek yang terdeteksi diberi label kelas yang benar. *Loss* ini bekerja dengan cara membandingkan label prediksi dengan label sebenarnya, lalu menghitung perbedaan di antara keduanya. Semakin kecil nilai *classification loss*, semakin akurat model dalam mengklasifikasikan objek karena kesalahan prediksi yang terjadi digunakan untuk menyesuaikan bobot model agar lebih baik pada iterasi berikutnya. Fungsi ini sangat krusial dalam aplikasi nyata, seperti kendaraan otonom atau sistem keamanan berbasis kamera, di mana kesalahan klasifikasi dapat menimbulkan konsekuensi serius. Dengan demikian, *classification loss* tidak hanya membantu model belajar dari kesalahan, tetapi juga meningkatkan keandalannya dalam mengenali dan membedakan berbagai kelas objek secara konsisten (Jocher, Chaurasia, & Qiu, Ultralytics YOLOv8, 2023) (Potts, n.d.).

2.9 Distribution Focal Loss

Distributed Focal Loss (DFL) adalah fungsi *loss* yang dirancang untuk meningkatkan akurasi deteksi objek. Berbeda dengan fungsi kerugian standar, DFL berfokus pada contoh-contoh yang sulit dideteksi. DFL membantu model dengan memberikan bobot lebih besar pada contoh-contoh yang menantang, sehingga memudahkan model untuk belajar darinya. Fokus pada kasus-kasus kompleks ini membantu model menjadi lebih akurat seiring waktu. Cara kerjanya adalah dengan merepresentasikan lokasi koordinat kotak pembatas sebagai sebuah distribusi probabilitas, bukan sebagai satu nilai tunggal. Pendekatan ini memungkinkan model untuk fokus pada nilai-nilai yang paling mungkin di sekitar label target, sehingga menghasilkan prediksi lokasi objek yang lebih akurat dan andal, terutama dalam kasus objek yang ambigu atau sulit dideteksi (Jocher, Chaurasia, & Qiu, Ultralytics YOLOv8, 2023) (yolov8.org, n.d.).

2.10 Penelitian Sebelumnya

CNN efektif dalam mengekstraksi fitur dari data visual, seperti gambar dan *video*. Misalnya, sebuah penelitian oleh Arnold Sachith A Hans dan Smitha Rao menggunakan arsitektur CNN-LSTM untuk mendeteksi emosi dari data *video*, mencapai akurasi 78,52% pada dataset CREMA-D dan 63,35% pada dataset RAVDESS, yang menunjukkan potensi model pembelajaran mendalam dalam menangkap fitur temporal dan spasial yang kompleks untuk deteksi emosi (Hans & Rao , 2021).

Penelitian Faza N Azizi, Arrie Kurniawardhani, dan Irving V Paputungan menguji tiga metode optimasi, yaitu ADAM, SGD, dan RMS menggunakan CNN, serta mengadakan eksperimen dengan variasi jumlah *epoch* untuk mendeteksi emosi berdasarkan ekspresi. Dataset yang digunakan berasal dari FER2013, KDEF, dan CK+. Hasil studi tersebut mengindikasikan bahwa optimasi dengan ADAM memberikan tingkat akurasi paling tinggi dibandingkan dengan metode optimasi lainnya, mencapai akurasi tertinggi sebesar 82% pada dataset KDEF (Azizi, Kurniawardhani, & Paputungan, 2022).

Dalam penelitiannya, Kartika Salma dan Syarif Hidayat menggunakan algoritma YOLOv8 untuk FER (*Facial Expression Recognition*). Mereka secara khusus membuat dataset pelatihan dengan dua kelas, 'antusias' dan 'tidak antusias', yang berasal dari dataset FER2013. Penelitian ini menggunakan empat model YOLOv8 yaitu YOLOv8n, YOLOv8s, YOLOv8m, dan YOLOv8l. Hasil pengujian menunjukkan bahwa YOLOv8n mencapai akurasi tertinggi sebesar 95,3% dalam mendeteksi ekspresi antusias, sedangkan YOLOv8m memiliki performa terbaik dalam mendeteksi ekspresi non-antusias dengan akurasi sebesar 94,1%. (Salma & Hidayat, 2024).

Penelitian Jeri dan Zaid Syarif Hidayat menyelidiki ukuran *batch* yang optimal untuk melatih model YOLOv8 dalam pendeteksian objek untuk kendaraan otonom. Dengan menggunakan dataset simulasi lalu lintas CARLA dari Roboflow, yang terdiri dari 1.719 gambar, mereka bereksperimen dengan berbagai ukuran *batch* (4, 8, 16, 32, dan 64) dan pengaturan *hyperparameter*. Hasilnya menunjukkan bahwa peningkatan jumlah *epoch* dan ukuran *batch* meningkatkan mAP (*Mean Precision Average*) model. Konfigurasi terbaik mencapai mAP 98,2% dengan menggunakan 100 *epoch*, ukuran *batch* 32, dan resolusi gambar 640x640. Temuan ini memberikan wawasan yang berharga untuk mengoptimalkan model deteksi objek dan berfungsi sebagai referensi untuk studi masa depan dalam teknologi kendaraan otonom (Jeri & Hidayat, 2024).

Dalam penelitian Akhilesh Sharma, Vipan Kumar, dan Louis Longchamps mereka mengevaluasi kinerja YOLOv8, YOLOv9, YOLOv10, YOLOv11, dan Faster R-CNN untuk mendeteksi beberapa spesies *weed*. Mereka mengembangkan dataset gambar *weed* beranotasi yang terdiri dari 2.348 gambar, termasuk spesies seperti *cocklebur* (*Xanthium strumarium L.), dandelion* (*Taraxacum officinale*), *common waterhemp* (*Amaranthus tuberculatus*), *Palmer amaranth* (*Amaranthus palmeri*) dan *common lambsquarters* (*Chenopodium album L.)*. Hasil pengujian menunjukkan bahwa YOLOv9 mencapai akurasi tertinggi dengan mAP (*Mean Precision Average*) sebesar 0,935, sedangkan YOLOv11 adalah model tercepat dengan waktu

inferensi 13,5 ms. Secara keseluruhan, model YOLO menunjukkan akurasi dan kecepatan yang lebih unggul dibandingkan dengan Faster R-CNN, sehingga lebih cocok untuk aplikasi deteksi *weed* secara *real-time* (Sharma , Kumar , & Longchamps, 2024).

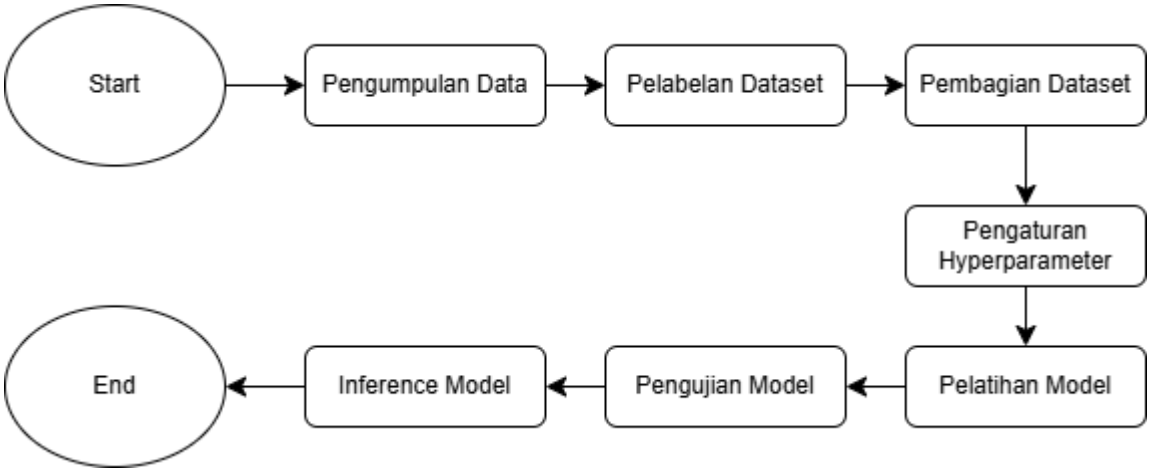
Penelitian terdahulu yang merupakan Sebuah studi validasi menilai efektivitas KDEF dan menemukan bahwa tingkat keberhasilan rata-rata dalam mengidentifikasi emosi target mencapai 71,87%. Basis data KDEF memudahkan peneliti memilih rangsangan berdasarkan kriteria tertentu, seperti jenis kelamin atau tingkat intensitas emosi, sehingga dapat digunakan untuk berbagai desain penelitian. Meski memiliki kelebihan, KDEF juga memiliki keterbatasan, terutama terkait perbedaan budaya dan faktor gender. Namun, KDEF tetap menjadi kumpulan gambar ekspresi wajah emosional yang valid dan banyak digunakan. Struktur basis data ini juga mencakup skor *hit rate* yang tidak bias, sehingga peneliti dapat memilih rangsangan yang seimbang untuk setiap emosi berdasarkan ekspresi terbaik, yang pada akhirnya meningkatkan keandalan penilaian emosi. KDEF telah banyak dimanfaatkan dalam berbagai penelitian dan berkontribusi besar dalam memahami proses pengenalan dan pemrosesan emosi dalam bidang psikologi (Goeleven, Raedt, Leyman, & Verschuere, 2008).. (Goeleven, Raedt, Leyman, & Verschuere, 2008)

Tabel 2.2 Kajian Literatur

No	Judul	Dataset	Metode	Hasil
1	A CNN-LSTM Based Deep Neutral Networks for Facial Emotion Detection in Videos	Ravdess dan CREMA-D	CNN-lSTM	Hasil pengujian mencapai akurasi 78,52% pada dataset CREMA-D dan 63,35% pada dataset RAVDESS
2	Deteksi Antusiasme Siswa dengan Algoritma YOLOv8 pada Proses Pembelajaran Daring	FER2013	YOLOv8	Hasil pengujian YOLOv8n memberikan hasil terbaik dengan mAP 95,3% dan *inference time* 69,2ms
3	Facial Expression Image based Emotion Detection using Convolutional Neural Network	FER2013, KDEF dan CK+	CNN	Hasil pengujian Akurasi tertinggi 82% pada dataset KDEF dengan optimasi adam
4	Penentuan Ukuran Batch Optimal Untuk Pelatihan YOLOv8 Dalam Pendeteksian Objek Pada Kendaraan Otonom	skema lalu lintas CARLA(Car Learning to Act) dari Roboflow universe	YOLOv8	Skema terbaik dengan nilai mAP sebesar 98,2% adalah 100 *epoch*, *batch size* 32, dan ukuran gambar 640x640
5	Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species	Weeds Dataset dari Roboflow universe	YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN	Hasil pengujian YOLOv9 mencapai akurasi tertinggi 93,5%, sedangkan YOLOv11 memiliki waktu inferensi terendah (13,5ms). Hasil keseluruhan model YOLO menunjukkan akurasi dan kecepatan yang lebih unggul dibandingkan dengan Faster R-CNN.
6	Facial expressions of emotion (KDEF): Identification under different display-duration conditions	KDEF	-	Hasil menunjukkan Dataset KDEF mencapai indeks keberhasilan rata-rata 71,87% untuk mengidentifikasi emosi target dengan benar di antara para partisipan

BAB III
METODOLOGI PENELITIAN

Tahapan atau langkah-langkah dalam penelitian ini akan digambarkan melalui *flowchart* atau diagram alir pada Gambar 3.1



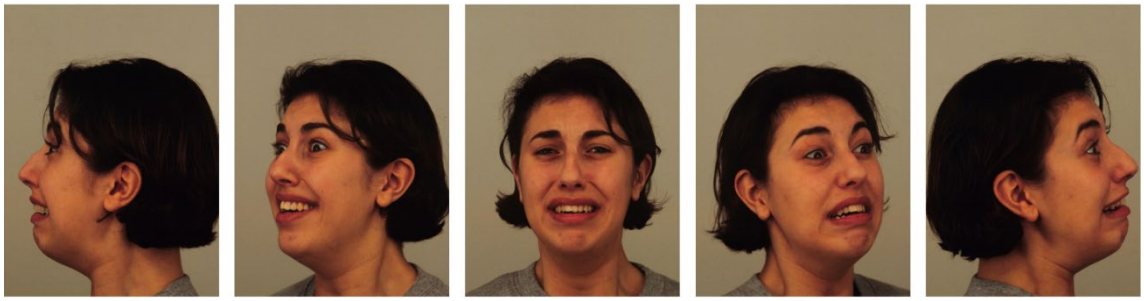
Gambar 3.1 Flowchart

3.1 Tahap Pengumpulan Data

Pada penelitian ini, dataset pelatihan yang digunakan adalah KDEF (*Karolinska Directed Emotional Faces*) yang memiliki format warna RGB dengan resolusi gambar 562×762 piksel dan dapat diunduh melalui tautan resmi di situs www.kdef.se. Dataset memiliki 7 kelas emosi, 5 *angle*, dan 2 sesi pemotretan dengan latar belakang berwarna cokelat muda yang berisi 4.900 gambar yang bisa digunakan untuk pelatihan model. Kelas emosi terserbut ialah marah, jijik, takut, bahagia, netral, sedih, dan terkejut seperti pada Gambar 3.2.

Gambar 3.2 Tujuh Ekspresi pada KDEF

Angle dibagi menjadi profil kiri penuh, profil setengah kiri, lurus, profil setengah kanan, dan profil kanan penuh. Kumpulan gambar ini terdiri dari 70 aktor amatir, 35 wanita dan 35 pria berusia antara 20 dan 30 tahun. Mereka tidak berjenggot, berkumis, memakai anting-anting atau kacamata, dan sebaiknya tidak memakai riasan wajah yang terlihat selama 2 sesi pemotretan (Lundqvist, Flykt, & Öhman, 1998).



Gambar 3.3 Tujuh Ekspresi pada KDEF

Dari 4.900 gambar yang tersedia penulis hanya mengambil 1.470 data dengan 3 angle saja yaitu profil setengah kiri, profil setengah kanan, dan lurus. Berdasarkan pertimbangan ini,

serta untuk mengoptimalkan efisiensi komputasi dan durasi pelatihan model maka hanya gambar dari tiga sudut pandang saja yang diambil. Namun, jumlah gambar yang digunakan dalam pelatihan menjadi 1.468 karena dua gambar rusak dan hanya menampilkan layar hitam. Dataset pengujian untuk *inference* menggunakan 7 emosi berbeda dari *video* Ravdess (Livingstone & Russo) dan gambar buatan sendiri berisi 4 orang yang diambil menggunakan *handphone* untuk *inference* model yang sudah dilatih.

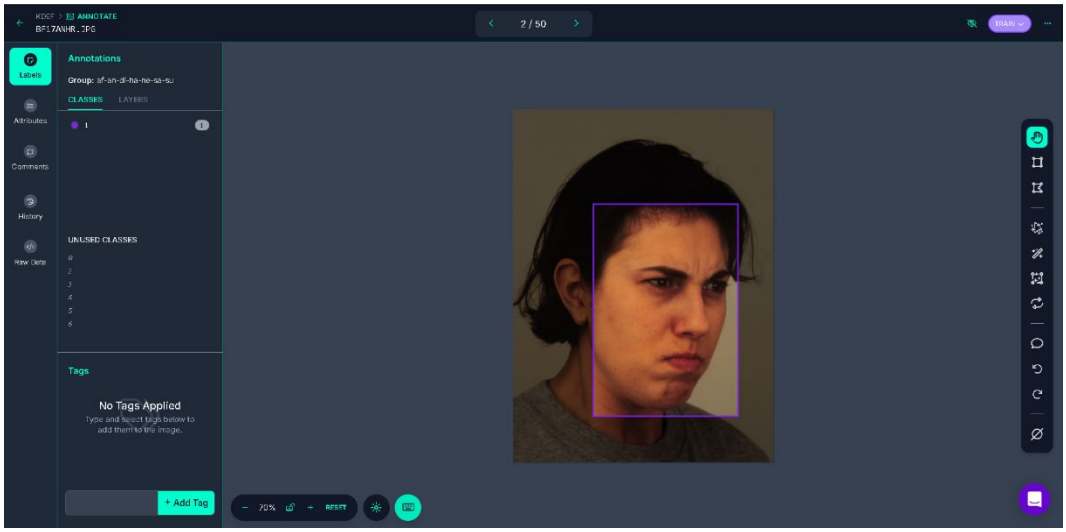
3.2 Tahap Pelabelan Dataset

Pelabelan memainkan peran penting dalam mengembangkan model pendeteksian objek. Tujuan utamanya adalah untuk menetapkan informasi tentang nama kelas objek yang akan diidentifikasi dan menentukan posisi kotak pembatas yang mengelilingi objek. Hal ini memastikan bahwa ketika aplikasi berjalan, sistem pendeteksian memprioritaskan untuk mengidentifikasi wajah tanpa terganggu oleh objek atau area yang tidak relevan dalam gambar. Peneliti melabeli gambar berdasarkan informasi dari dataset yang sudah di atur emosinya pada tiap gambar. Pada tahap awal pelabelan peneliti mengkategorikan 7 emosi tersebut ke dalam label berisi nilai numerik mulai dari 0 hingga 6, seperti yang ditunjukkan pada Tabel 3.1.

Tabel 3.1 Label Emosi

Emosi	takut	marah	jijik	bahagia	netral	sedih	kaget
Label	0	1	2	3	4	5	6

Kategori emosi yang sudah dibuat digunakan untuk melakukan tahap pelabelan gambar. Tahap ini dilakukan satu per satu dengan bantuan *tool* Roboflow (Roboflow Inc., n.d.) seperti pada Gambar 3.3. Tahap pelabelan dilakukan dengan cara memberi *bounding box* sebagai area pembatas setiap objek dan pemberian label emosi sesuai informasi pada dataset KDEF. Hasilnya disimpan dalam *file* .txt dengan *format* data YOLO.



Gambar 3.4 Proses Pelabelan dengan Roboflow

3.3 Tahap Pembagian Dataset

Tahap Pembagian dataset dilakukan dengan menggunakan *tool* Roboflow (Roboflow Inc., n.d.). dataset asli dibagi menjadi dua bagian utama: data pelatihan dan data validasi, dengan menggunakan rasio pembagian 80:20 (Adinata, Rohana, Baihaqi, & Faisal, 2024). Pembagian ini diterapkan secara proporsional di setiap label untuk memastikan distribusi yang seimbang. Dari keseluruhan data, 80% dialokasikan sebagai data pelatihan, yang setara dengan total 1.175 gambar. Porsi ini digunakan untuk melatih model, memungkinkannya untuk belajar dan mengenali pola dalam gambar.

Sementara itu, 20% data lainnya yang terdiri dari 293 gambar diperuntukkan sebagai data validasi, dengan pembagian yang juga dipertahankan secara proporsional. Data validasi memiliki peran penting untuk menguji kinerja model yang telah dilatih, terutama dalam menguji kemampuannya untuk mengidentifikasi ekspresi wajah secara akurat dan melakukan generalisasi dengan baik. Lalu, untuk *inference* menggunakan data di luar dari dataset yaitu dataset Ravdess dan foto dari *handphone*.

3.4 Tahap Mengatur *Hyperparameter*

Memilih *hyperparameter* yang sesuai adalah langkah penting dalam meningkatkan kinerja model selama pelatihan. Penelitian ini menekankan tiga *hyperparameter* utama: ukuran *batch*, jumlah *epoch*, dan resolusi gambar. Berdasarkan temuan dari studi sebelumnya, penelitian ini menetapkan pemilihan untuk tiga *hyperparameter*. Ukuran *batch* sebesar 32

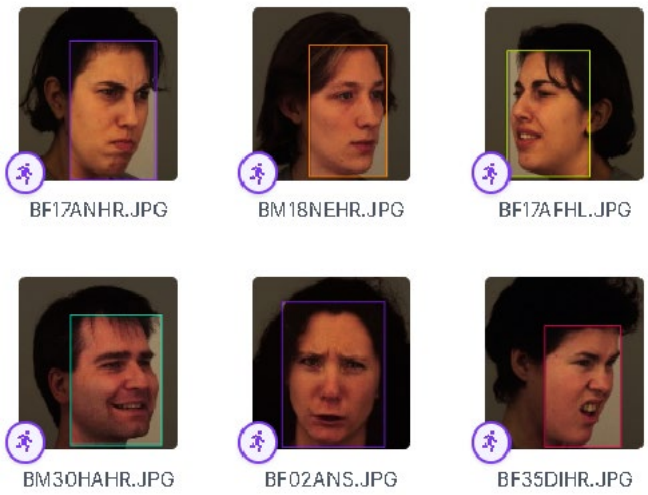
dipilih untuk menyeimbangkan efisiensi komputasi dengan stabilitas pelatihan. Selanjutnya, jumlah *epoch* ditetapkan pada 100 untuk memberikan waktu yang cukup bagi model dalam mencapai konvergensi yang matang dan memaksimalkan akurasi.

Terakhir, model-model yang diuji dalam penelitian ini meliputi YOLOv8, YOLOv9, YOLOv10, YOLOv11, dan YOLOv12 yang dilatih dengan resolusi gambar diatur pada 640x640 piksel, karena terbukti bahwa gambar beresolusi lebih tinggi memungkinkan ekstraksi fitur yang lebih kaya, yang secara langsung berkontribusi pada peningkatan nilai mAP (Jeri & Hidayat, 2024). Parameters lainnya seperti *weight decay, momentum, learning rate, device, dropout, workers* dan *validation* diatur sebagai default.

3.5 Tahap Melatih Model

Setelah pelabelan kelas objek disiapkan dan data dibagi, langkah selanjutnya adalah memulai proses pelatihan. Tujuan pelatihan ini adalah untuk memungkinkan model memproses gambar dan pelabelan sehingga setiap kelas dapat mengembangkan pola atau karakteristik yang berbeda yang dapat digunakan komputer sebagai acuan untuk mengambil keputusan atau membuat prediksi. Setelah proses pelatihan selesai, model yang digunakan dalam penelitian ini dipilih berdasarkan *best* model, bukan model terakhir dari seluruh *epoch*. Pemilihan *best* model ini dilakukan dengan mengacu pada nilai akurasi validasi, karena akurasi pada data validasi lebih merepresentasikan kemampuan generalisasi model dibandingkan hanya menggunakan akurasi *training*. Dengan demikian, model yang dipilih merupakan model yang memiliki performa terbaik pada data validasi. Sebagai acuan evaluasi, penelitian ini menggunakan akurasi sebagai patokan utama, bukan nilai loss. Hal ini dikarenakan akurasi lebih langsung menggambarkan kemampuan model dalam mengklasifikasikan objek ke dalam kelas yang benar, sedangkan nilai *loss* hanya merepresentasikan ukuran kesalahan prediksi.

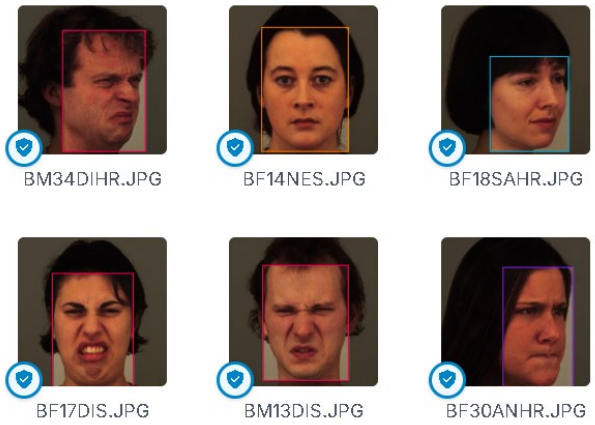
Dengan demikian, akurasi menjadi indikator yang lebih relevan untuk menilai keandalan model dalam konteks implementasi nyata, khususnya pada sistem deteksi emosi berbasis wajah. Pelatihan ini menggunakan 1.468 gambar yang diperoleh dari pembagian dataset. Model yang dilatih dalam studi ini meliputi YOLOv8n, YOLOv9n, YOLOv10n, YOLOv11n, dan YOLOv12n. Varian nano dari masing-masing model digunakan supaya dapat diimplementasikan pada perangkat dengan keterbatasan sumber daya, sehingga cocok untuk aplikasi *real-time* di lingkungan dengan kapasitas komputasi terbatas. Contoh gambar yang digunakan ditampilkan pada Gambar 3.4. Hasilnya, termasuk kinerja model dan durasi pelatihan, disimpan dalam output.



Gambar 3.5 Data Pelatihan

3.6 Tahap Pengujian Model

Pengujian model dilakukan dengan menggunakan 20% data validasi dari total dataset yang telah dipisahkan pada saat proses *data-splitting*, yaitu sebanyak 293 citra. Model-model ini dievaluasi dengan menggunakan 293 gambar yang diperoleh dari pembagian dataset. Contoh gambar data validasi yang digunakan dalam proses pengujian dapat dilihat pada Gambar 3.5.



Gambar 3.6 Data Validasi

3.7 Tahap Inference Model

Tahap inference model dilakukan pada gambar hasil foto menggunakan handphone serta pada tujuh video emosi berbeda dari dataset RAVDESS (Livingstone & Russo). Setiap video hanya menampilkan satu orang dengan satu jenis emosi pada durasi sekitar 3 detik dengan 30 *frame per second* dan resolusi 1280×720 piksel, dengan latar belakang yang relatif seragam tanpa adanya variasi signifikan. YOLOv12n diuji pada dua skenario berbeda. Pertama, pengujian dengan gambar dari video RAVDESS dilakukan untuk mengevaluasi stabilitas model. Kedua, pengujian dengan gambar yang diambil dari ponsel bertujuan untuk menilai kemampuan model dalam mendeteksi emosi pada banyak wajah sekaligus, sebagaimana kasus nyata ketika berhadapan dengan banyak konsumen dalam satu frame.

Hasil inference untuk video RAVDESS dengan satu wajah diukur berdasarkan emosi yang paling dominan muncul sepanjang durasi video. Hal ini dikarenakan prediksi emosi pada setiap frame dapat berfluktuasi, sehingga keberhasilan deteksi ditentukan oleh konsistensi model dalam menghasilkan label emosi yang sama secara mayoritas. Dengan demikian, suatu deteksi dianggap stabil apabila prediksi emosi yang benar mendominasi keseluruhan frame meskipun terdapat variasi kecil atau perubahan sesaat. Sementara itu, untuk kasus image multi-wajah, keberhasilan pengujian diukur melalui confidence score yang dihasilkan model serta kesesuaian klasifikasi pada setiap wajah terhadap label emosi yang sebenarnya.

BAB IV
PENELITIAN

4.1 Implementasi

Tahapan yang ada dalam diagram alur pada Gambar 3.1 diimplementasikan dalam program menggunakan Python versi 3.11.10, TensorFlow versi 2.18.0, dan OpenCV versi 4.10.0. Implementasi ini dilakukan secara sistematis, mengikuti urutan langkah demi langkah yang digambarkan dalam diagram alir. Setelah kode ditulis, kode tersebut dieksekusi di Google Colab menggunakan GPU T4, yang menyediakan 15 GB RAM GPU, 12,7 GB RAM sistem, dan 112,6 GB ruang disk. Google Colab dipilih karena kemampuannya mendukung pemrograman dan eksekusi Python secara interaktif sambil menawarkan akses gratis ke sumber daya GPU berkemampuan tinggi, menjadikannya sangat cocok untuk pelatihan dan eksperimen model deep learning. Beberapa perpustakaan Python digunakan untuk memudahkan proses pengembangan, termasuk os versi 3.11.10 untuk berinteraksi dengan sistem operasi (seperti pengelolaan direktori dan file), glob versi 0.7 untuk mengambil jalur file yang sesuai dengan pola tertentu, dan shutil versi 3.5.2 untuk operasi file tingkat tinggi seperti menyalin, memindahkan, dan menghapus file atau direktori. Bagian Hasil dan Pembahasan memberikan penjelasan rinci tentang langkah-langkah dan kode program yang sesuai untuk setiap proses dalam sistem.

4.2 Tahap Mengumpulkan data

Setelah dataset diunduh dari situs resmi, proses pengumpulan data yang diperlukan dilakukan untuk mengambil dataset yang dibutuhkan saja dengan menggunakan kode pada Gambar 4.1.

```
import os
import glob
import shutil


def cari_gambar(base_folder):
    # Buat list prefiks folder
    prefixes = ['BF', 'BM']
```

```
    # List untuk menyimpan semua target folder
    target_folders = []

    # Generate semua kombinasi folder
    for prefix in prefixes:
        for i in range(1, 36):
            target_folders.append(f'{prefix}{str(i).zfill(2)}')
    # List untuk menyimpan path file yang sesuai
    selected_images = []

    # Iterasi setiap folder target
    for folder in target_folders:
        folder_path = os.path.join(base_folder, folder)

        # Pastikan folder ada
        if os.path.exists(folder_path):
            # Cari file yang berakhiran HR, HL dan S
            sudut_hr = os.path.join(folder_path, '*HR.JPG')
            sudut_hl = os.path.join(folder_path, '*HL.JPG')
            sudut_s = os.path.join(folder_path, '*S.JPG')

            # Tambahkan semua file yang sesuai ke dalam list
            selected_images.extend(glob.glob(sudut_hr))
            selected_images.extend(glob.glob(sudut_hl))
            selected_images.extend(glob.glob(sudut_s))

    return selected_images

# Gunakan fungsi
base_folder = 'E:/KDEF_and_AKDEF/KDEF'
image_paths = cari_gambar(base_folder)

# Buat folder utama fixdataset
```

```
main_output_folder = 'fixdataset'
if not os.path.exists(main_output_folder):
    os.makedirs(main_output_folder)

# Buat subfolder untuk setiap jenis gambar
subfolder_mapping = {
    'HR': os.path.join(main_output_folder, 'half_right'),
    'HL': os.path.join(main_output_folder, 'half_left'),
    'S': os.path.join(main_output_folder, 'straight')
        }

        # Buat subfolder jika belum ada
        for folder in subfolder_mapping.values():
            if not os.path.exists(folder):
                os.makedirs(folder)

        # Salin file ke folder yang sesuai
        for path in image_paths:
            filename = os.path.basename(path)

            # Tentukan subfolder berdasarkan akhiran file
            if filename.endswith('HR.JPG'):
                destination_folder = subfolder_mapping['HR']
            elif filename.endswith('HL.JPG'):
                destination_folder = subfolder_mapping['HL']
            elif filename.endswith('S.JPG'):
                destination_folder = subfolder_mapping['S']

            # Tentukan path tujuan lengkap
            destination = os.path.join(destination_folder, filename)

            # Salin file
            shutil.copy2(path, destination)
```

```
        print(f"Menyalin {filename} ke folder {os.path.basename(destination_folder)}")

    print(f"\nTotal gambar yang disalin: {len(image_paths)}")
```

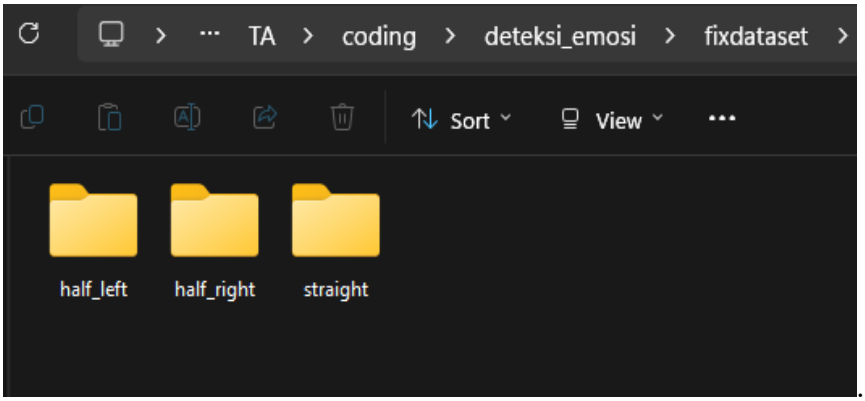
Gambar 4.1 Kode Pengambilan Data

Kode ini memanfaatkan pustaka ‘os’, ‘glob’, dan ‘shutil’ dari Python untuk melakukan tugas-tugas ini secara efisien. Awalnya, fungsi ‘cari_gambar(base_folder)’ mendefinisikan cakupan pencarian dengan membuat daftar folder target berdasarkan dua awalan yang telah ditentukan, “BF” dan “BM,” yang kemudian digabungkan dengan nilai numerik mulai dari 01 hingga 35. Hal ini memastikan bahwa *script* secara sistematis mencari file yang relevan di semua direktori potensial dalam kumpulan data. Fungsi ini kemudian mengulang melalui folder target ini, memverifikasi keberadaannya sebelum menerapkan kriteria pencocokan nama file tertentu menggunakan modul `glob`. *script* diprogram untuk mengidentifikasi file gambar yang diakhiri dengan akhiran “HR.JPG” (gambar setengah kanan), “HL.JPG (gambar setengah kiri), dan ‘S.JPG’ (gambar lurus). Jalur file yang teridentifikasi dikumpulkan ke dalam daftar untuk diproses lebih lanjut.

Setelah proses pemilihan gambar, *script* membuat hierarki direktori terstruktur untuk mengatur gambar yang diekstrak. Direktori keluaran utama, yang ditetapkan sebagai “fixdataset,” berfungsi sebagai tempat penyimpanan pusat untuk menyimpan gambar yang telah dikategorikan. Di dalam direktori ini, tiga subfolder “half_right,” “half_left,” dan “straight” dibuat secara dinamis untuk menyimpan gambar berdasarkan akhiran nama file yang sesuai. Struktur hirarkis ini meningkatkan pengorganisasian kumpulan data, memastikan pengambilan dan pemanfaatan yang efisien dalam pemrosesan data berikutnya atau aplikasi pembelajaran mendalam.

Setelah struktur folder dibuat, *script* melakukan iterasi melalui daftar gambar yang dipilih, menentukan folder tujuan yang sesuai berdasarkan akhiran nama file, dan menyalin gambar ke direktori yang sesuai menggunakan fungsi ‘shutil.copy2()’. Operasi ini memastikan bahwa setiap gambar ditempatkan secara sistematis dalam kategori yang ditentukan tanpa mengubah struktur kumpulan data asli. Selain itu, *script* ini memberikan umpan balik *real-time* dengan mencetak pesan konfirmasi, yang mengindikasikan transfer file yang berhasil untuk hasil dari

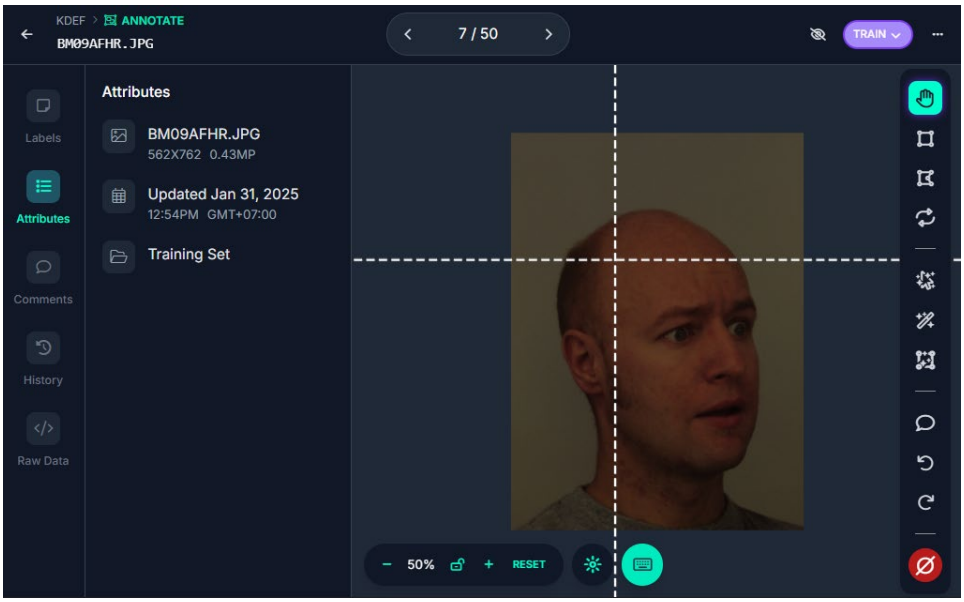
script dapat dilihat pada gambar 4.2. Setelah itu data di upload ke platform Roboflow agar bisa di labeli.



Gambar 4.2 Hasil dari Proses Pengumpulan data

4.3 Tahap Pelabelan Data

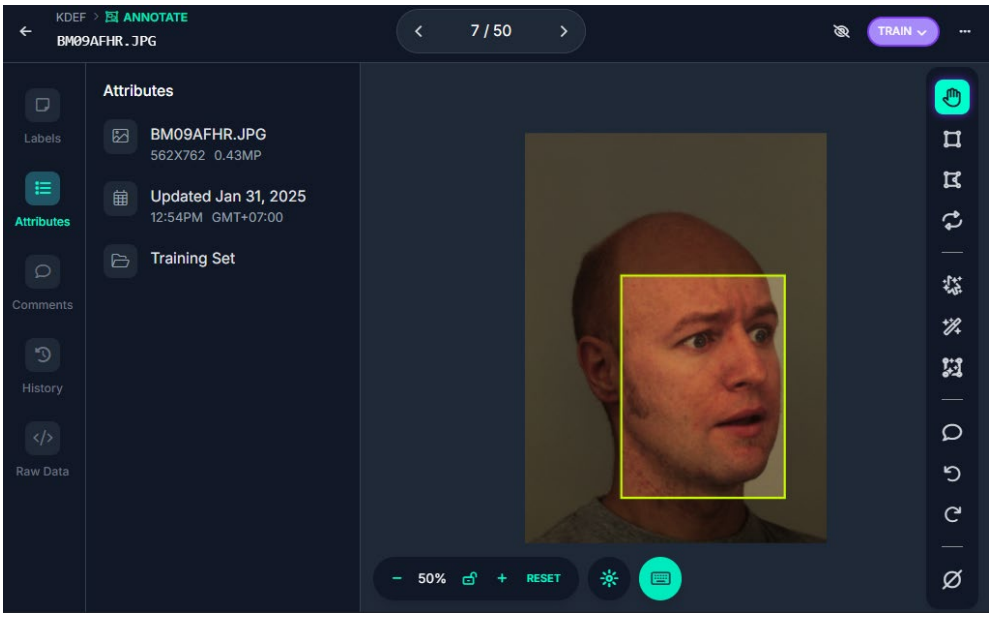
Dataset yang telah diunggah ke Roboflow hanya berupa kumpulan gambar tanpa informasi tambahan mengenai posisi dan jenis objek yang terkandung di dalamnya seperti pada gambar 4.3.



Gambar 4.3 Data Sebelum Dilabeli

Proses pelabelan kemudian dilakukan menggunakan tool Roboflow dengan membuat kotak pembatas (*bounding box*) pada setiap objek yang menjadi target deteksi, disertai

pemberian label sesuai nama kelas masing-masing objek. Hasilnya, setiap gambar kini memiliki label yang terdiri dari koordinat bounding box dan label kelas, seperti yang ditunjukkan pada Gambar 4.4.



Gambar 4.4 Data Setelah Dilabeli

Dataset yang telah dilabeli ini kemudian siap digunakan sebagai data pelatihan untuk model deteksi. *Bounding box* ini memberikan informasi penting tentang lokasi dan ukuran objek dalam gambar. Hasil dari proses pelabelan dapat dilihat pada Gambar 4.5.

```
0 0.42362989323843425 0.5661154855643045 0.541814946619217 0.5591863517060368

```

Gambar 4.5 Hasil dari Proses Pelabelan dengan Roboflow

Nilai pertama, yaitu “0”, mewakili label kelas takut yang menunjukkan bahwa objek dalam gambar termasuk ke dalam kategori emosi takut. Empat nilai desimal berikutnya merepresentasikan koordinat kotak pembatas yang telah dinormalisasi, yakni 0,423 untuk posisi tengah sumbu x, 0,566 untuk posisi tengah sumbu y, 0,541 untuk lebar, dan 0,599 untuk tinggi kotak pembatas. Semua nilai tersebut diskalakan dalam rentang 0 hingga 1 untuk menjaga konsistensi pada gambar dengan ukuran yang bervariasi. *Format* anotasi ini sangat

penting untuk melatih model *deep learning* dalam tugas pendeteksian objek, memungkinkan model untuk secara akurat melokalisasi dan mengklasifikasikan objek di dalam gambar.

4.4 Tahap Pembagian dataset

Tahap pembagian dataset berdasarkan pembagian dataset dengan 80% dialokasikan untuk pelatihan dan 20% untuk validasi, total 1.175 gambar ditetapkan untuk pelatihan sementara 293 gambar dicadangkan untuk validasi. Pembagian ini dilakukan secara proporsional untuk kategori emosi untuk memastikan distribusi yang seimbang sepeti pada Tabel 4.1.

Tabel 4.1 Pembagian Dataset

Kategori	**Emosi**							**Total**
	Affraid	***Angry***	***Disgust***	***Happy***	***Neutral***	***Sad***	***Surprise***	
Data latih	168	168	167	168	168	168	168	1175
Data validasi	42	42	42	42	41	42	42	293

4.5 Tahap Mengatur *Hyperparameter*

Mengatur *Hyperparameter* dapat dilakukan dengan mengatur *variable* ”epochs” untuk jumlah iterasi pelatihan, ” imgsz” untuk resolusi gambar dan ”batch” untuk jumlah *batch size* seperti pada Gambar 4.6. nilai default tiap parameters yaitu *weight decay* = 0.0005 , *momentum* = 0.937, *learning* rate = 0.01, *optimizer* = ‘auto’, *workers* = 8 dan validation = *True* (Ultralytics, n.d.) (Jocher & Qiu, Ultralytics YOLO11, 2024).

```
results = model.train(data="data.yaml", epochs=100, imgsz=640, batch=32)
```

Gambar 4.6 Kode Mengatur Hyperparameter

parameter data merujuk pada berkas konfigurasi yang berisi informasi tentang struktur dataset yang digunakan, termasuk lokasi folder data pelatihan dan validasi, daftar nama kelas objek, serta jumlah kelas (number of classes). Berkas data.yaml ini sangat penting karena digunakan model sebagai acuan untuk membaca data dan label selama proses pelatihan seperti pada Gambar 4.7.

```yaml
train: train/images
val: valid/images

nc: 7
names: ['0', '1', '2', '3', '4', '5','6']
```

Gambar 4.7 Kode File data.yaml

Parameter train dan val menunjukkan jalur direktori tempat penyimpanan gambar pelatihan dan validasi. *Parameter* nc, yang merupakan singkatan dari "*number of classes*", diatur menjadi 7, yang menjelaskan tujuh kategori emosional yaitu takut, marah, jijik, bahagia, netral, sedih dan kaget. Sementara itu, daftar nama secara eksplisit mendefinisikan label kelas yang sesuai dengan emosi ini. Konfigurasi ini biasanya digunakan dalam kerangka kerja *deep learning*, khususnya dalam deteksi objek dan tugas klasifikasi gambar, untuk memastikan model memetakan gambar dengan benar ke kategori masing-masing.

4.6 Tahap Pelatihan

Setelah data telah dilabeli Langkah selanjutnya yaitu masuk ke tahap pelatihan pada tahap pelatihan dilakukan dengan menggunakan metode *transfer learning* (Ultralytics, n.d.). Kode program untuk tahap ini ditulis di Google Colab, dengan kode pelatihan yang ditampilkan sspada Gambar 4.8.

```
from ultralytics import YOLO


#Load model
model=YOLO("YOLO11n.pt")
#Train model
results = model.train(data="data.yaml", epochs=100, imgsz=640, batch=32)
```

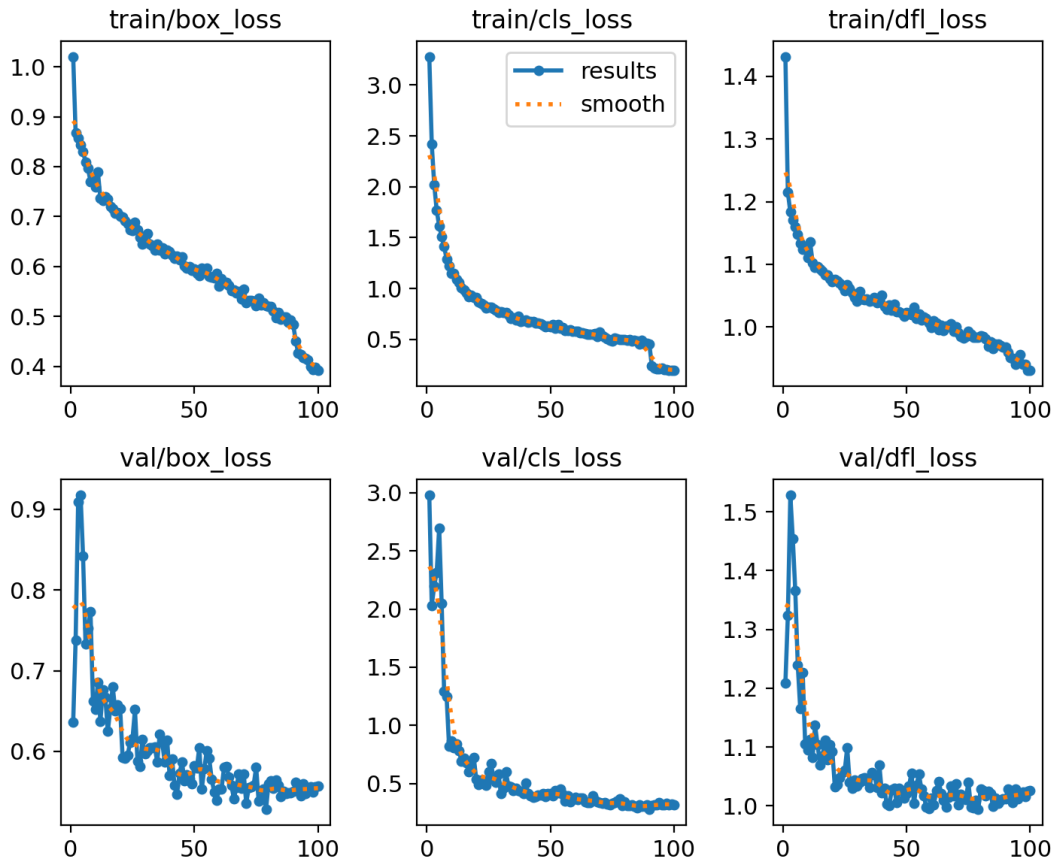
Gambar 4.8 Kode Pelatihan Model YOLOv11n

Tahapnya dimulai dengan mengimpor pustaka ultralytics, yang dirancang khusus untuk bekerja dengan model YOLO untuk tugas-tugas seperti deteksi objek. Pustaka ultralytics menawarkan berbagai fungsi untuk menyederhanakan tugas-tugas seperti memuat model YOLO, melatihnya, menguji kinerjanya, dan membuat prediksi.

Setelah pustaka Ultralytics diimpor, langkah berikutnya adalah memanggil kelas YOLO di dalam pustaka tersebut. Model YOLO yang telah dilatih sebelumnya kemudian dapat dimuat dari file, misalnya YOLOv11n.pt. *File* ini berisi bobot hasil pelatihan yang spesifik untuk varian model tertentu. Untuk melatih atau menguji varian model YOLO lainnya, pengguna hanya perlu mengganti nama *file*, misalnya dari YOLOv11n.pt ke YOLOv12n.pt sesuai dengan varian yang diinginkan. Dengan demikian, proses pergantian model menjadi sederhana tanpa perlu mengubah struktur kode secara keseluruhan.

4.6.1 Hasil Pelatihan

YOLOv12n

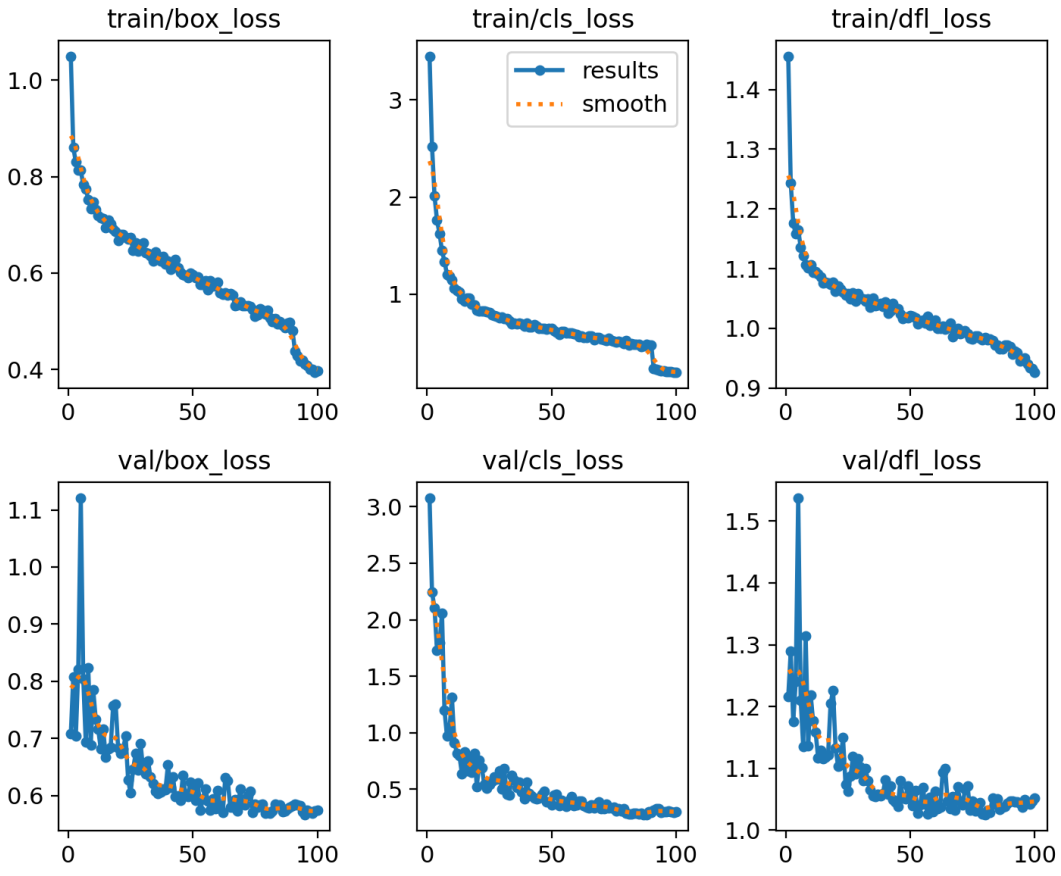


Gambar 4.9 Grafik Loss YOLOv12n

Pada proses pelatihan YOLOv12n memakan waktu 0.788 jam dan epoch dengan hasil terbaik adalah epoch ke-87. Grafik dari data pelatihan YOLOv12n pada Gambar 4.9

menunjukkan performa model selama 100 *epoch*. Dapat dilihat bahwa nilai *loss* menurun secara bertahap dan konsisten. Pada awal pelatihan, nilai untuk masing-masing adalah 1,055 *train/box_loss,* 3,314 *train/cls_loss*, dan 1,466 *train/dfl_loss*. Mengindikasikan bahwa model masih berada pada tahap awal pembelajaran untuk memprediksi kotak pembatas, mengklasifikasikan objek, dan mengestimasi distribusi. Namun, seiring berjalannya pelatihan, nilai-nilai ini menurun secara signifikan, mencapai sekitar 0,396 *train/box_loss*, 0,197 *train/cls_loss*, dan 0,921 *train/dfl_loss* pada *epoch* ke-100. Tren serupa terlihat pada data validasi, di mana *val/box_loss* dimulai dari 0,657 lalu terjadi lonjakan ke 0,9 setelah itu menurun menjadi sekitar 0,563, *val/cls_loss* dari 2,975 menjadi sekitar 0,261, dan *val/dfl_loss* dari 1,162 terjadi kenaikan sampai 1,5 hingga akhirnya turun sekitar 1,049. Penurunan ini mencerminkan bahwa model tidak hanya belajar dari data pelatihan tetapi juga mampu menggeneralisasi dengan baik untuk data validasi yang tidak terlihat.

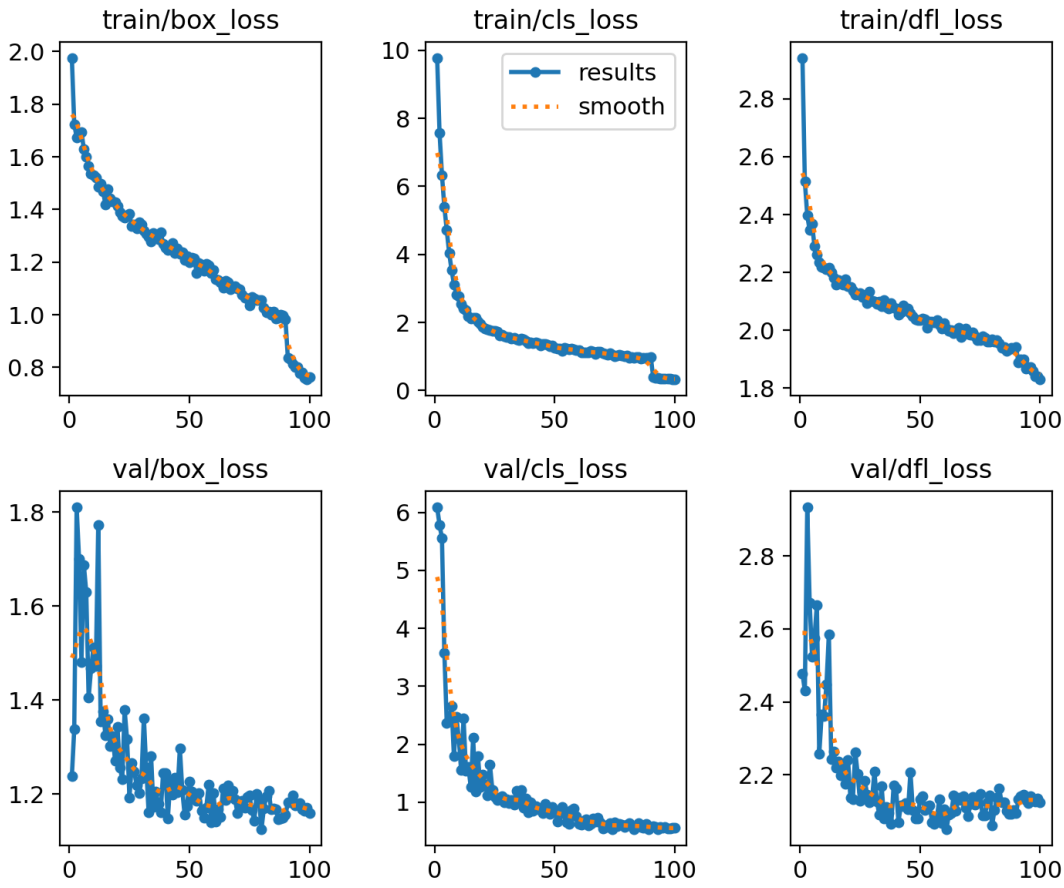
YOLOv11n



Gambar 4.10 Grafik Loss YOLOv11n

Pada proses pelatihan YOLOv12n memakan waktu 0,640 jam dan *epoch* dengan hasil terbaik adalah *epoch* ke-95. Visualisasi grafik *loss* pada Gambar 4.10 menunjukkan perkembangan kinerja model YOLOv11n selama 100 *epoch*. Sejak awal, model menunjukkan nilai *loss* yang tinggi-khususnya, *train/box_loss* sebesar 1,050, *train/cls_loss* sebesar 3,451, dan *train/dfl_loss* sebesar 1,455. Mengindikasikan tahap awal pembelajarannya untuk melokalisasi objek, mengklasifikasikan emosi, dan memprediksi distribusi jarak. Namun, seiring dengan berjalannya pelatihan, semua nilai *loss* terus menurun, mencapai sekitar 0,396 *train/box_loss*, 0,193 *train/cls_loss*, dan 0,925 *train/dfl_loss* pada *epoch* terakhir, yang mencerminkan peningkatan kemampuan model. Tren ini juga terlihat pada set validasi, dengan *val/box_loss* dari sekitar 0,707 dengan sedikit lonjakan hingga ke 1,1 lalu menurun dengan stabil ke 0,574, *val/cls_loss* menurun dari 3,076 menjadi 0,299, dan *val/dfl_loss* berawal dari 1,216 terjadi kenaikan sampai ke 1,5 hingga akhirnya menurun menjadi 1,051. Pengurangan yang konsisten tersebut menunjukkan bahwa model tidak hanya belajar secara efektif dari data pelatihan tetapi juga menggeneralisasi dengan baik untuk data yang tidak terlihat, tanpa ada tanda-tanda *overfitting* selama proses berlangsung.

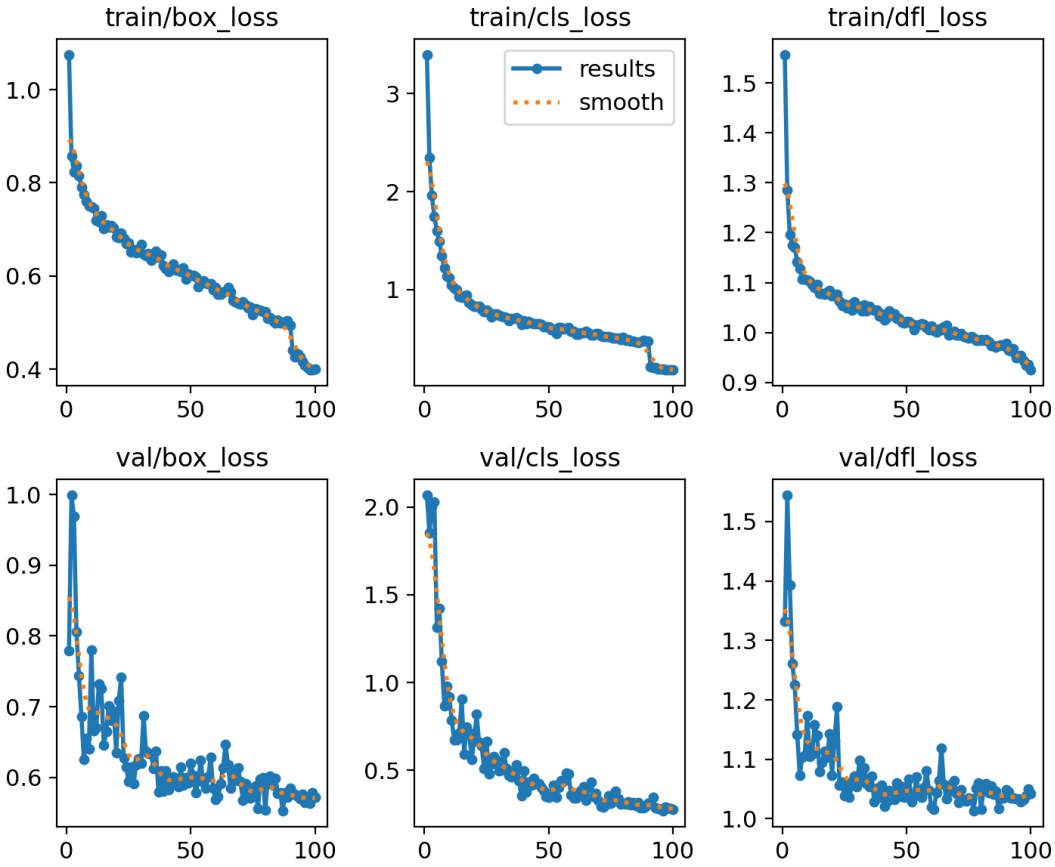
YOLO10n



Gambar 4.11 Grafik Loss YOLOv10n

Pada proses pelatihan YOLOv12n memakan waktu 0,761jam dan *epoch* dengan hasil terbaik adalah *epoch* ke-93. Grafik pada Gambar 4.11 menggambarkan kemajuan pembelajaran model YOLOv10n selama 100 *epoch*. Pada awalnya, model menunjukkan nilai *loss* yang relatif tinggi-yaitu, *train/box_loss* sebesar 1,976, *train/cls_loss* sebesar 9.781, dan *train/dfl_loss* sebesar 2,941. Menunjukkan tahap awal di mana model masih belajar untuk mendeteksi lokasi objek, mengklasifikasikan ekspresi emosional, dan memperkirakan distribusi jarak. Seiring berjalannya waktu, nilai *loss* ini secara konsisten menurun, masing-masing mencapai sekitar 0,753 *train/box_loss*, 0,322 *train/cls_loss*, dan 1,829 *train/dfl_loss*, pada *epoch* terakhir menunjukkan peningkatan yang jelas dalam kinerja model. Pola yang sama juga terlihat pada set validasi, di *mana val/box_loss* turun dari sekitar 1,237 menjadi 1,159, *val/cls_loss* turun dari 6,087 menjadi 0,556, dan *val/dfl_loss* turun dari 2,478 menjadi 2,124. Penurunan yang stabil di seluruh komponen kerugian ini menunjukkan bahwa model tidak hanya belajar secara efektif dari set pelatihan tetapi juga menggeneralisasi dengan baik untuk data yang tidak terlihat, tanpa ada tanda-tanda yang jelas dari *overfitting* selama proses pelatihan.

YOLOv9t

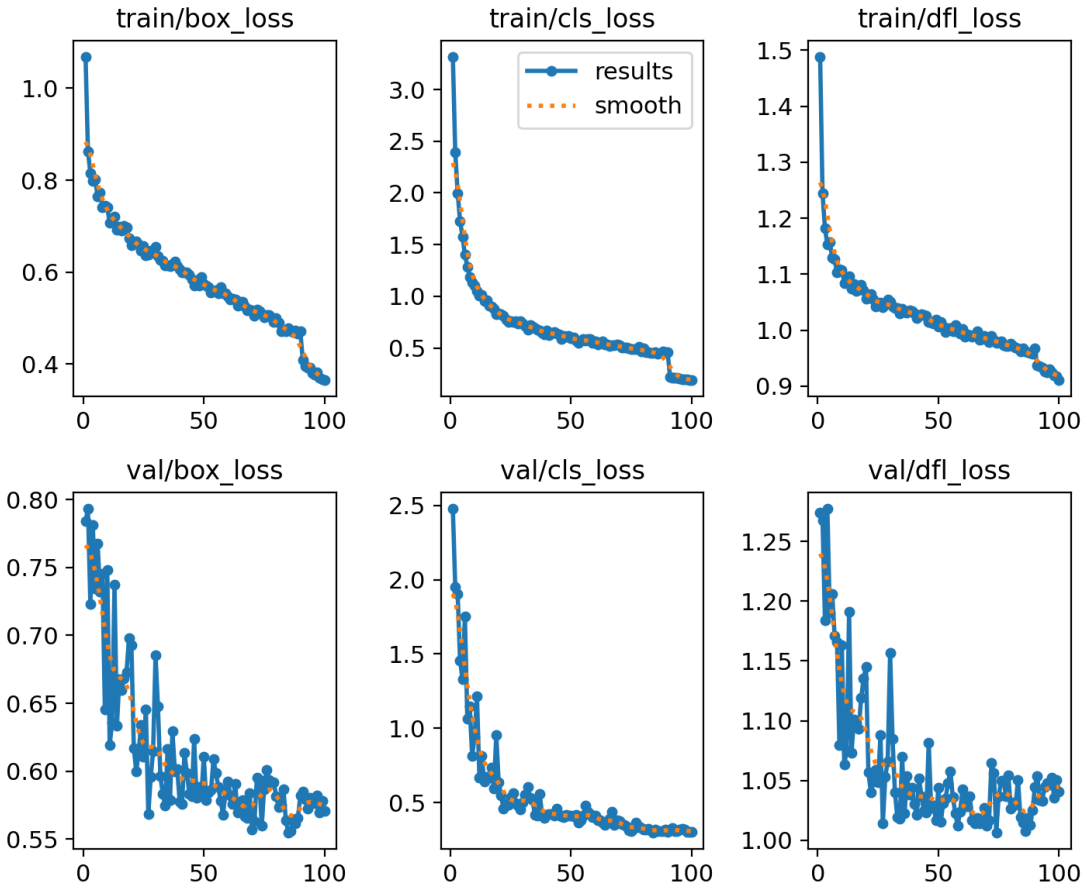


Gambar 4.12 Grafik Loss YOLOv9t

Pada proses pelatihan YOLOv12n memakan waktu 0,795 jam dan *epoch* dengan hasil terbaik adalah *epoch* ke-94. Perkembangan performa model YOLOv9t selama 100 epoch pelatihan tercermin dengan jelas pada Gambar 4.12. Pada awalnya, model mencatat nilai *loss* yang relatif tinggi-khususnya, *train/box_loss* sebesar 1,087, *train/cls_loss* sebesar 3,405, dan *train/dfl_loss* sebesar 1,565 menunjukkan fase pembelajaran awal, di mana model ini masih beradaptasi untuk melokalisasi objek, mengidentifikasi kelas emosi, dan memodelkan distribusi jarak. Seiring berjalannya pelatihan, kerugian ini terus menurun, dengan nilai akhir mendekati 0,403 *train/box_loss*, 0,177 *train/cls_loss*, dan 0,934 *train/dfl_loss*, yang menandakan peningkatan yang signifikan dalam kemampuan pembelajaran model.

Tren penurunan yang konsisten ini juga terlihat pada set validasi. *Val/box_loss* turun dari sekitar 0,784 menjadi 0,583, *sementara val/cls_loss* turun dari 2,116 menjadi 0,281, dan *val/dfl_loss* turun dari 1,336 menjadi 1,047. Pengurangan bertahap di semua metrik menegaskan bahwa model tidak hanya dapat belajar secara efektif dari data pelatihan tetapi juga dapat menggeneralisasi dengan baik untuk data validasi.

YOLOv8n



Gambar 4.13 Grafik Loss YOLOv8n

Pada proses pelatihan YOLOv12n memakan waktu 0.639 jam dan *epoch* dengan hasil terbaik adalah *epoch* ke-89. Kemajuan pelatihan model YOLOv8n, seperti yang disajikan dalam Gambar 4.13, menyoroti peningkatan kinerjanya selama 100 *epoch*. Pada tahap awal, model ini menghasilkan nilai *train/box_loss* sebesar 1,068, *train/cls_loss* sebesar 3,319, dan *train/dfl_loss* sebesar 1,488. Nilai tersebut menunjukkan bahwa model ini masih dalam proses belajar untuk mendeteksi batas-batas objek, memberikan label emosi, dan memodelkan fitur-fitur yang berhubungan dengan jarak. Seiring berjalannya waktu, kerugian ini menunjukkan lintasan penurunan yang konsisten, mencapai sekitar 0,365 *train/box_loss*, 0,187 *train/cls_loss*, dan 0,910 *train/dfl_loss* pada *epoch* terakhir, yang menunjukkan efektivitas model yang semakin meningkat.

Tren paralel terjadi pada data validasi, di mana *val/box_loss* menurun dari sekitar 0,783 menjadi 0,578, *val/cls_loss* turun dari 2,477 menjadi 0,302, dan *val/dfl_loss* menurun dari

1,274 menjadi 1,050. Penurunan yang stabil pada nilai-nilai ini menunjukkan bahwa model dapat menerapkan apa yang dipelajari pada data yang sebelumnya tidak terlihat tanpa *overfitting*. Pola keseluruhan ini menunjukkan proses pembelajaran yang kuat dan kemampuan model untuk menggeneralisasi dengan baik di luar set pelatihan.

4.6.2 Perbandingan hasil loss

Tabel 4.2 Perbandingan Hasil Loss

Model		Box_loss	Cls_loss	Dfl_loss
YOLOv12	Train	0.396	0.197	0.921
	Val	0.563	0.261	1.049
YOLOv11	Train	0.396	0.193	0.925
	Val	0.574	0.299	1.051
YOLOv10	Train	0.753	0.322	1.829
	Val	1.159	0.556	2.124
YOLOv9	Train	0.403	0.177	0.934
	Val	0.583	0.281	1.047
YOLOv8	Train	0.365	0.187	0.910
	Val	0.578	0.302	1.050

Berdasarkan Tabel 4.2 analisis nilai loss antara fase pelatihan dan validasi menunjukkan perbedaan performa yang signifikan pada setiap varian model YOLO. YOLOv8 tercatat memiliki nilai Box_loss dan dfl_loss terendah pada tahap pelatihan senilai 0.365 dan 0.910, yang mengindikasikan kemampuan yang baik dalam mempelajari lokasi *bounding box*. Namun, terjadi peningkatan nilai menjadi 0.578 pada fase validasi, yang menandakan adanya sedikit kecenderungan *overfitting* saat model dihadapkan pada data baru. Serupa dengan itu, YOLOv9 menunjukkan keunggulan pada tahap pelatihan dengan Cls_loss terendah senilai 0.177, yang mencerminkan efisiensi tinggi dalam mengklasifikasikan objek. Meskipun demikian, nilai validasinya juga meningkat menjadi 0.281. Di sisi lain, YOLOv10 menunjukkan performa yang paling tidak optimal, dengan nilai *loss* yang jauh lebih tinggi di semua metrik, terutama pada Dfl_loss pelatihan senilai 1.829, yang mengindikasikan kesulitan dalam memprediksi distribusi *bounding box* secara akurat.

Secara keseluruhan, YOLOv12 tampil sebagai model yang paling seimbang dan berkinerja terbaik. Model ini berhasil mencapai nilai Box_loss 0.563 dan Cls_loss senilai 0.261 terendah

pada fase validasi, yang menegaskan kemampuannya untuk generalisasi dengan baik pada data yang belum pernah dilihat sebelumnya. Kesenjangan yang tidak terlalu jauh antara nilai pelatihan dan validasi juga menunjukkan bahwa YOLOv12 memiliki resistensi yang lebih baik terhadap *overfitting* dibandingkan varian lainnya. Dengan demikian, meskipun model lain seperti YOLOv8 dan YOLOv9 unggul pada aspek spesifik selama pelatihan, YOLOv12 terbukti menjadi model yang paling andal dan konsisten.

4.7 Tahap Pengujian

4.7.1 Hasil Pengujian

YOLOv12n

```
Validating /content/drive/MyDrive/yolov12n/train4/weights/best.pt...
Ultralytics 8.3.88 🚀 Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla T4, 15095MiB)
YOLOv12n summary (fused): 159 layers, 2,558,093 parameters, 0 gradients, 6.3 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95):
                   all        293        293      0.977      0.986      0.993      0.871
                     0         42         42      0.975      0.928      0.989      0.856
                     1         42         42      0.992          1      0.995      0.888
                     2         42         42      0.967          1      0.995      0.871
                     3         42         42      0.991          1      0.995       0.89
                     4         41         41      0.998          1      0.995      0.857
                     5         42         42          1      0.996      0.995      0.883
                     6         42         42      0.919      0.976       0.99      0.852
Speed: 0.2ms preprocess, 2.9ms inference, 0.0ms loss, 4.9ms postprocess per image
```

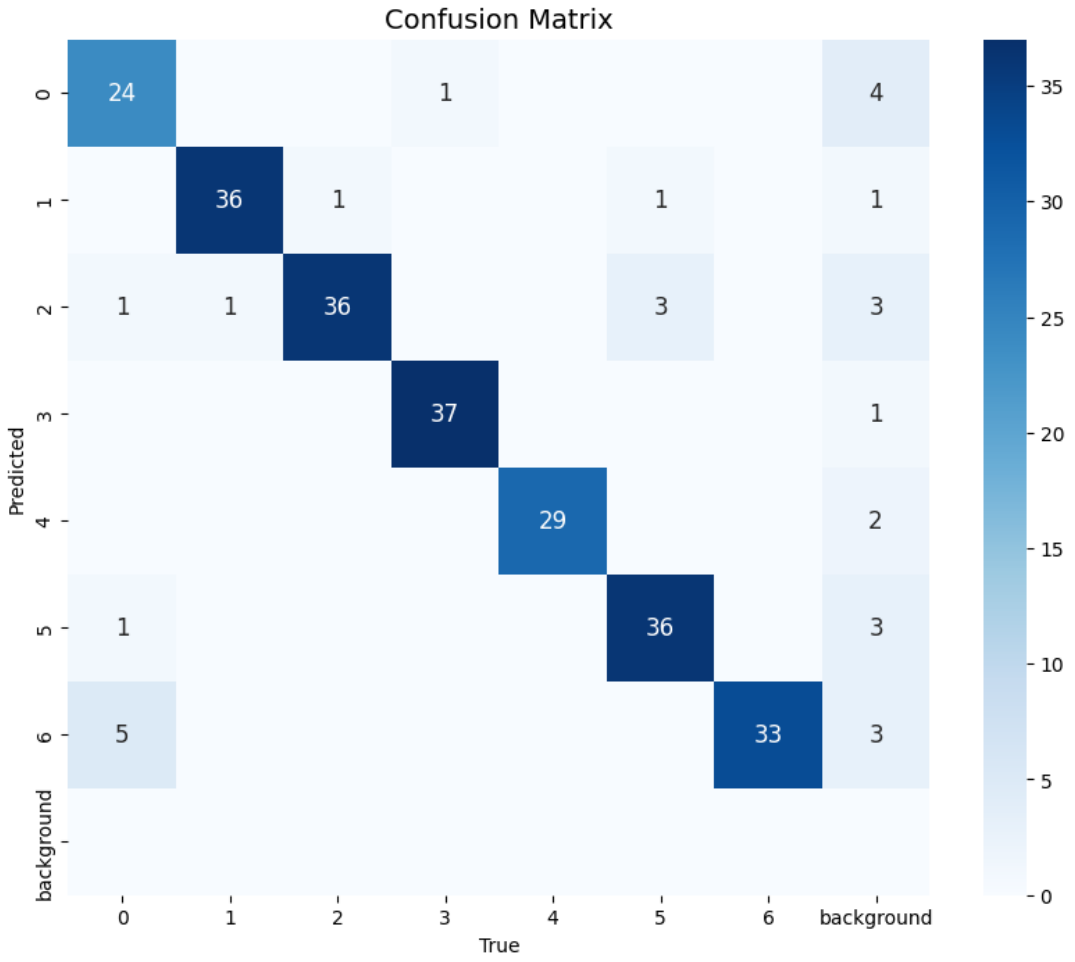
Gambar 4.14 Hasil Pengujian YOLOv12n

Berdasarkan Gambar 4.14 model YOLOv12n dirancang dengan 159 lapisan dan total 2.558.093 *parameter*, yang menawarkan keseimbangan antara kompleksitas arsitektur dan efisiensi komputasi. Berdasarkan hasil validasi, model ini mencapai nilai *recall* tertinggi yaitu 0,986 dibandingkan dengan model lain dalam percobaan. Hal ini menunjukkan kemampuan YOLOv12n yang sangat baik dalam mendeteksi objek yang ada di dalam gambar validasi.

Dalam hal presisi deteksi, model ini memperoleh mAP50 sebesar 0,993 dan mAP50:95 sebesar 0,871. Nilai mAP50 yang hampir sempurna menunjukkan bahwa YOLOv12n dapat menghasilkan prediksi *bounding box* yang sangat tepat pada *Thresshold* 0.5 Sementara itu, nilai mAP50:95 yang tinggi secara konsisten menegaskan kinerja deteksi model yang dapat diandalkan di berbagai tingkat *threshold*.

Mengenai kecepatan inferensi, YOLOv12n mencapai waktu inferensi 2,9 ms per gambar. Meskipun bukan yang tercepat di antara model yang dibandingkan, waktu pemrosesan ini tetap kompetitif untuk tugas pendeteksian objek *real-time*. Kombinasi dari jumlah lapisan yang efisien, ukuran *parameter* yang optimal, akurasi deteksi yang tinggi, dan kecepatan inferensi

yang dapat diterima menempatkan YOLOv12n sebagai model dengan kinerja terbaik dalam penelitian ini.

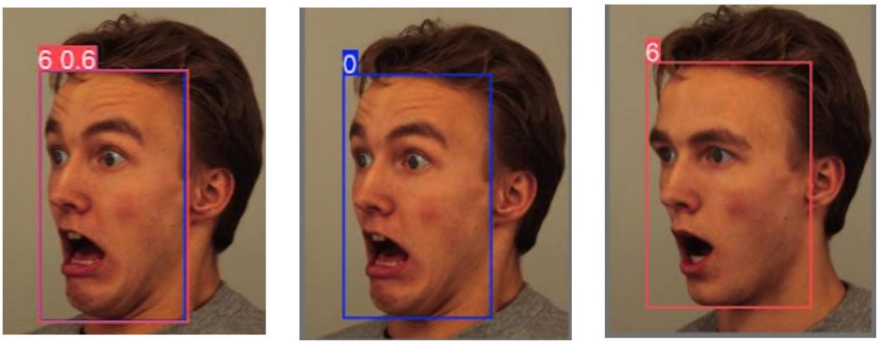


Gambar 4.15 Confusion Matrix pada YOLOv12n

Confusion matrix pada Gambar 4.15 mengilustrasikan kinerja klasifikasi model YOLOv12n dalam mendeteksi tujuh kategori emosi beserta kelas *background*. Garis diagonal menunjukkan jumlah prediksi yang benar, sementara selain di luar diagonal merepresentasikan kesalahan klasifikasi. Secara umum, model ini menunjukkan performa yang sangat baik dengan akurasi tinggi pada sebagian besar kelas, misalnya 37 prediksi benar untuk kelas 3 (bahagia), 36 untuk kelas 1 (marah), 36 untuk kelas 2 (jijik), dan 36 untuk kelas 5 (sedih). Meski demikian, masih terdapat sejumlah kesalahan yang patut diperhatikan. Kelas 0 (takut) relatif sulit dikenali secara konsisten, karena terdapat 5 contoh yang salah diprediksi sebagai kelas 6 (kaget), 4 contoh sebagai *backgroud*, dan 1 contoh sebagai kelas 5 (sedih) dan 2 (jijik). Kelas 2 (jijik) juga mencatat 2 kesalahan, di mana 1 contoh salah diprediksi sebagai latar belakang dan 1 contoh sebagai kelas 1 (marah). Kelas 3 (bahagia) juga mencatat 2 kesalahan, di mana 1 contoh

salah diprediksi sebagai latar belakang dan 1 contoh sebagai kelas 0 (takut). Pada kelas 4 (netral), terjadi 2 kesalahan ke dekteksi ke *background*. Untuk kelas 5 (sedih), terdapat 7 kesalahan, di mana 3 contoh salah diprediksi sebagai *backgorund*, 3 contoh sebagai kelas 2 (jijik) dan 1 contoh sebagai kelas 1 (marah). Terakhir kelas 6 (kaget) mencatat 3 kesalahan sebagai *background*.

Secara keseluruhan, hasil ini memperlihatkan bahwa YOLOv12n memiliki kemampuan yang kuat dalam mengenali emosi dengan tingkat kesalahan yang rendah. Pola kesalahan sebagian besar muncul pada kelas-kelas dengan kemiripan ekspresi, terutama antara takut dan kaget atau antara netral dan kelas lain. Dapat dilihat pad Gambar 4.16 yang merupakan contoh salah klasifikasi antara kelas 0 (takut) ke kelas 6 (kaget). Dimana ada sedikit perbedaan antara kedua kelas yaitu sedikit kerutan di dahi dan dagu. Dengan demikian, meskipun performanya konsisten tinggi pada kelas dominan seperti bahagia, marah, jijik, dan sedih, tantangan tetap ada pada ekspresi samar atau berdekatan, yang merupakan area potensial untuk pengembangan lebih lanjut.



Gambar 4.16 Contoh Kesalahan Prediksi pada YOLOv12n

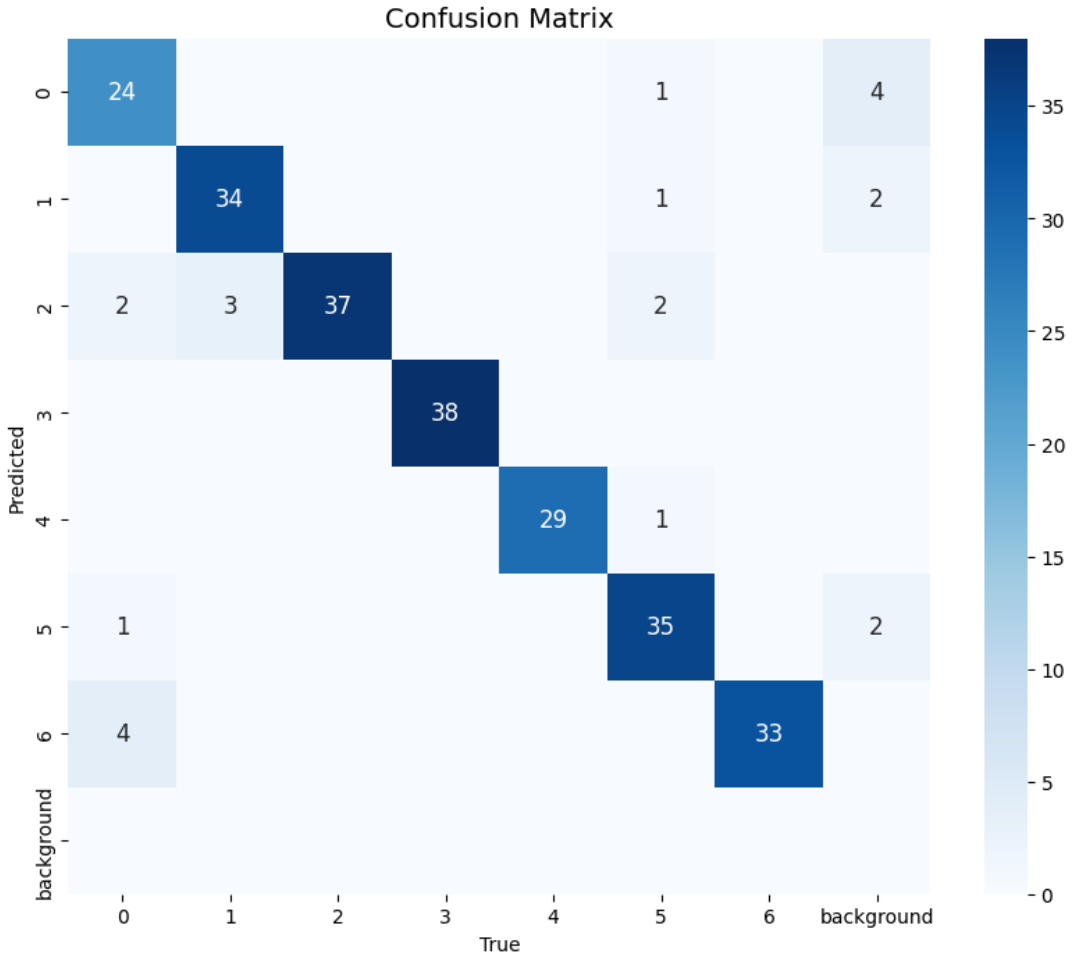
YOLOv11n

```
Validating /content/drive/MyDrive/yolov11n/train4/weights/best.pt...
Ultralytics 8.3.88 🚀 Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla T4, 15095MiB)
YOLO11n summary (fused): 100 layers, 2,583,517 parameters, 0 gradients, 6.3 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95):
                   all        293        293      0.936      0.967      0.989      0.868
                     0         42         42      0.906      0.922      0.975      0.842
                     1         42         42      0.933      0.998      0.992       0.88
                     2         42         42      0.881          1      0.992      0.872
                     3         42         42      0.955      0.976      0.987      0.874
                     4         41         41      0.976      0.992      0.994      0.881
                     5         42         42          1      0.884      0.991      0.869
                     6         42         42        0.9          1       0.99      0.859
Speed: 0.2ms preprocess, 2.2ms inference, 0.0ms loss, 6.6ms postprocess per image
```

Gambar 4.17 Hasil Pengujian YOLOv11n

Berdasarkan Gambar 4.16 model YOLOv11n, yang terdiri dari 100 lapisan dan total 2.583.517 *parameter*, dievaluasi pada set data validasi yang terdiri dari 293 gambar yang didistribusikan di tujuh kelas. Model ini mencapai nilai *recall* sebesar 0,967, yang mencerminkan kemampuan yang kuat untuk secara akurat mendeteksi sebagian besar contoh positif dalam set data. Dalam hal ketepatan deteksi, model memperoleh mAP50 sebesar 0,989, sedangkan mAP50:95 mencapai 0,868. Metrik ini menunjukkan bahwa model YOLOv11n memberikan kinerja deteksi yang tinggi secara konsisten, meskipun dengan sedikit penurunan ketika ambang batas menjadi lebih ketat.

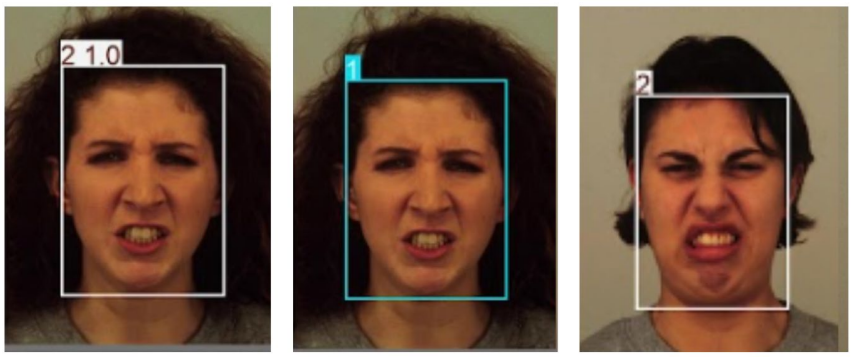
Selain itu, model ini menunjukkan kinerja komputasi yang efisien, mencapai waktu inferensi 2,2 ms per gambar, sehingga cocok untuk aplikasi pendeteksian objek secara *real-time*. Jika dibandingkan dengan model YOLOv12n, yang mencatat *recall* yang lebih tinggi tetapi waktu inferensi yang lebih lambat yaitu 2,9 ms per gambar, YOLOv11n menyajikan keseimbangan yang baik antara kinerja dan efisiensi pemrosesan. Pertukaran ini memposisikan YOLOv11n sebagai pilihan optimal untuk sistem yang membutuhkan hasil deteksi yang andal dan prediksi latensi rendah.



Gambar 4.18 Confusion Matrix pada YOLOv11n

Confusion matrix pada Gambar 4.13 menyajikan kinerja klasifikasi model YOLOv11n setelah pelatihan pada tugas pengenalan emosi multi-kelas. Sel diagonal menunjukkan jumlah prediksi yang benar, sedangkan sel di luar diagonal menggambarkan kesalahan klasifikasi. Secara umum, model mempertahankan akurasi yang tinggi, terutama pada kelas 3 (bahagia) dengan 38 prediksi benar, kelas 2 (jijik) dengan 37 prediksi benar, serta kelas 5 (sedih) dengan 35 prediksi benar. Namun demikian, terdapat beberapa pola kesalahan yang perlu diperhatikan. Kelas 0 (takut) relatif rentan salah dikenali, dengan total 4 contoh diklasifikasikan sebagai kelas 6 (kaget) dan 2 contoh lainnya salah terdeteksi sebagai kelas 2 (jijik). Hal ini menunjukkan adanya kedekatan fitur visual antara ekspresi takut dengan kaget maupun jijik. Kelas 1 (marah) juga mencatat tiga kesalahan yang semuanya diklasifikasikan sebagai jijik, mengindikasikan adanya kebingungan antar-kelas pada ekspresi negatif. Selain itu, kelas 4 (netral) menyumbang dua kesalahan, masing-masing diklasifikasikan sebagai bahagia dan sedih, yang dapat dipengaruhi oleh kesamaan ekspresi wajah dengan intensitas rendah. Sementara itu, kelas 5 (sedih) selain memiliki dua kesalahan yang diprediksi sebagai latar belakang, juga mengalami satu kesalahan yang terdeteksi sebagai kelas 0 (takut).

Secara keseluruhan, hasil ini memperlihatkan bahwa meskipun YOLOv11n mampu mengklasifikasikan sebagian besar emosi dengan baik, masih terdapat kecenderungan kesalahan pada kelas-kelas dengan ekspresi mirip atau samar, terutama antara takut, kaget, jijik, dan sedih. Dapat dilihat pada Gambar 4.19 yang merupakan contoh salah klasifikasi antara kelas 1 (marah) ke kelas 2 (jijik). Dimana ada banyak persamaan antara kedua kelas yaitu di bibir, garis pipi dan alis. Temuan ini penting untuk evaluasi lebih lanjut, khususnya dalam meningkatkan kemampuan model membedakan ekspresi yang memiliki kesamaan fitur visual.



Gambar 4.19 Contoh Kesalahan Prediksi pada YOLOv11n

YOLO10n

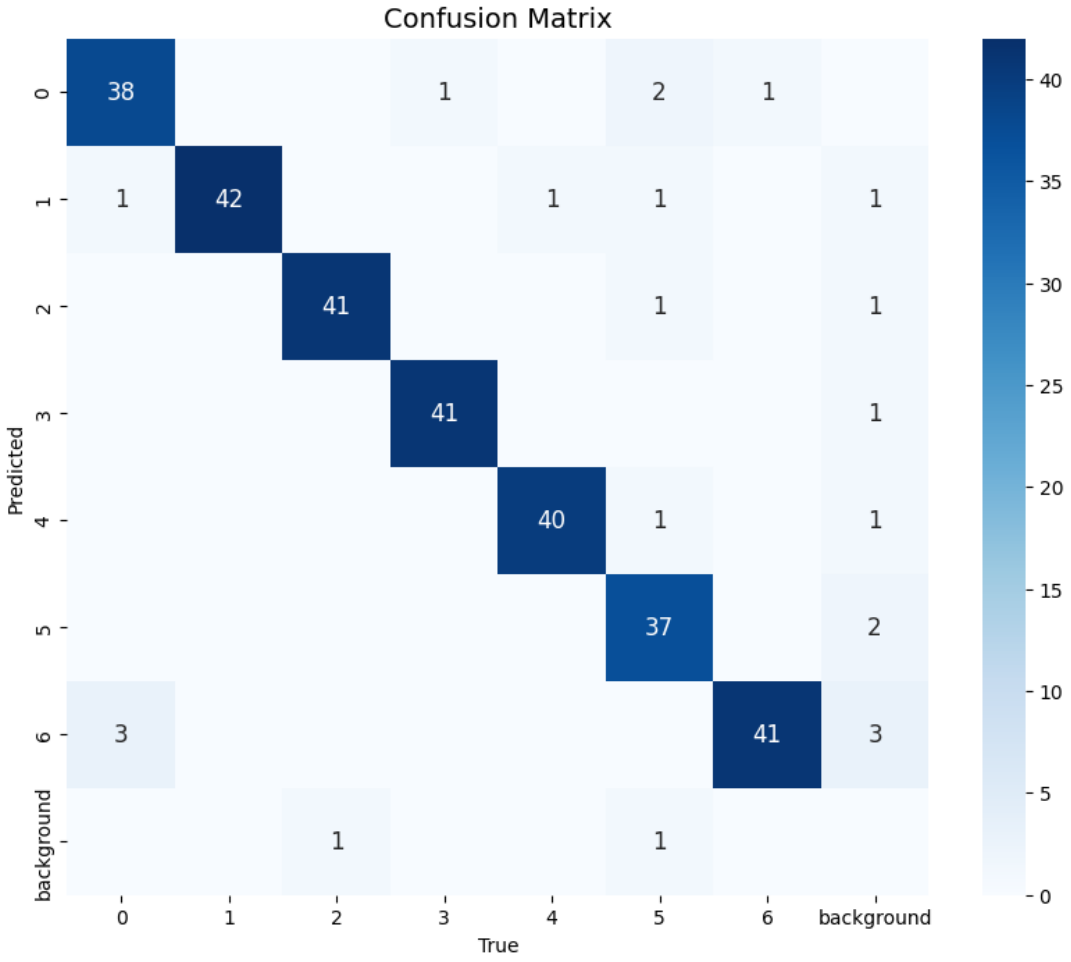
```
Validating /content/drive/MyDrive/yolov10n/train4/weights/best.pt...
Ultralytics 8.3.88 🚀 Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla T4, 15095MiB)
YOLOv10n summary (fused): 125 layers, 2,697,146 parameters, 0 gradients, 8.2 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95):
                   all        293        293      0.982      0.932      0.988      0.866
                     0         42         42      0.993      0.857      0.976      0.845
                     1         42         42       0.96          1      0.992      0.878
                     2         42         42      0.974      0.895      0.991      0.871
                     3         42         42      0.995      0.976      0.993      0.883
                     4         41         41      0.991      0.976      0.993      0.878
                     5         42         42          1      0.821       0.98      0.849
                     6         42         42      0.957          1      0.991      0.857
Speed: 0.2ms preprocess, 2.3ms inference, 0.0ms loss, 0.9ms postprocess per image
```

Gambar 4.20 Hasil Pengujian YOLOv10n

Berdasarkan Gambar 4.18 model YOLOv10n dievaluasi menggunakan dataset validasi yang berisi 293 gambar di 7 kelas. Model ini terdiri dari 125 lapisan dan sekitar 2.697.146 *parameter*. Hasil pengujian menunjukkan bahwa model tersebut mencapai *recall* sebesar 0,932, dan mAP50 sebesar 0,988. Sementara itu, mAP50:95 mencapai 0,866.

Ketika memeriksa kinerja per-kelas, model menampilkan nilai *recall* tertinggi sebesar 0,976 untuk kelas 3 (bahagia) dan 4, dengan *recall* terendah yang di kelas 5 (sedih) sebesar 0,821. Nilai mAP50 per kelas secara konsisten melebihi 0,976, menunjukkan presisi deteksi yang kuat di sebagian besar kategori, meskipun nilai mAP50:95 sedikit bervariasi, dengan kelas 0 (takut) memperoleh 0,845 dan kelas 3 (bahagia) mencapai yang tertinggi di 0,883.

Dalam hal kecepatan inferensi, model ini memproses setiap gambar dalam 2,3ms, menyeimbangkan presisi deteksi dengan kinerja komputasi yang efisien. Dengan demikian, model ini menunjukkan pertukaran yang menguntungkan antara presisi dan waktu inferensi, sehingga cocok untuk aplikasi deteksi *real-time* di mana kompleksitas model yang cukup tinggi dan pemrosesan yang cepat diperlukan.



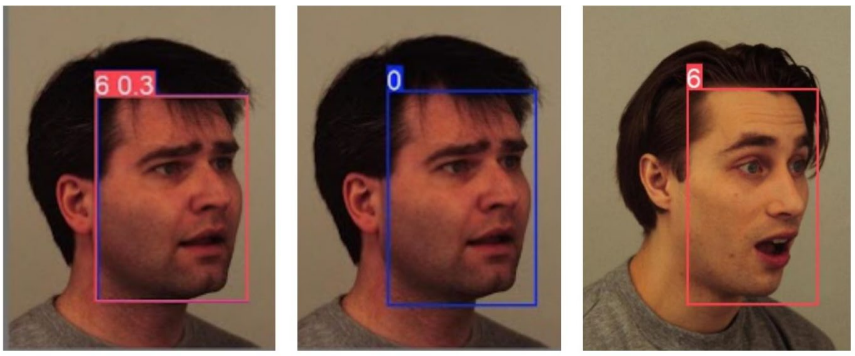
Gambar 4.21 Confusion Matrix pada YOLOv10n

Confusion matrix pada Gambar 4.19 merepresentasikan kinerja model YOLOv10n dalam tugas pendeteksian emosi multi-kelas. Sel diagonal menunjukkan jumlah prediksi yang benar, sedangkan sel di luar diagonal menggambarkan kesalahan klasifikasi. Secara umum, model ini memperlihatkan performa yang konsisten tinggi di hampir semua kategori, dengan 42 prediksi benar pada kelas 1 (marah), 41 pada kelas 2 (jijik), 41 pada kelas 3 (bahagia), 40 pada kelas 4 (netral), 41 pada kelas 6 (kaget), 38 pada kelas 0 (takut), dan 37 pada kelas 5 (sedih). Hal ini menegaskan bahwa model mampu mengklasifikasikan sebagian besar sampel secara akurat.

Meski demikian, terdapat sejumlah kesalahan klasifikasi yang dapat diperhatikan. Kelas 0 (takut) masih cukup rentan, dengan 3 contoh salah terdeteksi sebagai kelas 6 (kaget), 2 contoh sebagai kelas 4 (netral), dan 1 contoh sebagai latar belakang. Kesalahan serupa juga muncul pada kelas 6 (kaget), di mana 3 contoh salah diklasifikasikan sebagai kelas 0 (takut) dan 1 contoh sebagai kelas 4 (netral). Kelas 5 (sedih) memiliki dua kesalahan yang diklasifikasikan sebagai kelas 6 (kaget) dan satu kesalahan sebagai latar belakang, sementara kelas 2 (jijik)

mencatat satu kesalahan yang diprediksi sebagai latar belakang. Selain itu, beberapa kesalahan kecil muncul pada kelas 1 (marah) dan 4 (netral) yang masing-masing memiliki satu contoh salah klasifikasi ke kelas lain.

Secara keseluruhan, hasil ini menunjukkan bahwa YOLOv10n memiliki ketangguhan yang tinggi dalam menjaga konsistensi klasifikasi lintas kelas dengan tingkat kesalahan yang relatif rendah. Pola kesalahan terutama terjadi pada pasangan ekspresi yang memiliki kemiripan visual, seperti takut dan sedih dengan kaget, yang menyoroti tantangan utama dalam membedakan emosi dengan ciri ekspresi halus. Dapat dilihat pada Gambar 4.22 yang merupakan contoh salah klasifikasi antara kelas 0 (takut) ke kelas 6 (kaget). Dimana ada sedikit perbedaan antara kedua kelas yaitu kerutan di pipi dan mulut terbuka. Dengan demikian, meskipun masih terdapat sedikit kesalahan, model ini dapat dianggap andal dan lebih unggul dibandingkan versi sebelumnya.



Gambar 4.22 Contoh Kesalahan Prediksi pada YOLOv10n

YOLOv9t

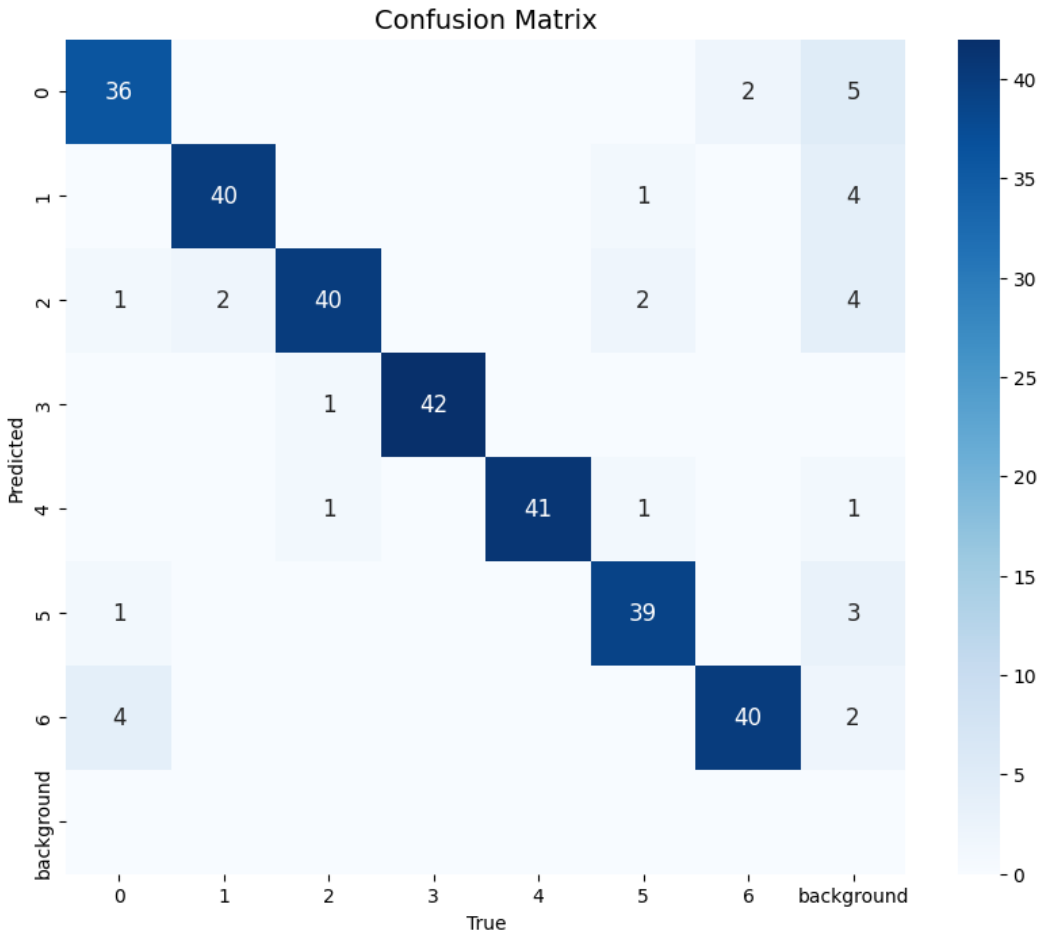
```
Validating /content/drive/MyDrive/yolov9t/train3/weights/best.pt...
Ultralytics 8.3.88 🚀 Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla T4, 15095MiB)
YOLOv9t summary (fused): 197 layers, 1,972,149 parameters, 0 gradients, 7.6 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95):
                   all        293        293      0.953      0.957      0.987      0.871
                     0         42         42      0.948      0.881      0.977      0.851
                     1         42         42      0.976      0.975      0.992      0.873
                     2         42         42      0.912      0.976      0.973      0.854
                     3         42         42      0.974          1      0.995      0.896
                     4         41         41      0.976      0.976      0.995      0.883
                     5         42         42      0.975      0.938       0.99      0.878
                     6         42         42      0.913      0.952      0.987      0.861
Speed: 0.2ms preprocess, 3.3ms inference, 0.0ms loss, 5.4ms postprocess per image
```

Gambar 4.23 Hasil Pengujian YOLOv9t

Berdasarkan Gambar 4.20 model YOLOv9t divalidasi pada dataset yang berisi 293 gambar yang didistribusikan ke dalam 7 kelas objek. Arsitektur model ini terdiri dari 197 lapisan dan sekitar 1.972.149 *parameter*. Berdasarkan evaluasi, YOLOv9t mencapai presisi

keseluruhan sebesar 0,953 dan *recall* sebesar 0,957. Model ini juga mencatat mAP50 sebesar 0,987 mAP50:95 sebesar 0,871.

Dalam hal kinerja per kelas, nilai *recall* tertinggi dicapai pada kelas 3 (bahagia), mencapai 1,000, sedangkan yang terendah pada kelas 0 (takut) sebesar 0,881. Nilai mAP50 per kelas menunjukkan nilai yang konsisten di atas 0,977, dengan mAP50 tertinggi tercatat di kelas 3 (bahagia) sebesar 0,995. Nilai mAP50:95 bervariasi, dengan nilai terbaik 0,896 juga ditemukan di kelas 3 (bahagia) dan terendah di kelas 2 (jijik) pada 0,854, yang menunjukkan kinerja deteksi yang relatif stabil di seluruh kategori. Mengenai kecepatan inferensi, YOLOv9t memproses gambar dalam waktu sekitar 3,3 ms per gambar. Waktu pemrosesan ini lebih tinggi daripada varian model lain yang dievaluasi dalam studi yang sama, tetapi diimbangi dengan tingkat *recall* dan presisi yang kompetitif. Oleh karena itu, YOLOv9t hadir sebagai pilihan yang layak untuk aplikasi yang memprioritaskan kinerja deteksi dengan tetap mempertahankan efisiensi komputasi yang moderat.



Gambar 4.24 Confusion Matrix pada YOLOv9t

Confusion matrix pada Gambar 4.21 mengilustrasikan kinerja klasifikasi model YOLOv9t dalam mendeteksi emosi multi-kelas. Model ini menunjukkan akurasi tinggi di sebagian besar kategori, dengan 42 prediksi benar pada kelas 3 (bahagia), 41 pada kelas 4 (netral), 40 pada kelas 1 (marah) dan kelas 2 (jijik), 40 pada kelas 6 (kaget), 39 pada kelas 5 (sedih), serta 36 pada kelas 0 (takut). Walaupun performanya cukup konsisten, terdapat beberapa kesalahan klasifikasi yang perlu diperhatikan. Kelas 0 (takut) cukup rentan, dengan 5 contoh salah dikenali sebagai kelas 6 (kaget) dan 2 contoh lainnya sebagai kelas 4 (netral). Kelas 2 (jijik) juga mencatat kebingungan dengan 2 contoh salah diprediksi sebagai kelas 6 (kaget), serta 2 contoh lainnya sebagai kelas 0 (takut). Kelas 1 (marah) memiliki 1 kesalahan ke kelas 2 (jijik) dan 4 kesalahan ke kelas 6 (kaget). Kelas 5 (sedih) mencatat kesalahan yang tersebar, dengan 1 contoh salah ke kelas 0 (takut) dan 3 contoh ke kelas 6 (kaget).

Selain itu, kelas 6 (kaget) sendiri masih memiliki 2 kesalahan yang salah diklasifikasikan sebagai kelas 5 (sedih). Dapat dilihat pada Gambar 4.25 yang merupakan contoh salah klasifikasi antara kelas 5 (sedih) ke kelas 0 (takut). Dimana ada banyak persamaan antara kedua kelas yaitu kerutan di pipi, alis dan bibir.

Gambar 4.25 Contoh Kesalahan Prediksi pada YOLOv9t

Secara keseluruhan, hasil ini menunjukkan bahwa YOLOv9t memiliki performa yang cukup baik dalam mengenali berbagai kategori emosi, dengan tingkat akurasi yang tinggi pada kelas dominan seperti bahagia dan netral. Namun, pola kesalahan menunjukkan kecenderungan kebingungan antar-ekspresi yang memiliki kemiripan visual, terutama antara takut dengan kaget serta jijik dengan kaget. Jika dibandingkan dengan YOLOv10n, YOLOv11n, dan YOLOv12n, model ini masih menghasilkan jumlah kesalahan yang relatif lebih banyak, sehingga meskipun konsisten, kestabilannya sedikit lebih rendah dibandingkan varian yang lebih baru.

YOLOv8n

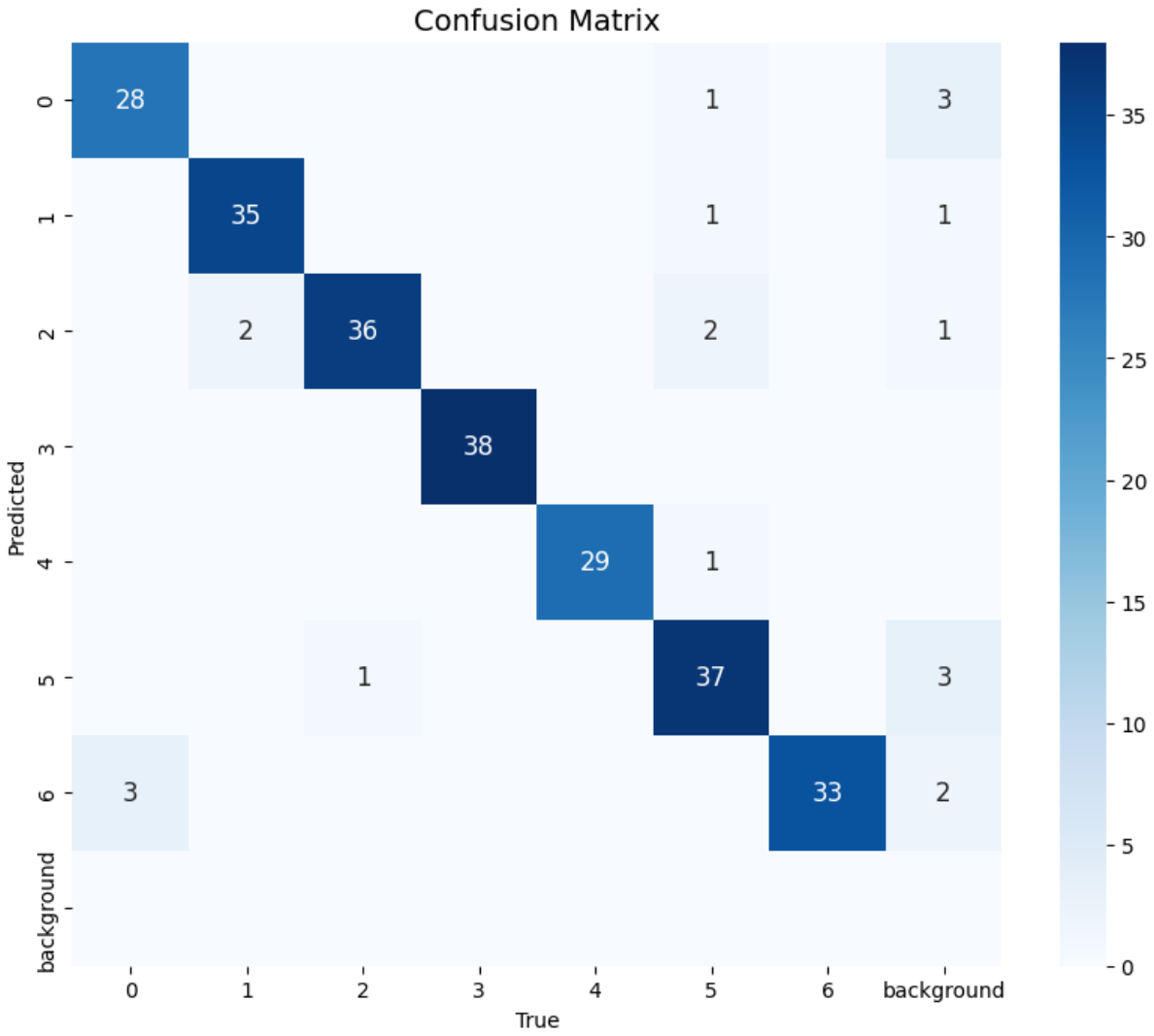
```
Validating /content/drive/MyDrive/yolov8n/train4/weights/best.pt...
Ultralytics 8.3.88 🚀 Python-3.11.11 torch-2.5.1+cu124 CUDA:0 (Tesla T4, 15095MiB)
Model summary (fused): 72 layers, 3,007,013 parameters, 0 gradients, 8.1 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95):
                   all        293        293      0.967      0.946      0.988      0.863
                     0         42         42      0.974      0.901      0.982      0.845
                     1         42         42      0.999      0.952      0.991      0.874
                     2         42         42       0.91      0.976      0.986      0.876
                     3         42         42      0.999          1      0.995      0.866
                     4         41         41          1      0.958      0.995       0.87
                     5         42         42       0.94      0.929       0.98      0.845
                     6         42         42       0.95      0.909      0.989      0.862
Speed: 0.2ms preprocess, 1.7ms inference, 0.0ms loss, 4.3ms postprocess per image
```

Gambar 4.26 Hasil Pengujian YOLOv8n

Berdasarkan Gambar 4.22 model YOLOv8n divalidasi pada set data yang berisi 293 gambar yang didistribusikan di 7 kelas objek. Arsitekturnya relatif ringan, terdiri dari 72 lapisan dan sekitar 3.007.013 *parameter*. Selama pengujian, YOLOv8n mencapai presisi keseluruhan sebesar 0,967 dan *recall* sebesar 0,946. Dalam hal presisi, model mencatat mAP50 sebesar 0,988 *dan* mAP50:95 sebesar 0,863.

Untuk kinerja per kelas, model menunjukkan nilai presisi dan *recall* yang tinggi, terutama di kelas 3 (bahagia) di mana ia mencapai *precision* 0,999 dan *recall* 1,000. Namun, *recall* terendah pada kelas 6 (kaget) dengan nilai 0.090, yang mengindikasikan penurunan yang signifikan dalam kemampuan deteksi untuk kelas ini. Meskipun demikian, sebagian besar kelas lainnya mencapai nilai *recall* di atas 0,90, dengan nilai mAP50 yang tinggi secara konsisten di atas 0,988. Nilai mAP50:95 menunjukkan sedikit variasi di seluruh kelas, dengan nilai tertinggi 0,876 ditemukan di kelas 2 (jijik).

Dari sudut pandang efisiensi, YOLOv8n menunjukkan waktu inferensi tercepat di antara semua model yang dievaluasi, dengan rata-rata 1,7ms per gambar. Hal ini menyoroti model YOLOv8n sebagai pilihan optimal untuk aplikasi yang membutuhkan kecepatan pemrosesan yang cepat, meskipun kinerjanya pada kelas tertentu seperti kelas 6 (kaget) harus dipertimbangkan ketika menilai kesesuaian untuk diterapkan dalam sistem *real-time*.



Gambar 4.27 Confusion Matrix pada YOLOv8n

Confusion matrix pada Gambar 4.23 memperlihatkan performa model YOLOv8n dalam mendeteksi emosi multi-kelas. Model ini mampu mengklasifikasikan dengan benar 38 contoh pada kelas 3 (bahagia), 37 pada kelas 5 (sedih), 36 pada kelas 2 (jijik), 35 pada kelas 1 (marah), 33 pada kelas 6 (kaget), 29 pada kelas 4 (netral), dan 28 pada kelas 0 (takut). Meskipun hasil ini menunjukkan kemampuan dasar dalam mengenali ekspresi wajah, sejumlah kesalahan klasifikasi masih cukup terlihat. Pada kelas 0 (takut), terdapat 3 contoh yang salah dikenali sebagai latar belakang dan 1 contoh salah ke kelas 4 (netral). Kelas 1 (marah) relatif stabil, hanya memiliki 1 kesalahan ke kelas 2 (jijik). Kelas 2 (jijik) mengalami 2 kesalahan ke kelas 0 (takut) dan 1 kesalahan ke kelas 1 (marah). Untuk kelas 4 (netral), terdapat 1 contoh yang salah diprediksi sebagai kelas 3 (bahagia), sedangkan kelas 5 (sedih) salah diklasifikasikan sebanyak 3 contoh ke kelas 6 (kaget) dan 1 contoh ke kelas 2 (jijik). Selain itu, kelas 6 (kaget)

juga mencatat 2 kesalahan ke kelas 5 (sedih) dan 3 contoh lainnya salah terdeteksi sebagai *background*. Dapat dilihat pada Gambar 4.28 yang merupakan contoh salah klasifikasi antara kelas 5 (sedih) ke kelas 2 (jijik). Dimana ada beberapa yaitu kerutan di pipi, alis, dagu dan dahi.

Gambar 4.28 Contoh Kesalahan Prediksi pada YOLOv8n

Secara keseluruhan, hasil ini menegaskan bahwa YOLOv8n memiliki akurasi yang lebih rendah dibandingkan dengan YOLOv9t, YOLOv10n, YOLOv11n, dan YOLOv12n. Kelemahan utamanya terletak pada kestabilan dalam mendeteksi kelas takut dan kaget, di mana tingkat kesalahannya lebih tinggi dibandingkan kelas lain. Hal ini menunjukkan bahwa versi awal YOLO ini kurang optimal dalam menangani dataset KDEF, terutama saat membedakan ekspresi emosional yang mirip atau samar.

4.7.2 Perbandingan Hasil Pengujian

Tabel 4.3 Perbandingan Model

model	layers	parameters	precision	recall	mAP50	mAP50:90	Inference time
YOLOv12	159	2,558,093	0.977	0.986	0.993	0.871	2.9ms
YOLOv11	100	2,583,517	0.936	0.967	0.989	0.868	2.2ms
YOLOv10	125	2,697,146	0.982	0.932	0.988	0.866	2.3ms
YOLOv9	197	1,972,149	0.953	0.957	0.987	0.871	3.3ms
YOLOv8	72	3,007,013	0.967	0.946	0.988	0.863	1.7ms

Tabel 4.2 menyajikan perbandingan lima model YOLO yang berbeda, termasuk YOLOv12, YOLOv11, YOLOv10, YOLOv9, dan YOLOv8. Berdasarkan sejumlah metrik

evaluasi seperti jumlah *layers*, *parameter*, presisi, *recall*, mAP50, mAP50:90, serta waktu inferensi. Secara umum, semua model menunjukkan kinerja yang baik untuk tugas deteksi *real-time* dengan presisi di atas 0,95 dan waktu inferensi di bawah 3 ms. Namun, terdapat perbedaan karakteristik yang menonjol pada masing-masing model. YOLOv10 mencatat presisi tertinggi sebesar 0,982, menandakan keunggulannya dalam menghasilkan prediksi yang akurat, meskipun *recall* (0,932) dan waktu inferensi (2,3 ms) masih kalah dibandingkan YOLOv12 dan YOLOv8. YOLOv12 sendiri menonjol dengan *recall* tertinggi (0,986) serta mAP50 tertinggi (0,993), menunjukkan konsistensinya dalam mendeteksi objek secara menyeluruh. Model ini juga relatif seimbang dengan presisi 0,977 dan waktu inferensi 2,9 ms, sehingga lebih stabil dibandingkan model lain.

Sebaliknya, YOLOv8 memiliki keunggulan pada kecepatan inferensi tercepat (1,7 ms) dan jumlah *layers* paling sedikit (72), sehingga cocok untuk implementasi pada perangkat dengan keterbatasan komputasi. Namun, nilai presisi (0,967) dan *recall* (0,946) sedikit lebih rendah dibandingkan versi terbaru. Dari sisi efisiensi, YOLOv9 menjadi model dengan jumlah parameter terkecil (1,97 juta), sehingga ideal untuk perangkat dengan sumber daya terbatas, meskipun presisi (0,953) dan recall (0,957) lebih rendah dibandingkan YOLOv12 dan YOLOv10. Sementara itu, YOLOv11 menunjukkan kinerja seimbang dengan presisi 0,975, *recall* 0,967, dan mAP50 sebesar 0,989, menjadikannya kompetitif dalam akurasi sekaligus efisien dalam inferensi.

Secara keseluruhan, persamaan yang dapat dilihat adalah semua model memiliki performa baik pada metrik utama sehingga layak untuk deteksi *real-time*. Perbedaannya terletak pada titik fokus keunggulan masing-masing: YOLOv8 cepat namun kurang akurat, YOLOv9 ringan namun presisinya terbatas, YOLOv10 sangat presisi namun mengorbankan *recall*, YOLOv11 seimbang di berbagai metrik, dan YOLOv12 paling unggul dalam generalisasi dengan kombinasi *recall* serta mAP tertinggi. Dengan demikian, kelemahan tiap model adalah konsekuensi dari trade-off antara akurasi dan kecepatan. Kelebihan YOLOv12 adalah keseimbangan optimal antara akurasi tinggi dan waktu inferensi yang masih efisien, sehingga direkomendasikan sebagai model paling sesuai untuk sistem deteksi emosi yang membutuhkan reliabilitas tinggi tanpa kehilangan performa *real-time*.

4.8 Tahap Inference

4.8.1 Hasil Inference pada Video RAVDESS

Pada sub bab ini Model YOLO dilakukan inference pada gambar dari *video* dengan ekspresi emosional dengan urutan deteksi dibuat dari kiri atas ke kanan berurutan sesuai pada tahap pelabelan.

YOLOv12n

Gambar 4.29 Hasil Inference YOLOv12n pada Gambar Video Ravdess

Berdasarkan Gambar 4.24 hasil pengujian model YOLOv12n pada tujuh gambar ekspresi wajah menunjukkan kinerja pendeteksian yang sangat akurat. Semua gambar berhasil diklasifikasikan ke dalam kategori emosi yang benar tanpa ada ketidakcocokan label. *Confident score* berkisar antara 0,65 hingga 0,94, dengan *confident score* tertinggi dicapai dalam mendeteksi ekspresi 3 (bahagia) dan 6 (kaget) sebesar 0,94. Sementara itu, *confident score* terendah pada ekspresi 0 (takut) sebesar 0,65, yang masih dalam ambang batas deteksi yang dapat diterima. Secara keseluruhan, model ini secara konsisten dan akurat mengidentifikasi setiap ekspresi wajah dalam gambar validasi. Hasil ini menunjukkan bahwa YOLOv12n memiliki kemampuan yang kuat dalam mengenali emosi wajah dan dapat diandalkan untuk tugas klasifikasi emosi berbasis gambar. Disarankan untuk terus mengevaluasi ambang batas *confident score* untuk menentukan nilai optimal untuk implementasi sistem *real-time.*

YOLOv11n

Gambar 4.30 Hasil Inference YOLOv11n pada Gambar Video Ravdess

Berdasarkan Gambar 4.25 hasil pengujian model YOLOv11n pada tujuh gambar ekspresi wajah menunjukkan kinerja deteksi yang akurat. Semua gambar diklasifikasikan dengan benar ke dalam kategori emosi masing-masing tanpa ada ketidakcocokan label. *Confident score* berkisar antara 0,56 hingga 0,94, dengan *confident score* tertinggi diperoleh dalam mendeteksi ekspresi 1 (marah) dan 2 (jijik) sebesar 0,94. Keyakinan terendah tercatat pada ekspresi 0 (takut) sebesar 0,56, yang masih berada dalam ambang batas deteksi yang dapat diterima. Secara keseluruhan, model ini menunjukkan deteksi yang konsisten dan dapat diandalkan dari ekspresi wajah dalam gambar validasi. Temuan ini menunjukkan bahwa YOLOv11n mampu mengenali emosi wajah secara efektif dan cocok untuk digunakan dalam tugas klasifikasi emosi berbasis gambar.

YOLOv10n

Gambar 4.31 Hasil Inference YOLOv10n pada Gambar Video Ravdess

Berdasarkan Gambar 4.26 hasil pengujian model YOLOv10n pada tujuh gambar ekspresi wajah menunjukkan kemampuan deteksi yang cukup baik, dengan satu kesalahan dalam mendeteksi emosi takut menjadi sedih. *Confident score* berkisar antara 0,70 hingga 0,97. Tingkat *confident score* tertinggi dicapai dalam mengenali ekspresi 1 (marah) sebesar 0,97, diikuti oleh ekspresi 2 (jijik) sebesar 0,94. *Confident score* deteksi terendah pada ekspresi 6 (kaget) pada 0,70. Secara keseluruhan, model YOLOv10n menunjukkan kinerja deteksi yang cukup konsisten. Model ini berhasil mempertahankan deteksi yang cukup akurat dengan satu ketidakcocokan di mana label ekspresi yang dilabeli sebagai 0 (takut) di deteksi sebagai 5 (sedih).

YOLOv9t

Gambar 4.32 Hasil Inference YOLOv9t pada Gambar Video Ravdess

Berdasarkan Gambar 4.27 performa model YOLOv9t dievaluasi dengan menggunakan tujuh gambar yang mewakili ekspresi wajah manusia yang berbeda. Model ini menunjukkan kemampuan yang secara umum dapat diandalkan untuk mendeteksi dan mengklasifikasikan ekspresi wajah, dengan *confident score* berkisar antara 0,69 hingga 0,96. *Confident score* tertinggi terdapat di ekspresi 3 (bahagia) sebesar 0,96, diikuti oleh 2 (jijik) sebesar 0,94, yang menunjukkan kinerja model yang kuat dalam mendeteksi emosi ini. Di sisi lain, *confident score* terendah tercatat untuk ekspresi "netral" sebesar 0,69 dan 6 (terkejut) sebesar 0,70, yang menunjukkan area potensial untuk perbaikan dalam mengenali kondisi emosi yang lebih halus atau ambigu. Terlepas dari hasil positif secara keseluruhan, ada kesalahan klasifikasi, di mana ekspresi yang dilabeli sebagai 0 (takut) di deteksi sebagai 1 (marah). Perbedaan tersebut menyoroti tantangan dalam membedakan antara ekspresi emosional yang terkait erat dan kebutuhan akan kemampuan klasifikasi yang lebih baik.

YOLOv8n

Gambar 4.33 Hasil Inference YOLOv8n pada Gambar Video Ravdess

Berdasarkan Gambar 4.28 hasil pengujian model YOLOv8n dievaluasi pada tujuh gambar ekspresi wajah yang mewakili berbagai kondisi emosional. Hasil pendeteksian menunjukkan kinerja yang secara umum kuat, dengan *confident score* berkisar antara 0,82 hingga 0,94. Tingkat *confident score* tertinggi dicapai dalam mengenali ekspresi 1 (marah), 2 (jijik) dan 4 (netral), masing-masing dengan *confident score* 0,94. Ekspresi 5 (sedih) dan 0 (takut) terdeteksi salah dalam mendeteksi di mana ekspresi yang dilabeli sebagai 0 (takut) dideteksi sebagai 1 (marah) dan ekspresi yang dilabeli sebagai 5 (sedih) dideteksi sebagai 0 (takut). Berdasarkan hasil ini model YOLOv8n termasuk terburuk dari model lainnya dengan 2 kesalahan deteksi.

Berdasarkan hasil *inference*, setiap model YOLO menunjukkan tingkat performa yang berbeda dalam mendeteksi emosi wajah. YOLOv12n merupakan model dengan performa paling baik karena mampu mengklasifikasikan seluruh ekspresi dengan benar tanpa kesalahan label. *Confident score* berkisar antara 0,65 hingga 0,94, dengan nilai tertinggi pada ekspresi 3 (bahagia) dan 6 (kaget) sebesar 0,94, sedangkan nilai terendah terdapat pada ekspresi 5 (sedih) sebesar 0,65. Pola ini menunjukkan bahwa meskipun akurasi tinggi, emosi takut masih menjadi tantangan bagi model. YOLOv11n juga memperlihatkan hasil yang konsisten, seluruh ekspresi berhasil terdeteksi tanpa kesalahan, dengan rentang *confident score* antara 0,56 hingga 0,94. Sama seperti YOLOv12n, kelemahan utama model ini juga terdapat pada ekspresi 5 (sedih) yang hanya mencapai skor 0,56 dan salah klasifikasi menjadi ekspresi 0 (takut).

Perbandingan Hasil inference pada RAVDESS

Berdasarkan hasil inference, setiap model YOLO menunjukkan tingkat performa yang berbeda dalam mendeteksi emosi wajah. YOLOv12n merupakan model dengan performa paling bagus karena mampu mengklasifikasikan seluruh ekspresi dengan benar tanpa kesalahan label. Skor kepercayaan berkisar antara 0,65 hingga 0,94, dengan nilai tertinggi pada ekspresi bahagia dan kaget (0,94), sedangkan nilai terendah terdapat pada ekspresi takut (0,65). Pola ini menunjukkan bahwa meskipun akurasi tinggi, emosi takut masih menjadi tantangan bagi model YOLOv12n. YOLOv11n juga memperlihatkan hasil yang konsisten, seluruh ekspresi berhasil terdeteksi tanpa kesalahan, dengan rentang skor kepercayaan antara 0,56 hingga 0,94. Sama seperti YOLOv12n, kelemahan utama model ini juga terdapat pada ekspresi takut yang hanya mencapai skor 0,56.

Berbeda dengan kedua model tersebut, YOLOv10n menampilkan kinerja cukup baik dengan rentang confident score antara 0,70 hingga 0,97. Model ini sangat kuat pada ekspresi marah (0,97) dan jijik (0,94), namun terjadi satu kesalahan klasifikasi ketika ekspresi takut salah dikenali sebagai sedih. YOLOv9t menunjukkan performa yang dapat diandalkan dengan skor kepercayaan hingga 0,96 untuk ekspresi bahagia dan 0,94 untuk jijik. Selain itu, terjadi kesalahan klasifikasi pada ekspresi takut yang terdeteksi sebagai marah. YOLOv8n menampilkan performa paling rendah, meskipun memiliki skor kepercayaan cukup tinggi pada beberapa ekspresi (hingga 0,94 untuk marah, jijik, dan netral), model ini melakukan dua kesalahan deteksi: ekspresi takut terklasifikasi sebagai marah dan ekspresi sedih terklasifikasi sebagai takut.

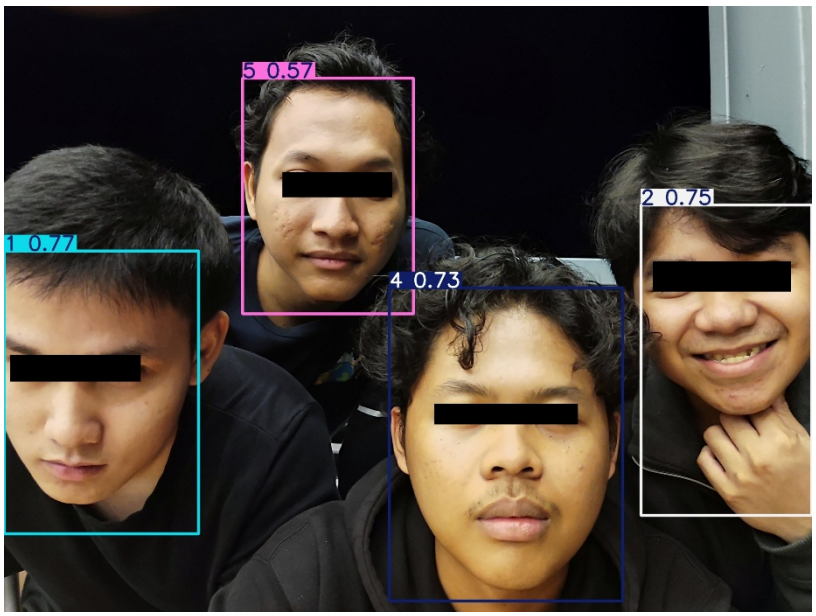
Dari hasil ini dapat disimpulkan bahwa persamaan antar-model terletak pada kemampuan mendeteksi ekspresi yang jelas seperti marah, jijik, dan bahagia dengan skor kepercayaan tinggi. Selain itu, semua model konsisten menunjukkan kelemahan pada ekspresi takut, baik berupa skor kepercayaan rendah maupun salah klasifikasi. Perbedaannya terletak pada tingkat kestabilan: YOLOv12n dan YOLOv11n tidak melakukan kesalahan deteksi, YOLOv10n dan YOLOv9t mengalami satu kesalahan, sedangkan YOLOv8n mengalami dua kesalahan. Kelebihan masing-masing model dapat disimpulkan sebagai berikut: YOLOv12n unggul dalam kestabilan dan generalisasi, YOLOv11n konsisten meskipun skor minimumnya lebih rendah, YOLOv10n sangat presisi pada kelas tertentu, YOLOv9t efisien dan kuat pada emosi positif seperti bahagia, sementara YOLOv8n lebih cepat tetapi paling lemah dalam akurasi.

Kelemahan utama dari seluruh model adalah keterbatasan dalam mengenali ekspresi takut dan kebingungan antar-emosi yang memiliki kemiripan visual.

4.8.2 Hasil Inference pada Image

Pada sub bab ini Model YOLO dilakukan inference pada gambar yang berisi empat orang dengan ekspresi emosional yang diketahui: marah, sedih, netral, dan bahagia, dari kiri ke kanan. Penutupan pada mata dilakukan hanya untuk laporan bukan pada inference agar tidak melanggar privasi pemilik wajah.

YOLOv12n



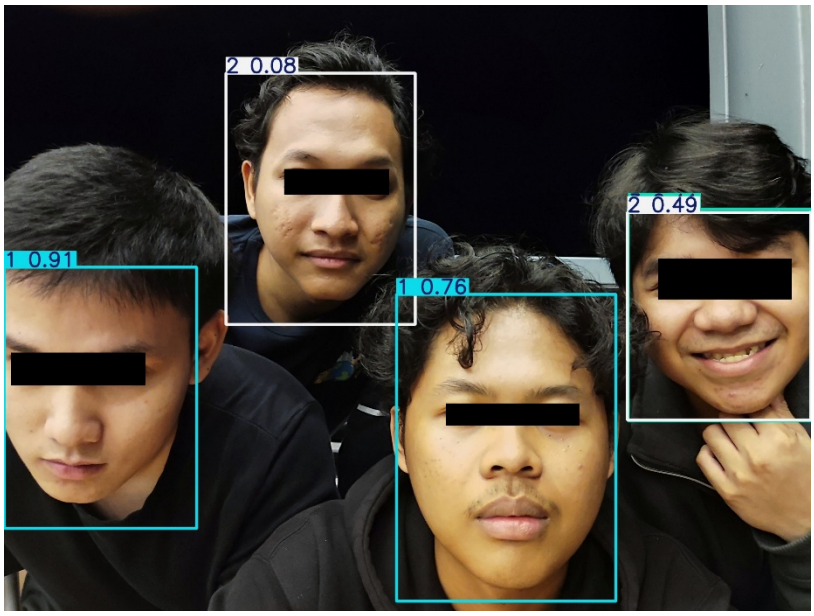
Gambar 4.34 Hasil Inference YOLOv12n pada Image

Berdasarkan Gambar 4.29 hasil pengujian pada gambar kelompok yang terdiri dari empat orang menunjukkan bahwa model ini mampu mendeteksi setiap wajah dengan label emosi yang sesuai dan *confident score* berkisar antara 0,57 hingga 0,77. Model ini berhasil mengidentifikasi emosi yang ditampilkan oleh setiap subjek: *confident score* tertinggi tercatat sebesar 0,77 untuk ekspresi 1 (marah), sementara yang terendah adalah 0,57 untuk ekspresi 5 (sedih).

Meskipun terdapat berbagai sudut, pencahayaan, dan orientasi wajah yang bervariasi dalam foto grup, model ini mempertahankan kemampuannya untuk mendeteksi semua wajah

secara dengan satu ketidakcocokan label 3 (bahagia) menjadi 2 (jijik). Hasil ini menyoroti ketangguhan model dalam menangani skenario multi-wajah dalam kondisi yang kurang terkendali. Namun demikian, tingkat *confident score* yang relatif lebih rendah pada pendeteksian tertentu dan satu kesalahan deteksi, menunjukkan perlu untuk disempurnakan.

YOLOv11n



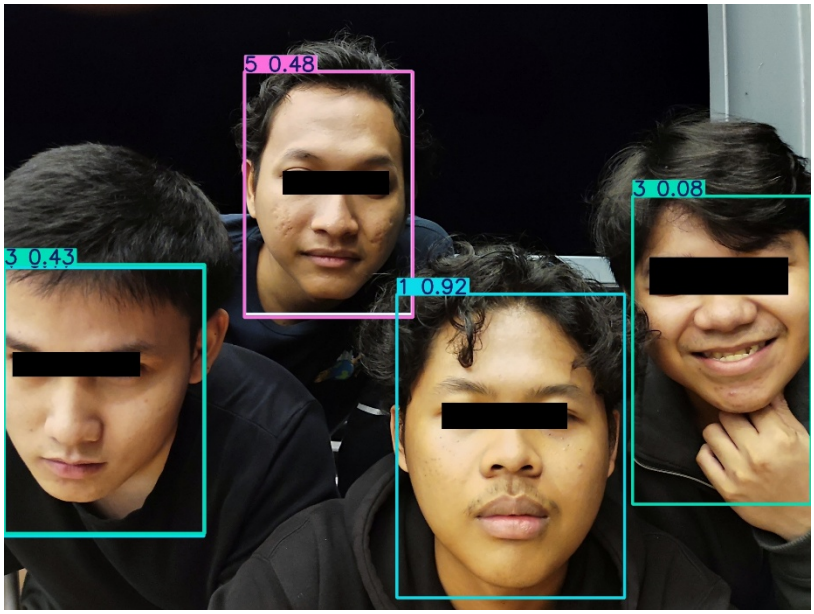
Gambar 4.35 Hasil Inference YOLOv11n pada Image

Berdasarkan Gambar 4.30 pada hasil Inference pada YOLOv11n, model mendeteksi empat wajah dan memberikan label emosi dengan *confident score* mulai dari 0,08 hingga 0,91. *Confident score* tertinggi sebesar 0,91 dicapai untuk ekspresi marah pada individu pertama yang berada di sisi kiri gambar. Pada individu kedua yang berada di tengah atas, terjadi salah klasifikasi dari ekspresi 5 (sedih) menjadi 2 (jijik) dengan *confident score* terendah sebesar 0,08. Selanjutnya, individu ketiga yang berada di sisi kanan atas juga mengalami salah klasifikasi dari ekspresi 4 (netral) menjadi 1 (marah) dengan *confident score* sebesar 0,76. Adapun individu keempat yang berada di sisi kanan bawah diklasifikasikan salah dari ekspresi 3 (bahagia) menjadi jijik dengan *confident score* sebesar 0,49.

Dibandingkan dengan hasil deteksi sebelumnya, versi model ini menunjukkan kesalahan yang cukup banyak pada individu kedua sampai akhir, dengan keyakinan yang sangat rendah pada satu subjek. Model ini tetap konsisten dalam mendeteksi semua wajah yang ada tanpa

melewatkan satu pun, yang mengindikasikan keandalannya dalam skenario multi-wajah, meskipun hasil deteksi masih banyak yang salah dalam menentukan emosi.

YOLOv10n

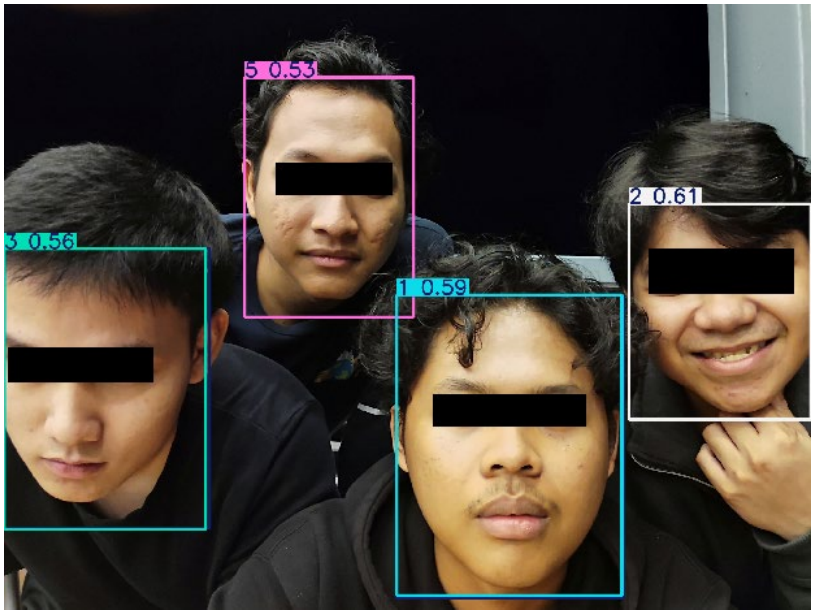


Gambar 4.36 Hasil Inference YOLOv10n pada Image

Berdasarkan Gambar 4.31 pada hasil deteksi menggunakan model YOLOv10n, sistem berhasil mengidentifikasi empat wajah di dalam gambar dan mengklasifikasikan emosinya dengan *confident score* mulai dari 0,08 hingga 0,92. *Confident score* tertinggi tercatat sebesar 0,92 untuk ekspresi 1(marah) pada individu yang terletak di tengah bawah. *Confident score* terendah muncul pada individu di bagian kanan atas dengan nilai 0,08 untuk ekspresi 3 (bahagia).

Dibandingkan dengan hasil YOLOv11n sebelumnya, YOLOv10n menunjukkan kemampuan yang cukup baik dalam mendeteksi seluruh bagian wajah tetapi dengan *confident score* yang sedikit lebih rendah secara keseluruhan. Model ini mempertahankan konsistensi pendeteksian di semua wajah tanpa melewatkan satu pun, menunjukkan kemampuan pendeteksian objek yang dapat diandalkan dalam foto grup. Namun, keyakinan klasifikasi emosi tetap bervariasi. Model ini hanya memiliki dua kesalahan pada emosi 1 (marah) menjadi 3 (bahagia) dan emosi 4 (netral) menjadi 1 (marah), menunjukkan peluang untuk perbaikan dalam membedakan emosi.

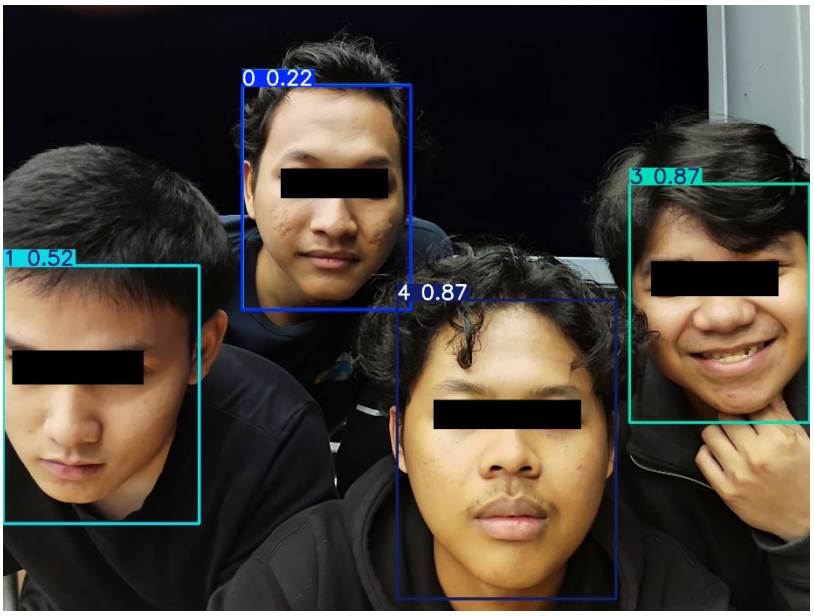
YOLOv9t



Gambar 4.37 Hasil Inference YOLOv9t pada Image

Berdasarkan Gambar 4.32 dalam hasil deteksi menggunakan model YOLOv9t, sistem berhasil mendeteksi empat wajah dalam gambar. *Confident score* berkisar antara 0,53 hingga 0,61, yang relatif seimbang dan konsisten dibandingkan dengan versi sebelumnya. *Confident score* tertinggi pada individu di kanan atas dengan nilai 0,61 untuk ekspresi 3 (bahagia), sedangkan yang terendah adalah 0,53 untuk individu di tengah atas dengan label 5 (sedih). Model ini memiliki dua kesalahan pada emosi 1 (marah) menjadi 3 (bahagia) dan emosi 3 (bahagia) menjadi 2(jijik). Jika dibandingkan dengan model YOLOv10n dan YOLOv11n, YOLOv9t menunjukkan konsistensi *confident score* yang sedikit lebih baik di semua wajah yang terdeteksi, yang mengindikasikan pola prediksi yang lebih stabil. Namun, tetap menunjukkan bahwa model ini masih memiliki ruang untuk memperkuat kepastian dalam klasifikasi emosi.

YOLOv8n



Gambar 4.38 Hasil *Inference* YOLOv8n pada *Image*

Berdasarkan Gambar 4.33 model YOLOv8n dengan tepat mendeteksi ekspresi 1 (marah), 4 (netral), dan 3 (bahagia), dengan *confident score* masing-masing 0,52, 0,87, dan 0,87. Namun, ekspresi "sedih" salah dideteksi sebagai 0 (takut) dengan *confident score* rendah sebesar 0,22. Hasil ini menunjukkan bahwa meskipun YOLOv8n bekerja dengan baik pada emosi yang lebih jelas seperti 3 (bahagia) dan 4 (netral), ia kesulitan dengan emosi negatif yang kurang dapat dibedakan, seperti 0 (takut). Namun, tetap menunjukkan bahwa model ini masih memiliki ruang untuk memperkuat kepastian dalam klasifikasi emosi.

Perbandingan Hasil Inference pada Image

Hasil pengujian pada YOLOv12n menunjukkan bahwa model mampu mendeteksi dan mengklasifikasikan setiap wajah dengan label emosi yang sesuai. *Confidence score* berkisar antara 0,57 hingga 0,77, dengan skor tertinggi sebesar 0,77 pada ekspresi marah dan terendah sebesar 0,57 pada ekspresi sedih. Meskipun terdapat variasi sudut dan pencahayaan dalam foto grup, YOLOv12n tetap dapat mendeteksi semua wajah dengan hanya satu kesalahan, yaitu ekspresi bahagia yang terdeteksi sebagai jijik. Hal ini menegaskan ketangguhan model dalam

menghadapi skenario multi-wajah meskipun masih terdapat kelemahan dalam menangani ekspresi dengan perbedaan tipis.

Pada YOLOv11n, model mendeteksi keempat wajah dengan *confidence score* yang bervariasi antara 0,08 hingga 0,91. Skor tertinggi tercatat pada ekspresi marah dengan nilai 0,91, sedangkan skor terendah muncul pada ekspresi 2 (jijik) dengan hanya 0,08. Dibandingkan dengan YOLOv12n, hasil deteksi YOLOv11n menampilkan lebih banyak kesalahan terutama pada individu kedua hingga akhir, serta menunjukkan tingkat keyakinan yang sangat rendah pada satu subjek. Walaupun demikian, model ini tetap konsisten dalam mendeteksi seluruh wajah yang ada, yang berarti tetap andal pada aspek pendeteksian objek meski kurang presisi dalam klasifikasi emosi.

YOLOv10n menunjukkan performa cukup baik dengan *confidence score* antara 0,08 hingga 0,92. Deteksi terbaik muncul pada ekspresi 1 (marah) dengan nilai 0,92, sementara terendah tercatat pada ekspresi bahagia dengan nilai 0,08. Model ini berhasil mendeteksi semua wajah tanpa ada yang terlewat, tetapi menghasilkan dua kesalahan klasifikasi: ekspresi 1 (marah) terdeteksi sebagai ekspresi 3 (bahagia) dan ekspresi 4 (netral) terdeteksi sebagai ekspresi 1 (marah). Kesalahan ini menandakan masih adanya tantangan dalam membedakan ekspresi yang memiliki kesamaan fitur wajah.

Berbeda dengan itu, YOLOv9t menampilkan confidence score yang lebih seimbang, berkisar antara 0,53 hingga 0,61. Skor tertinggi diperoleh pada ekspresi 3 (bahagia) dengan nilai 0,61, sedangkan skor terendah pada ekspresi 5 (sedih) senilai 0,53. Meskipun tingkat keyakinan cukup rendah dibandingkan model lain, hasil ini menunjukkan konsistensi pada seluruh wajah yang terdeteksi. Namun, dua kesalahan klasifikasi masih ditemukan, yaitu ekspresi 1 (marah) yang terdeteksi sebagai 3 (bahagia) dan ekspresi 3 (bahagia) yang terdeteksi sebagai ekspresi 2 (jijik). Hal ini menegaskan bahwa meskipun YOLOv9t lebih stabil, ia masih memerlukan perbaikan dalam akurasi klasifikasi antar ekspresi yang mirip.

Sementara itu, YOLOv8n berhasil mendeteksi ekspresi marah, netral, dan bahagia dengan *confidence score* masing-masing 0,52, 0,87, dan 0,87. Namun, ekspresi sedih salah diklasifikasikan sebagai takut dengan *confidence score* rendah sebesar 0,22. Kesalahan ini menunjukkan bahwa YOLOv8n memiliki keterbatasan dalam membedakan ekspresi negatif yang serupa. Meski demikian, model tetap mampu memberikan deteksi yang cukup akurat pada ekspresi yang lebih jelas seperti ekspresi 3 (bahagia) dan 4 (netral).

Secara keseluruhan, hasil *inference* pada image multi-wajah menunjukkan bahwa seluruh model mampu mendeteksi keempat wajah, tetapi akurasi klasifikasi emosi bervariasi. YOLOv12n menampilkan performa terbaik dengan hanya satu kesalahan klasifikasi dan skor kepercayaan relatif stabil. YOLOv11n dan YOLOv10n masih andal dalam pendeteksian wajah namun rentan melakukan kesalahan klasifikasi, terutama pada ekspresi yang mirip. YOLOv9t lebih konsisten dalam skor kepercayaan tetapi akurasinya belum optimal. YOLOv8n menunjukkan performa terlemah, terutama dalam mengenali emosi negatif dengan tingkat keyakinan yang rendah.

4.8.3 Perbandingan Hasil Inference Pada Satu Wajah dan Banyak Wajah

Berdasarkan hasil inference pada *video* RAVDESS (satu wajah) dan *image* multi-wajah, terlihat perbedaan yang signifikan dalam kestabilan, akurasi, dan konsistensi antar model YOLO yang diuji.

Pada skenario satu wajah sebagian besar model menunjukkan performa yang relatif stabil dengan tingkat *confidence score* tinggi. Model YOLOv12n dan YOLOv11n mampu mengklasifikasikan seluruh ekspresi dengan benar tanpa kesalahan label. YOLOv10n dan YOLOv9t menampilkan hasil yang cukup baik namun dengan satu kesalahan yaitu dari ekspresi 0 (takut) ke ekspresi 5 (sedih) di pertengahan sampai akhir *frame* dan ekspresi 0 (takut) ke ekspresi 1 (angry) di awal sampai pertengahan *frame*. Sementara YOLOv8n menunjukkan performa paling rendah dengan dua kesalahan klasifikasi dari ekspresi 0 (takut) ke ekspresi 1 (marah) di akhir *frame* pada dan ekspresi 5 (sedih) ke ekspresi 0 (takut) di awal sampai pertengahan menuju akhir *frame*. Persamaan di antara model-model ini adalah bahwa ekspresi seperti marah, jijik, dan bahagia dapat terdeteksi dengan tingkat kepercayaan tinggi. Namun, kelemahan utamanya terletak pada kesulitan dalam mengenali ekspresi “takut” yang sering menghasilkan *confidence score* lebih rendah atau salah klasifikasi seperti pada Tabel 4.4.

Tabel 4.4 Hasil Klasifikasi Ravdess

Model		Emotion						
		0 (Affraid)	1 (Angry)	2 (Disgusted)	3 (happy)	4 (neutral)	5 (sad)	6 (surprised)
YOLOv12n	Confident score	0.65	0.88	0.93	0.94	0.93	0.93	0.94
	Hasil Klasifikasi	0 (Affraid)	1 (Angry)	2 (Disgusted)	3 (happy)	4 (neutral)	5 (sad)	6 (surprised)
YOLOv11n	Confident score	0.56	0.94	0.94	0.93	0.93	0.8	0.92
	Hasil Klasifikasi	0 (Affraid)	1 (Angry)	2 (Disgusted)	3 (happy)	4 (neutral)	5 (sad)	6 (surprised)
YOLOv10n	Confident score	0.71	0.97	0.94	0.91	0.85	0.92	0.7
	Hasil Klasifikasi	5 (sad)	1 (Angry)	2 (Disgusted)	3 (happy)	4 (neutral)	5 (sad)	6 (surprised)
YOLOv9t	Confident score	0.78	0.92	0.93	0.96	0.69	0.92	0.7
	Hasil Klasifikasi	1 (Angry)	1 (Angry)	2 (Disgusted)	3 (happy)	4 (neutral)	5 (sad)	6 (surprised)
YOLOv8n	Confident score	0.85	0.94	0.94	0.91	0.94	0.82	0.93
	Hasil Klasifikasi	1 (Angry)	1 (Angry)	2 (Disgusted)	3 (happy)	4 (neutral)	0 (Affraid)	6 (surprised)

Tabel 4.5 Hasil Klasifikasi pada Multi-Wajah

Model		Emotion			
		1 (Angry)	5 (sad)	4 (neutral)	3 (happy)
YOLOv12n	Confident score	0.77	0.57	0.73	0.75
	Hasil Klasifikasi	1 (Angry)	5 (sad)	4 (neutral)	2 (disgusted)
YOLOv11n	Confident score	0.91	0.08	0.76	0.49
	Hasil Klasifikasi	1 (Angry)	2 (disgusted)	1 (Angry)	2 (disgusted)
YOLOv10n	Confident score	0.43	0.49	0.92	0.08
	Hasil Klasifikasi	3 (happy)	5 (sad)	1 (Angry)	3 (happy)
YOLOv9t	Confident score	0.56	0.53	0.59	0.61
	Hasil Klasifikasi	3 (happy)	5 (sad)	1 (Angry)	2 (disgusted)
YOLOv8n	Confident score	0.52	0.22	0.87	0.87
	Hasil Klasifikasi	1 (Angry)	0 (affraid)	4 (neutral)	3 (happy)

Sebaliknya, pada skenario banyak wajah, performa model cenderung menurun karena kompleksitas meningkat, seperti variasi sudut, pencahayaan, dan banyak individu dalam satu gambar dapat dilihat pada Tabel 4.5. YOLOv12n tetap menunjukkan kinerja terbaik, berhasil mendeteksi semua wajah dengan relatif baik meskipun terdapat satu kesalahan pada ekspresi bahagia yang salah dikenali sebagai jijik. YOLOv11n dan YOLOv9t menunjukkan kesalahan yang lebih sering contohnya YOLOv11n tidak memiliki prediksi yang benar dan YOLOv9t hanya dapat menprediksi 1 wajah yang benar, terutama pada individu dengan ekspresi yang lemah s atau wajah yang posisinya sedikit menunduk s. YOLOv10n menampilkan *confidence score* yang lebih merata dengan *confident score* tidak ada yang dibawah 0,5 g di semua wajah, tetapi tetap menghasilkan tiga kesalahan deteksi yaitu ekspresi 1 (marah) ke ekspresi 5 (sedih) , yaitu ekspresi 4 (netral) ke ekspresi 1 (marah) dan ekspresi 3 (bahagia) ke ekspresi 2 (jijik) . YOLOv8n memperlihatkan kinerja terburuk, dengan kesalahan pada emosi sedih yang salah dideteksi sebagai takut dengan *confidence score* sangat rendah sebesar 0,22. Hal ini menandakan bahwa model versi awal lebih kesulitan menghadapi kondisi multi-wajah dibanding versi yang lebih baru.

Dari perbandingan keduanya, dapat ditarik beberapa kesimpulan. Persamaan antara hasil *inference* satu wajah dan banyak wajah adalah kemampuan model mendeteksi emosi yang jelas

seperti marah dan bahagia dengan tingkat keyakinan yang tinggi. Perbedaannya terletak pada performa dalam mendekteksi dan mengklasifikasi emosi dengan benar model lebih baik pada kasus satu wajah dimana kesalahan klasifikasi sebanyak 4 dari 26, sementara pada banyak wajah performa menurun dimana kesalahan klasifikasi pada multi-wajah sebanyak 9 dari 20 karena faktor kompleksitas visual. Kelebihan YOLOv12n adalah konsistensi dan akurasi tinggi, baik di satu wajah maupun banyak wajah. YOLOv11n hanya unggul pada satu wajah dengan tanpa kesalahan tapi salah semua dalam klasifikasi di multi-wajah. YOLOv10n unggul pada presisi ekspresi marah dan jijik, tetapi lebih rentan salah klasifikasi di multi-wajah. YOLOv9t memiliki kelebihan dalam menjaga *confidence score* seimbang di banyak individu, meskipun masih menghasilkan kesalahan label. YOLOv8n paling lemah, meski cepat, ia kesulitan membedakan emosi negatif dengan jelas.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang membandingkan kinerja algoritma YOLOv8 hingga YOLOv12 untuk deteksi emosi pada ekspresi wajah, diperoleh bahwa YOLOv12 merupakan model dengan performa terbaik. Dasar penentuan ini adalah capaian metrik evaluasi yang unggul, yaitu *recall* sebesar 0,986 dan mAP50 tertinggi sebesar 0,993, disertai *loss* pelatihan dan validasi yang seimbang sehingga menunjukkan kemampuan generalisasi yang baik serta minim risiko *overfitting*. Dalam konteks penggunaannya, YOLOv12n diuji pada dua skenario berbeda. Pertama, pengujian dengan gambar dari video RAVDESS dilakukan untuk mengevaluasi stabilitas model dalam kondisi data terkontrol. Kedua, pengujian dengan gambar yang diambil dari ponsel bertujuan untuk menilai kemampuan model dalam mendeteksi emosi pada banyak wajah sekaligus, sebagaimana kasus nyata ketika berhadapan dengan banyak konsumen dalam satu frame. Hasil pengujian menunjukkan bahwa model ini mampu mempertahankan akurasi deteksi yang tinggi dan *confidence score* yang stabil meskipun terdapat variasi pencahayaan dan jumlah wajah. Temuan menarik di hasil *inference* juga menunjukkan bahwa, meskipun secara umum akurasinya tinggi, terdapat satu kasus kesalahan klasifikasi yang mengindikasikan adanya tantangan pada ekspresi wajah dengan kemiripan fitur tertentu. Secara keseluruhan, YOLOv12n menjadi pilihan terbaik untuk mendeteksi emosi wajah dengan presisi tinggi pada implementasi *real-time* di berbagai lingkungan.

5.2 Saran

1. Perlu dilakukan peningkatan lebih lanjut pada dataset, terutama dengan menambahkan variasi ekspresi wajah yang lebih kompleks dan kondisi pencahayaan yang beragam untuk meningkatkan kemampuan generalisasi model.
2. Diperlukan pengujian lebih luas pada video atau gambar agar performa model dapat diuji dalam kondisi lingkungan yang tidak terkontrol.
3. Perlu dilakukan eksplorasi terhadap teknik augmentasi data dan *fine-tuning parameter* pelatihan untuk mengurangi kesalahan klasifikasi emosi yang mirip, seperti “takut” dan “sedih”.

DAFTAR PUSTAKA

Hidayatullah, P., Syakranib, N., Sholahuddin, M. R., Gelar, T., & Tubagus, R. (2025). YOLOv8 to YOLO11: A Comprehensive Architecture In-depth Comparative Review. *arXiv*. Diambil kembali dari https://arxiv.org/abs/2501.13400

Adinata, A. R., Rohana, T., Baihaqi, K. A., & Faisal, S. (2024). Implementasi Algoritma Convolutional Neural Network dan YOLOV8 Untuk Klasifikasi Ras Kucing. *Building of Informatics, Technology and Science (BITS), 6*(3), 1658-1667. Diambil kembali dari https://doi.org/10.47065/bits.v6i3.5913

Azizi, F. N., Kurniawardhani, A., & Paputungan, I. V. (2022). Facial Expression Image based Emotion Detection using Convolutional Neural Network. *2022 IEEE 20th Student Conference on Research and Development (SCOReD).* Bangi: IEEE. doi:10.1109/SCOReD57082.2022.9974104

Chaitra, T. K., M., C. C., & Dr. M. , Z. K. (t.thn.). Face Detection and Feature Extraction. *International journal of engineering research and technology, 6*(13). Diambil kembali dari https://www.ijert.org/face-detection-and-feature-extraction

Debelee, T. G., Kebede, S. R., Schwenker, F., & Matewos, Z. (2020). Deep Learning in Selected Cancers' Image Analysis—A Survey. *Journal of Imaging*.

Fabri, M. (2014). Mediating the Expression of Emotion in Educational CVEs: An Experimental. Diambil kembali dari https://www.researchgate.net/figure/The-six-universal-emotions-and-neutralexpression

Goeleven, E., Raedt, R. D., Leyman, L., & Verschuere, B. (2008). The Karolinska Directed Emotional Faces: A validation study. *Cognition and Emotion, 22*(6). doi:10.1080/02699930701626582

Görtler, J., Hohman, F., Moritz, D., Wongsuphasawat, K., Ren, D., Kirchner, M., . . . Patel, K. D. (2022). Neo: Generalizing Confusion Matrix Visualization to Hierarchical and Multi-Output Labels. *CHI '22: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, (hal. 1 - 13). doi:10.1145/3491102.3501823

Hans, A. S., & Rao , S. (2021). A CNN-LSTM Based Deep Neural Networks For Facial Emotion Detection Videos. *INTERNATIONAL JOURNAL OF ADVANCES IN SIGNAL AND IMAGE SCIENCES, 7*(1). doi:10.29284/ijasis.7.1.2021.11-20

Hasan, M. A., Lazem, A. H., Alkhafaji, M. A., & Taher, H. B. (2023). Facial Human Emotion Recognition by Using YOLO Faces. *JOINCS (Journal of Informatics, Network, and Computer Science), 6*(2). doi:10.21070/joincs.v6i2.1629

Jegham, N., Abdelatti, M. F., & Koh, C. Y. (2025). YOLO Evolution: A Comprehensive Benchmark and Architectural Review of YOLOv12, YOLO11, and Their Previous Versions. doi:10.13140/RG.2.2.15952.83201

Jeri, & Hidayat, Z. S. (2024). Penentuan Ukuran Batch Optimal Dalam Pelatihan Model YOLOv8 untuk Deteksi Objek Pada Kendaraan Otonom. *Jurnal Ilmiah NERO, 9*(1).

Jocher, G., & Qiu, J. (2024). *Ultralytics YOLO11*, 11.0.0. Diambil kembali dari https://github.com/ultralytics/ultralytics

Jocher, G., Chaurasia, A., & Qiu, J. (2023). *Ultralytics YOLOv8*, 8.0.0. Diambil kembali dari https://github.com/ultralytics/ultralytics

KAMBLE, K., & Sengupta, J. (2023). A comprehensive survey on emotion recognition based on electroencephalograph (EEG) signals. *Multimedia Tools and Applications*.

Keightley, M. L., Chiew, K. S., Winocur, G., & Grady, C. L. (2007). Age-related differences in brain activity underlying identification of emotional expressions in faces. *Social cognitive and affective neuroscience, 2*(4), 292–302. Diambil kembali dari https://doi.org/10.1093/scan/nsm024

Khanam, R., & Hussain, M. (2024). YOLOv11: An Overview Of The Key Architectural Enhancements. *arXiv*. doi:10.48550/arXiv.2410.17725

Liu, C., & Hui, L. (2023). ReLU soothes the NTK condition number and accelerates optimization for wide neural networks. *ArXiv*. doi:10.48550/arXiv.2305.08813

Livingstone, S. R., & Russo, F. A. (t.thn.). The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. *PLoS One, 13*(5). doi:10.1371/journal.pone.0196391

Lundqvist, D., Flykt, A., & Öhman, A. (1998). The Karolinska Directed Emotional Faces - KDEF. *Stockholm: Karolinska Institute, Department of Clinical Neuroscience, Psychology Section.*

O'Shea, K., & Nash, R. (2015). An Introduction to Convolutional Neural Networks. *ArXiv*. doi:10.48550/arXiv.1511.08458

Phung, & Rhee. (2019). A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets. *Applied Sciences*.

Potts, L. (t.thn.). *What loss functions are used in YOLOv8 for training?* Diambil kembali dari yolov8architecture.com: https://yolov8architecture.com/loss-functions-in-yolov8/

Pratama, N. K., & Anggraeny, F. T. (2023). Deteksi Lampu Lalu Lintas dan Zebra Cross Menggunakan MobilenetV2 Single Shot Detector. *Jurnal Informatika Kaputama (JIK), 7*(2), 225-232. doi:10.59697/jik.v7i2.140

R., R., & P., S. (2023). A Review of Convolutional Neural Networks, its Variants and Applications. *2023 International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS).*

Rasheed , A. F., & Zarkoosh, M. (2024). YOLOv11 Optimization for Efficient Resource Utilization. *arXiv*. doi:10.48550/arXiv.2412.14790

Redmon, J., Divvala, S. K., Girshick, R. B., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788. Diambil kembali dari https://api.semanticscholar.org/CorpusID:206594738

Roboflow Inc. (t.thn.). Diambil kembali dari Roboflow: https://roboflow.com/

Salma, K., & Hidayat, S. (2024). Deteksi Antusiasme Siswa dengan Algoritma Yolov8 pada Proses Pembelajaran Daring. *Jurnal Indonesia: Manajemen Informatika dan Komunikasi (JIMIK), 5*(2). doi:10.35870/jimik.v5i2.716

Sharma , A., Kumar , V., & Longchamps, L. (2024). Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species. *Smart Agricultural Technology, 9*. doi:10.1016/j.atech.2024.100648

Sharma, C., Singh, H., Rana, H., & Kumar, K. (2023). Emotion Detection using Machine Learning. *International Journal for Research in Applied Science & Engineering*.

Surwade, N., Gundlapalli, C., Hussain, A., & Das, A. (2023). Machine Learning SIGNATURE VERIFICATION to eradicate Banking Fraud, accelerate Anti-Money Laundering Compliance. doi:10.13140/RG.2.2.12919.80808

Tian, Y., Ye, Q., & Doermann, D. (2025). YOLOv12: Attention-Centric Real-Time Object Detectors. *ArXiv*. Diambil kembali dari https://arxiv.org/abs/2502.12524

Ultralytics. (t.thn.). *train-settings*. Dipetik Agustus 18 , 2025, dari Ultralytics: https://docs.ultralytics.com/usage/cfg/#train-settings

Wang, A., Chen, H., Liu, L., Chen, K., Li, Z., Han, J., & Ding, G. (2024). YOLOv10: Real-Time End-to-End Object Detection. *ArXiv*. Diambil kembali dari https://arxiv.org/abs/2405.14458

Wang, B. (2022). A Parallel Implementation of Computing Mean Average Precision. *arXiv*. doi:10.48550/arXiv.2206.09504

Wang, C.-Y., Yeh, I.-H., & Liao, H.-Y. M. (2024). YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *ArXiv*. Diambil kembali dari https://arxiv.org/abs/2402.13616

yolov8.org. (t.thn.). *What is DFL loss in yolov8?* Diambil kembali dari yolov8.org: https://yolov8.org/what-is-box-loss-in-yolov8/