

# DAFTAR ISI

<b>BAB 1 PENDAHULUAN .....</b>	<b>2</b>
1.1 Latar belakang.....	2
1.2 Rumusan Masalah.....	3
1.3 Tujuan .....	3
1.4 Manfaat .....	3
1.5 Batasan Masalah .....	3
<b>BAB II TINJAUAN PUSAKA .....</b>	<b>4</b>
2.1. Penelitian Sebelumnya.....	4
2.2. You Only Look Once (YOLO).....	5
2.3. YOLOv2.....	12
2.4. YOLOv3.....	12
2.4. YOLOv4.....	13
2.5. YOLOv5.....	13
2.6. YOLOv6.....	14
2.7. YOLOv7.....	15
2.8. YOLOv8.....	16
<b>BAB III METODOLOGI .....</b>	<b>17</b>
3.1 Problem Scopping .....	17
3.2 Data Acquisition.....	19
3.3 Data Exploration .....	20
3.4 Modelling .....	21
<b>DAFTAR PUSTAKA.....</b>	<b>23</b>

# **BAB 1 PENDAHULUAN**

## **1.1 Latar belakang**

Perkembangan teknologi telah mengarah pada digitalisasi pekerjaan dan mengubah cara orang memproduksi dan berinteraksi. Perkembangan teknologi global pertama kali diamati sejak Revolusi Industri Pertama di Inggris pada awal abad ke-18 hingga awal abad ke-19. Revolusi pertama ini ditandai dengan ditemukannya mesin uap dan penggunaan energi batu bara. Penemuan teknologi pertama ini berhasil mengubah struktur perekonomian masyarakat dari basis pertanian menjadi basis industri. Perkembangan selanjutnya terjadi pada akhir abad ke-19, dengan ditemukannya sumber energi baru: listrik, gas, dan minyak, menandai dimulainya Revolusi Industri Kedua yang diprakarsai oleh Amerika Serikat (Asiati et al., n.d.).

Banyak negara yang beriklim tropis dan tanah yang subur menjadikan suatu negara yang ideal untuk menanam berbagai jenis tanaman, termasuk apel yang tumbuh di daerah beriklim sejuk seperti dataran tinggi dan pegunungan. Pengelolaan hasil panen apel membutuhkan kualitas buah yang baik agar produk olahan juga memiliki kualitas terjamin. Metode konvensional dalam kualitas apel masih menggunakan indera penglihatan manusia berdasarkan bentuk fisik buah. Namun, metode ini memiliki beberapa kekurangan, seperti penghitungan yang tidak akurat dan biaya tenaga kerja yang tinggi.

Algoritma pembelajaran mendalam telah memperoleh hasil deteksi yang memuaskan berdasarkan kemampuan ekstraksi fitur yang kuat dari Convolutional Neural Network (CNN) dan telah muncul sebagai rute pengembangan signifikan untuk deteksi apel (Kapach et al., 2012). Salah satu algoritma dalam computer vision yang masih berkembang dan banyak digunakan saat ini adalah metode YOLO (You Only Look Once). Pengembangan metode yolo dengan algoritma CNN yaitu YOLOv8 memiliki kinerja yang lebih baik dalam hal akurasi dan kecepatan dalam proses deteksi objek dibandingkan dengan versi YOLO yang sebelumnya [6].

Oleh karena itu, penelitian ini mengusulkan sistem deteksi kualitas dan perhitungan buah apel menggunakan metode YOLOv8. Pada penelitian ini dilakukan deteksi dan perhitungan buah apel kualitas baik dan busuk dengan sampel yang digunakan sebanyak 3000 citra yang akan dibagi dalam dua kelas yaitu fresh(segar) dan Stale(busuk). Dengan melakukan penelitian system deteksi kualitas kesegaran apel melalui pelatihan model YOLOv8 diharapkan Meningkatkan akurasi deteksi kualitas apel dan konsistensi hasil dibandingkan dengan metode manual dan juga dapat Mengurangi waktu dan tenaga yang dibutuhkan untuk memeriksa dan menghitung apel secara manual.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka dirumuskan beberapa masalah sebagai berikut:

1. Bagaimana implementasi YOLOv8 untuk deteksi fokus berdasarkan ekspresi wajah?
2. Bagaimana hasil keakuratan deteksi fokus berdasarkan ekspresi wajah menggunakan YOLOv8?

## 1.3 Tujuan

Tujuan dari penelitian ini adalah untuk mengukur keakuratan pendeteksian tingkat konsentrasi atau fokus karyawan dari ekspresi wajah menggunakan algoritma YOLOv8.

## 1.4 Manfaat

Penelitian yang dilakukan memiliki keuntungan sebagai berikut:

1. Algoritma YOLOv8 dapat digunakan untuk mengetahui tingkat akurasi deteksi fokus karyawan berdasarkan ekspresi wajah.
2. Algoritma YOLOv8 dapat digunakan untuk mendeteksi konsentrasi karyawan berdasarkan ekspresi wajah.
3. Dapat mendeteksi fokus pada pegawai berdasarkan ekspresi wajah menggunakan algoritma YOLOv8.
4. Dapat dijadikan sebagai standar untuk penelitian lebih lanjut mengenai deteksi objek.

## 1.5 Batasan Masalah

Pada penelitian ini, batasan masalah yang dibahas adalah sebagai berikut:

1. Dataset yang digunakan merupakan kombinasi format gambar (.jpg) dari open source Kaggle dan foto pribadi.
2. Dataset gambar yang digunakan dianotasi oleh peneliti sesuai format YOLO.
3. Eksperimen penelitian dapat dilakukan secara real-time melalui video streaming.
4. Penelitian ini dilakukan untuk mengetahui tingkat akurasi dengan menggunakan arsitektur YOLOv8.

## **BAB II TINJAUAN PUSAKA**

### **2.1. Penelitian Sebelumnya**

Penelitian terdahulu diperoleh dengan mencari dan mengumpulkan jurnal penelitian yang telah dilakukan. Penelitian terdahulu diperlukan sebagai penelitian pembandingan maupun penelitian acuan. Hal ini dalam Upaya agar penelitian menghasilkan hasil yang lebih baik. Berikut penelitian terdahulu yang berkaitan dengan topik penelitian ini:

#### **1. Pendekatan Deteksi Objek Menggunakan YOLO**

Salah satu penelitian yang menjadi rujukan adalah karya Redmon et al. (2016) yang memperkenalkan YOLO (You Only Look Once) sebagai metode deteksi objek yang efisien. Penelitian ini menekankan pada arsitektur YOLO yang memungkinkan deteksi objek dilakukan dalam satu proses inferensi. Dalam pengembangannya, YOLOv8 menunjukkan peningkatan dalam hal akurasi dan kecepatan, sehingga sangat cocok untuk aplikasi berbasis video.

#### **2. Deteksi Buah Menggunakan YOLO**

Sa et al. (2018) melakukan penelitian tentang deteksi buah dalam lingkungan alami menggunakan YOLO. Fokus penelitian ini adalah bagaimana YOLO mampu mengidentifikasi buah dengan tingkat akurasi tinggi, bahkan dalam kondisi latar belakang yang kompleks seperti dedaunan dan bayangan. Selain itu, Bargoti dan Underwood (2017) mendalami deteksi buah di kebun dengan mengintegrasikan YOLO untuk mendeteksi objek kecil di area terbuka.

#### **3. Deteksi Kesegaran Buah**

Penelitian oleh Mahesh et al. (2020) mengembangkan sistem berbasis CNN untuk mendeteksi tingkat kematangan dan kesegaran buah, termasuk apel. Studi ini menggunakan fitur visual seperti warna, tekstur, dan pola untuk mengevaluasi kesegaran secara otomatis.

#### **4. Aplikasi Video dalam Deteksi Buah**

Studi oleh Li et al. (2021) menunjukkan bagaimana data video dapat digunakan untuk mendeteksi buah secara real-time. Penelitian ini memanfaatkan urutan waktu dalam

video untuk meningkatkan akurasi deteksi pada objek yang bergerak atau kondisi pencahayaan yang berubah.

## 5. Keunggulan YOLOv8

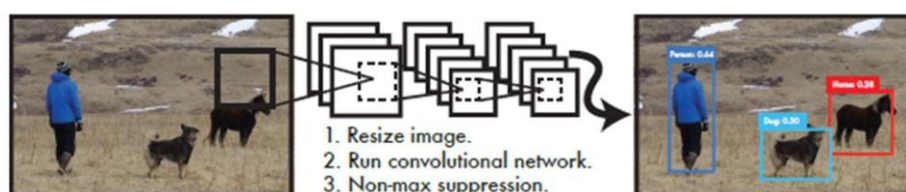
Jocher et al. (2023) memperkenalkan YOLOv8 dengan kemampuan deteksi yang lebih baik dibandingkan versi sebelumnya. Dengan model yang lebih ringan, YOLOv8 mampu mendeteksi objek kecil dan menangani deteksi dalam video streaming secara efisien, menjadikannya alat yang ideal untuk analisis kesegaran apel berbasis video.

## 6. Deteksi Kualitas Apel dengan Teknologi Citra

Arivazhagan et al. (2019) menggunakan metode pengolahan citra untuk mendeteksi kualitas apel berdasarkan warna dan tekstur. Penelitian ini menunjukkan bahwa teknologi berbasis citra dapat diintegrasikan dengan YOLO untuk meningkatkan keakuratan deteksi kualitas dan kesegaran buah.

## 2.2. You Only Look Once (YOLO)

YOLO adalah jaringan konvolusional terpadu yang secara bersamaan dapat memprediksi beberapa kotak pembatas dan probabilitas kelas untuk frame ini. YOLO sepenuhnya terlatih dalam hal gambar dan mengoptimalkan kinerja pengenalan dengan cepat. Model terintegrasi ini memiliki beberapa keunggulan dibandingkan metode deteksi objek tradisional. Pertama, YOLO sangat cepat. Membingkai deteksi sebagai masalah regresi, Bochkovskiy et al. Jaringan saraf Bochkovskiy et al. Gunakan gambar baru selama pengujian untuk memprediksi deteksi. Jaringan dasar ini berjalan pada 45 frame per detik tanpa melakukan batch pada GPU Titan-X, dan versi yang lebih cepat berjalan pada lebih dari 150 fps. Dengan kata lain, video streaming Bochkovskiy dkk dapat diproses secara real time dengan penundaan kurang dari 25 ms. Selain itu, YOLO mencapai lebih dari dua kali lipat akurasi rata-rata sistem real-time lainnya (Redmon et al., n.d.).



Gambar 2.11 Konsep Deteksi Objek Menggunakan YOLO (Redmon et al., n.d.)

Kedua, YOLO dikenal di seluruh dunia karena membuat prediksi gambar. Tidak seperti metode berbasis jendela geser dan proposal wilayah, YOLO memeriksa keseluruhan gambar selama pelatihan dan pengujian dan secara implisit menyandikan informasi kontekstual tentang kelas dan tampilannya.

Metode deteksi terbaik, Fast R-CNN, menimbulkan kesalahan pada bidang latar belakang pada gambar objek karena tidak dapat mendeteksi konteks yang lebih besar. YOLO menghasilkan kurang dari setengah jumlah kesalahan di latar belakang dibandingkan dengan Fast R-CNN.

Ketiga, YOLO mempelajari representasi objek yang dapat digeneralisasikan. Saat dilatih pada gambar alam dan diuji pada karya seni, YOLO secara signifikan mengungguli metode deteksi terkemuka seperti DPM dan RCNN. YOLO lebih serbaguna dan kecil kemungkinannya untuk crash ketika diterapkan pada domain baru atau input yang tidak terduga.

You Only Look Once (YOLO) merupakan salah satu algoritma deep learning yang memanfaatkan jaringan syaraf pada CNN dalam pendeteksian objek. Algoritma tersebut akan membagi citra dalam bentuk grid berukuran  $s \times s$  yang kemudian pada setiap grid akan memprediksi bounding box serta peta kelas pada masing-masing grid. Apabila pada salah satu grid terdeteksi objek, maka pada grid tersebut akan diprediksi oleh bounding box yang mengelilingi objek tersebut. Nilai confidence akan dihitung pada masing-masing bounding box yang kemudian akan diseleksi berdasarkan nilai yang diperoleh (Adhi Wicaksono et al., 2021). Secara formal Redmon et al., mendefinisikan confidence sebagai:

$$Pr(Object) * IOU_{truth \ pred} . \quad (2.3)$$

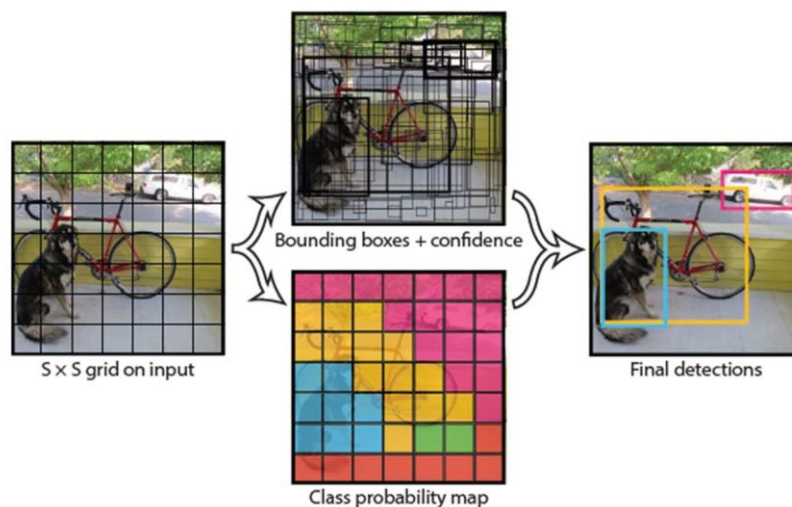
Jika tidak ada objek di sel ini, nilai kepercayaan harus 0. Jika tidak, skor kepercayaan sama dengan perpotongan gabungan (IOU) antara blok prediksi dan kebenaran yang mendasarinya. Setiap kotak pembatas berisi 5 prediksi:  $x, y, w, h$ , dan keyakinan. Koordinat  $(x, y)$  mewakili pusat grid relatif terhadap batas sel grid. Lebar dan tinggi diprediksi berdasarkan keseluruhan gambar. Akhirnya, kepercayaan prediksi adalah IOU antara blok prediksi dan blok kebenaran yang mendasarinya.

Setiap sel grid juga memprediksi probabilitas bersyarat Kelas  $C$ ,  $Pr(Kelas | Objek)$ . Probabilitas ini tergantung pada sel mesh di mana objek berada. Redmon dkk. Memprediksi

hanya satu set probabilitas kelas per sel grid, terlepas dari jumlah kotak B. Pada saat pengujian, Redmon et al. Kalikan probabilitas kelas bersyarat dengan kepercayaan yang diprediksi dari masing-masing kotak,

$$Pr(Class | Object) * Pr(Object) * IOU = Pr(Class) * IOU \quad (2.4)$$

yang memberi Redmon et al. skor confidence khusus kelas untuk setiap kotak. Skor ini mengkodekan probabilitas kelas itu muncul di dalam kotak dan seberapa baik kotak yang diprediksi cocok dengan objek (Redmon et al., n.d.).



Gambar 2.12 Model Deteksi Menggunakan YOLO (Redmon et al., n.d.)

Proses bounding box dilakukan dengan cara menginputkan gambar yang selanjutnya akan ditentukan bounding box dan penentuan kelas yang sesuai. Hasil deteksinya akan menampilkan bounding box yang telah terdapat nilai akurasi dan kelas objek.

Berikut adalah macam-macam parameter yang dapat diatur atau dikonfigurasi pada YOLO:

#### 1. Ukuran batch

Ukuran batch adalah variabel yang menentukan berapa banyak gambar atau data pelatihan yang diinput selama pelatihan. Semakin kecil ukuran paket, semakin cepat proses pelatihan. Di sisi lain, semakin besar nilai ukuran paket, semakin banyak kapasitas penyimpanan yang diperlukan, sehingga memperpanjang proses pelatihan. Ini juga mempengaruhi keakuratan sistem. Nilai yang lebih besar dari ukuran paket yang digunakan menghasilkan akurasi yang lebih tinggi karena sistem mempelajari lebih banyak fitur.

## 2. Segmentasi

Departemen membagi biaya penyebaran menjadi unit yang lebih kecil yang disebut mini-batch. Membaginya menjadi 8 subbagian menggunakan nilai batch 64 memberikan nilai 8. Ini berarti proses pelatihan selesai untuk 8 gambar per minibat. Proses ini diulang delapan kali sampai proses pembelajaran untuk satu bagian selesai. Sistem kemudian memproses batch berikutnya dengan biaya 64. Proses segmentasi bertujuan untuk mempercepat proses pembelajaran sekaligus meningkatkan akurasi GPU (Kusuma et al., 2021).

## 3. Saluran atau Channels

Nilai saluran menentukan kedalaman gambar berdasarkan data yang digunakan selama proses pelatihan. Jika data yang dilatih menggunakan gambar RGB, variabel saluran harus 3. Di sisi lain, saat menggunakan citra grayscale, kanal menggunakan nilai 1 (Kusuma et al., 2021).

## 4. Tingkat pembelajaran atau Learning Rate

Laju pembelajaran adalah penentu bobot yang diperbarui selama kesalahan propagasi balik. Tingkat pembelajaran juga menentukan tingkat pengulangan untuk mencapai fungsi kerugian minimum. Semakin cepat proses belajar maka semakin tinggi pula kecepatan belajarnya. Namun jika learning rate terlalu tinggi dapat terjadi fluktuasi nilai loss function yang tidak beraturan, sehingga diperlukan beberapa kali percobaan untuk mendapatkan nilai learning rate yang optimal (Kusuma et al., 2021).

## 5. Jumlah maksimum batch

Jumlah maksimum batch adalah jumlah iterasi dari proses data pelatihan. Semakin tinggi nilai paket maksimum, semakin banyak sistem akan mempelajari data pelatihan. Jumlah data pelatihan tidak boleh melebihi jumlah maksimum paket. Nilai paket maksimal harus disesuaikan dengan jumlah kelas objek yang akan dideteksi (Kusuma et al., 2021).

$$\text{Max Batches} = \text{number of class} \times 2000 \quad (2.5)$$



Pada proses evaluasi data, perlu dipahami beberapa pengertian yang didapatkan pada saat proses evaluasi data, seperti mAP, loss, accuracy, precision, recall, F1 Score, IOU, bounding box, non-max suppression.

### 1. Mean Average Precision (mAP)

Nilai mAP adalah metrik yang mengukur kinerja model deteksi objek. Sederhananya, mAP adalah rata-rata presisi rata-rata (AP) dan mengukur seberapa baik kinerja file bobot data pelatihan. Sebelum menghitung MAP, kita perlu menetapkan ambang batas untuk IoU untuk memvalidasi objek yang terdeteksi (Kusuma et al., 2021).

### 2. Loss

Pada YOLOv4, setiap 3 skala berbeda akan menghitung loss masing-masing, yang mana hasil dari ketiga loss tersebut akan dirata-rata untuk tahap latih berikutnya. Fungsi loss yang digunakan adalah Mean Square Error (MSE) dan Binary Cross Entropy (BCE).

Mean Square Error (MSE) adalah fungsi loss regresi yang paling umum digunakan. MSE adalah jumlah jarak kuadrat antara variabel target dengan prediksi. Hasilnya akan selalu positif, terlepas dari tanda nilai prediksi, nilai sebenarnya, dan nilai sempurna adalah 0,0. Penguadratan memiliki arti bahwa kesalahan yang lebih besar, akan menghasilkan lebih banyak kesalahan dibandingkan dengan kesalahan yang kecil, yang berarti bahwa model dihukum karena membuat kesalahan yang lebih besar (Giancini et al., n.d.).

Fungsi loss Binary Cross Entropy berguna untuk binary classification. Karena hanya dibutuhkan satu node untuk melakukan klasifikasi 2 kelas, yaitu “focus” dan “not\_focus”. Nilai output dari BCE akan diteruskan ke fungsi aktivasi sigmoid sehingga range dari output adalah (0-1) (Giancini et al., n.d.).

### 3. Accuracy

Nilai mAP adalah metrik yang mengukur kinerja model deteksi objek. Sederhananya, mAP adalah rata-rata presisi rata-rata (AP), ukuran seberapa baik kinerja file dengan data pelatihan berbobot. Sebelum menghitung mAP, kita perlu menetapkan ambang batas pada IoU untuk menentukan validasi. Daccuracy adalah parameter uji yang menentukan keandalan suatu sistem dalam mengklasifikasikan

objek. Parameter akurasi digunakan untuk menentukan keakuratan sistem deteksi manusia berdasarkan data uji. Akurasi dapat ditentukan dengan menggunakan persamaan objek yang terdeteksi (Kusuma et al., 2021):

$$A = \frac{TP}{TP + FP} \quad (2.6)$$

$$A = \frac{TP}{TP + FP} \times 100\% \quad (2.7)$$

#### 4. Precision

Akurasi adalah rasio jumlah objek yang terdeteksi dengan benar atau benar-benar positif untuk semua data positif yang diprediksi. Persamaannya adalah:

$$Presisi = \frac{TP}{TP + FP} \times 100\% \quad (2.8)$$

Dari persamaan (3.4), dapat diketahui bahwa semakin besar nilai False Positive, maka presisi semakin rendah dan begitu pun kebalikannya (Kusuma et al., 2021).

#### 5. Recall

Recall adalah rasio objek yang terdeteksi dengan benar atau positif sejati dengan semua positif. Nilai recall yang tinggi menunjukkan bahwa sistem dapat mengklasifikasikan kelas objek dengan benar. Recall dapat dihitung menggunakan persamaan berikut: (Kusuma et al., 2021):

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (2.9)$$

#### 6. F1 Score

Skor F1 membandingkan rata-rata akurasi dan daya ingat. Nilai Skor F1 tertinggi adalah 1 dan nilai terendah adalah 0. Nilai Skor F1 yang mendekati 1 menunjukkan kinerja sistem yang baik. Secara matematis, F1 dapat dilihat dari persamaan:  $\times$

$$F1 \text{ score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (2.10)$$

#### 7. Intersection Over Union (IOU)

IoU adalah metrik yang mengukur akurasi sistem yang mendeteksi objek pada kumpulan data yang terlatih. IoU membandingkan nyata atau objek dalam gambar dengan kotak pembatas yang diprediksi dari model. Nilai IoU untuk deteksi objek berfungsi sebagai ambang batas. Ambang batas IoU yang umum digunakan adalah 0,5 dan 0,75. Penelitian ini menggunakan nilai cutoff sebesar 0,5. Jika ambang IoU

adalah 0,5, entitas didefinisikan sebagai true positive (TP). Ambang batas IoU < Pada 0,5, objek didefinisikan sebagai positif palsu (FP). (Kusuma et al., 2021).

IOU dapat didefinisikan pada persamaan berikut:

$$IOU = \frac{A \cap B}{A \cup B} \quad (2.11)$$

## 8. Bounding Box

Saat melakukan input citra, sebuah target pada network tersebut akan pertama dipilih, kemudian titik tengah gambar tersebut akan ditentukan menggunakan anchor box tepat pada ground truth. Jadi, ketika memprediksi bounding box, box tersebut akan disesuaikan sesuai pada anchor untuk menyesuaikan target objek, oleh karena itu koordinat offset perlu dilakukan. Pada setiap bounding box, YOLOv4 mengestimasi 4 koordinat bounding box dengan symbol (tx,ty,th,tw) (Giancini et al., n.d.).

Rumus untuk bounding box adalah:

$$bx = \sigma(tx) + cx \quad (2.12) \quad by = \sigma(ty) + cy \quad (2.13)$$

$$bw = pw \cdot c^{tw} \quad (2.14)$$

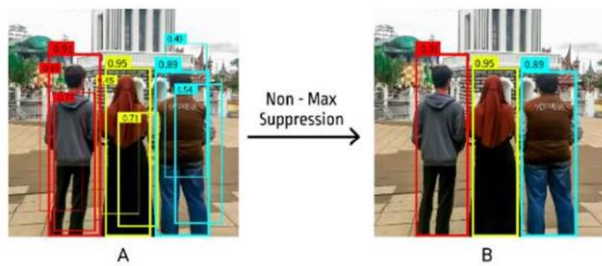
$$bh = pw \cdot c^{th} \quad (2.15)$$

Dimana (tx,ty) merupakan titik tengah dari bounding box, kemudian (tw,th) merupakan ukuran bounding box, (pw,ph) merupakan ukuran dari segmentasi anchor box, dan (cx,cy) merupakan koordinat offset. Karena normalisasi, maka nilai dari koordinatnya berkisar diantara 0-1. YOLOv4

mengestimasi nilai objek pada setiap bounding box sejumlah 3 anchor berdasarkan  $[B \cdot (5 + C)]$  setiap skala (Giancini et al., n.d.).

## 9. Non-Max Suppression (nms)

Non-Max Supression (nms) berfungsi untuk menghilangkan bounding box berlebih dengan konsep seleksi seperti IOU. Proses nms akan mengambil bounding box dengan nilai probabilitas tertinggi. Kemudian menghilangkan bounding box lainnya yang memiliki nilai probabilitas terendah. Misalkan, terdapat 4 bounding box dalam 1 objek, maka 2 bounding box lainnya akan dihilangkan (Giancini et al., n.d.).



Gambar 2.13 Konsep Non-Supression Max

Hasil nms pada Gambar 2.13, merubah hasil bounding box yang mulanya mengalami tumpukkan, kemudian menjaddi 1 bounding box saja. Sehingga hasil yang ditampilkan lebih baik dan mudah dilihat.

### 2.3. YOLOv2

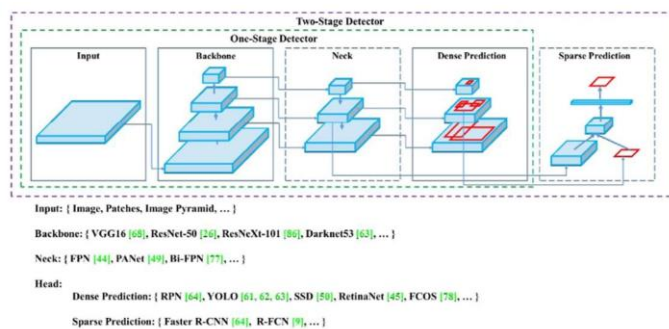
YOLOv2 merupakan pengembangan lanjutan dari versi sebelumnya, dengan beberapa perubahan pada arsitektur dan cara prediksi objek. Persamaan YOLOv2 dengan versi sebelumnya yaitu YOLOv2 juga melakukan pembagian citra dalam grid  $S \times S$ . YOLOv2 menggunakan arsitektur Darknet-19 yang memiliki 23 jaringan konvolutional dan 5 pooling layer tanpa menggunakan fully connected layer, dan menggunakan anchor box untuk memprediksi bounding box, sehingga jaringan dapat mengambil citra dengan ukuran yang bermacam-macam. Hasil dari jaringan ini akan memberikan koordinat dari bounding box dimana setiap anchor akan diprediksi berdasarkan nilai IOU. Bounding box yang memiliki nilai IOU terbesar, akan dijadikan sebagai hasil dari deteksi objek (Redmon et al., n.d.).

### 2.4. YOLOv3

YOLOv3 merupakan pengembangan terakhir dari YOLO yang dikembangkan oleh Joseph Redmon dan Ali Farhadi pada tahun 2018. Persamaan yang dimiliki oleh YOLOv2 dan YOLOv3 adalah penggunaan konsep bounding box menggunakan K-means pada anchor prior pada setiap deteksi yang dilakukan. Perbedaan yang dimiliki oleh YOLOv3 adalah penggunaan 3 ukuran yang berbeda, yaitu  $13 \times 13$ ,  $26 \times 26$ , dan  $52 \times 52$  yang diambil dari proses ekstrasi fitur dari YOLOv3, yaitu Darknet-53. Darknet-53 memiliki 53 convolution layer pada proses ekstrasi dan menggunakan residual block dibandingkan menggunakan pooling (Redmon et al., n.d.).

## 2.4. YOLOv4

YOLOv4 merupakan pengembangan terbaru dari YOLO yang dikembangkan oleh Alexey Bochkovskiy. Alexey Bochkovskiy mengatakan bahwa, “YOLOv4 lebih cepat 2 kali lipat dari dengan perbandingan performa. YOLOv4 juga meningkatkan AP dan FPS dari YOLOv3 sebesar 10% dan 12%” (Bochkovskiy et al., 2020). Tujuan utama dibangunnya YOLOv4 adalah untuk merancang kecepatan operasi yang cepat dari detektor objek dalam sistem produksi dan optimasi untuk komputasi paralel, daripada indikator teoritis volume komputasi rendah (BFLOP) (Bochkovskiy et al., 2020).



Gambar 2.14 Detektor Objek YOLOv4 (Bochkovskiy et al., 2020).

Peningkatan lain yang dilakukan oleh Alexey Bochkovskiy adalah penggunaan deteksi 2 tahap. Detektor tahap pertama terdiri dari input, backbone (VGG16, RESNet-50, Darknet53), neck (FPN, PANet, BiFPN), dan dense prediction (RPN, YOLO, SSD, FCOS). Selain itu, detektor tahap kedua memiliki

sparse prediction (Faster R CNN, R-FCN) untuk memprediksi objek dengan memahami kotak pembatas dan kelas-kelas pada objek.

## 2.5. YOLOv5

YOLOv5 didasarkan pada arsitektur deteksi YOLO dan menggunakan beberapa strategi optimasi algoritmik dari bidang jaringan saraf konvolusional. Secara otomatis mempelajari jangkar kotak pembatas, augmentasi data mosaik, dan subnet lintas tahap. Dibandingkan dengan solusi sebelumnya, model YOLO adalah pendeteksi objek pertama yang menggabungkan metode prediksi kotak pembatas dan label kelas dalam jaringan perbedaan ujung ke ujung. Jaringan YOLOv5 terdiri dari tiga komponen utama: 1) tulang punggung, 2) leher, dan 3) keluaran (Horvat & Gledec, n.d.).

Pertama, terminal masukan melakukan tugas prapemrosesan data seperti menambahkan data mosaik dan pengisian gambar adaptif. Untuk beradaptasi dengan kumpulan data yang

berbeda, YOLOv5 mengintegrasikan komputasi kerangka jangkar adaptif ke dalam inputnya, sehingga memungkinkannya menyesuaikan ukuran bingkai jangkar awal secara otomatis seiring dengan perubahan kumpulan data (Horvat & Gledec, n.d.).

Tulang punggung adalah jaringan saraf konvolusional yang menggabungkan dan membentuk fitur gambar pada berbagai perincian. Ini terutama menggunakan subjaringan lintas tahap (CSP) dan pengumpulan piramida spasial (SPP) untuk mengekstrak peta fitur dengan ukuran berbeda dari gambar masukan melalui beberapa konvolusi dan pengumpulan (Horvat & Gledec, n.d.).

Arsitektur BottleneckCSP mengurangi jumlah penghitungan dan meningkatkan kecepatan inferensi, dan struktur SPP memungkinkan ekstraksi fitur multi-skala untuk peta fitur yang sama dan menghasilkan peta fitur 3 skala, sehingga berkontribusi pada peningkatan akurasi pengenalan Masu (Horvat & Gledec, n.d.).

Jaringan saraf leher mewakili serangkaian lapisan yang memadukan dan menggabungkan fitur gambar untuk dijadikan prediksi. Struktur piramida fitur FPN dan PAN digunakan di jaringan leher. Struktur FPN menyampaikan fitur semantik yang kuat dari peta fitur teratas ke peta fitur bawah (Horvat & Gledec, n.d.).

Pada saat yang sama, struktur PAN menyampaikan lokalisasi fitur yang kuat dari peta fitur tingkat rendah ke peta fitur tingkat tinggi. Kedua struktur ini meningkatkan fitur yang diekstraksi dari lapisan jaringan berbeda dalam fusi tulang punggung, sehingga semakin meningkatkan kemampuan deteksi. Sebagai langkah deteksi terakhir, kami terutama menggunakan keluaran head untuk memprediksi target dengan ukuran berbeda di peta fitur (Horvat & Gledec, n.d.).

## 2.6. YOLOv6

Dibandingkan dengan YOLOv5, YOLOv6 memiliki banyak peningkatan dalam hal tulang belakang, leher, kepala, kehilangan fungsi, penyampaian kuantitatif dan strategi pelatihan. YOLOv6 menyediakan lima model terlatih (YOLOv6-N, YOLOv6-T, YOLOv6-S, YOLOv6-M, YOLOv6-L) untuk berbagai skenario aplikasi industri. Bagian pertama adalah struktur jaringan EfficientRep Backbone yang diusulkan oleh YOLOv6, yang menggunakan empat RepBlock yang terhubung secara seri dalam jaringan ekstraksi fitur. Tiga lapisan terakhir jaringan menghasilkan tiga cabang hasil ekstraksi fitur, C3, C4, dan C5, yang diumpankan ke Neck (Norkobil Saydirasulovich et al., 2023).

RepBlock adalah salah satu struktur terpenting dari tulang punggung EfficientRep, yang terinspirasi oleh jaringan repVGG dan menggunakan struktur multi-cabang selama fase pelatihan untuk meningkatkan kinerja fase ekstraksi fitur secara keseluruhan. Selama fase inferensi, struktur multi-cabang diubah secara setara menjadi struktur jalur tunggal, sehingga menghemat memori dan meningkatkan efisiensi selama inferensi (Norkobil Saydirasulovich et al., 2023).

Struktur jaringan meningkatkan kinerja ekstraksi fitur dengan mempercepat kecepatan perhitungan dan menghemat biaya operasional. Setiap RepBlock terdiri dari struktur tiga cabang, dengan tambahan cabang konvolusional paralel dan cabang pemetaan identitas ditambahkan ke setiap lapisan konvolusional, dan terakhir satu tambahan per elemen. Konvolusi sangat efisien pada GPU dan CPU umum, membuat jaringan deteksi lebih cepat dan tangguh (Norkobil Saydirasulovich et al., 2023).

## 2.7. YOLOv7

YOLO-V7 menunjukkan kinerja yang baik dalam mendeteksi kesalahan. Berdasarkan tugas deteksi cacat baja, artikel ini mengusulkan peningkatan algoritma YOLO-V7 yang bertujuan untuk meningkatkan akurasi dan kecepatan deteksi cacat pada permukaan baja (Wang et al., 2022).

Performa algoritme diverifikasi pada kumpulan data publik GC10-DET dan NEU-DET. YOLO-V7 didasarkan pada konsep seri YOLO dan kecepatan serta akurasinya melebihi semua detektor objek yang dikenal dari 5 FPS hingga 160 FPS. YOLO-V7 adalah kombinasi sempurna antara kecepatan dan presisi, menjadikannya favorit industri. Metode ini didasarkan pada YOLO-V7 dan mengusulkan algoritma baru untuk mendeteksi cacat permukaan strip baja (Wang et al., 2022).

Secara keseluruhan, YOLO-V7 pertama-tama mengubah ukuran gambar masukan menjadi 640 x 640, memasukkannya ke jaringan tulang punggung, kemudian mengeluarkan tiga lapisan peta fitur dengan ukuran berbeda melalui jaringan utama, dan kemudian menggunakan RepConv untuk mengeluarkan hasil prediksi. Bagian tulang punggung YOLO-V7 terutama menggunakan ELAN, struktur MP, dan fungsi aktivasi Silu (Wang et al., 2022).

Struktur ELAN dapat dilatih dan dikonvergensi secara efisien dengan mengontrol jalur gradien jaringan yang lebih dalam. Hal yang sama berlaku untuk ELAN-W. Tambahkan mekanisme perhatian ECA ke bagian bawah tulang belakang. Modul ini mengikuti bagian dari mekanisme perhatian SE. Peningkatan utama pada mekanisme perhatian SE adalah ukuran

kernel konvolusi satu dimensi dipilih secara adaptif untuk mempertahankan dimensi selama interaksi lintas saluran lokal, yang mengurangi kompleksitas jaringan dan meningkatkan kinerja model (Wang et al., 2022).

Di ECA, gambar fitur asli adalah masukan pertama, dan semua saluran dari gambar asli dirata-ratakan dan digabungkan secara global. Kemudian, konvolusi satu dimensi cepat berukuran  $Q$  digunakan untuk menghasilkan bobot saluran, dan probabilitas saluran berbeda yang sesuai dihitung dan dibandingkan dengan gambar asli. Fitur masukan dikalikan sebagai masukan ke tingkat berikutnya. Dalam metode ini, nilai  $Q$  ditentukan melalui fungsi fitting, dan nilainya sebanding dengan dimensi saluran  $C$  (Wang et al., 2022).

## 2.8. YOLOv8

YOLOv8 adalah jenis model jaringan saraf untuk deteksi objek dan termasuk dalam keluarga model You Only Look Once (YOLO). Tujuan dari rangkaian ini adalah untuk mencapai deteksi objek secara real-time, mengidentifikasi dan melokalisasi objek dalam gambar secara akurat, dan menggambarkan kotak pembatas dan kategori objek (Tang & Guo, 2024).

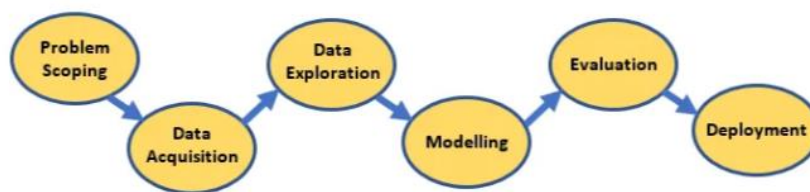
Dibangun di atas YOLOv7, YOLOv8 meningkatkan akurasi dan kinerja deteksi dengan menyempurnakan arsitektur jaringan, mengoptimalkan parameter model, dan mengoptimalkan strategi pelatihan. Model ini biasanya menggunakan arsitektur jaringan saraf konvolusional (CNN) dan memanfaatkan kotak jangkar serta jaringan piramida fitur untuk mendeteksi objek dengan berbagai ukuran dan rasio aspek (Tang & Guo, 2024).

YOLOv8 menggabungkan teknik dan strategi canggih untuk mencapai akurasi deteksi yang lebih tinggi dan kecepatan pemrosesan yang lebih cepat. Dengan memperkenalkan metode ekstraksi fitur baru, peningkatan fungsi kerugian, struktur model yang lebih efisien, dan teknik pelatihan yang efektif, YOLOv8 mewakili langkah maju dan kemajuan yang signifikan di bidang deteksi objek. Arsitektur jaringan YOLOv8 terdiri dari tiga bagian utama: tulang punggung, leher, dan kepala (Tang & Guo, 2024).



## BAB III METODOLOGI

Sistem deteksi yang dikembangkan berkaitan erat dengan model *deep learning* yang juga merupakan otak dari aplikasi. Model *deep learning* berisi pengetahuan dari data *train* yang di latih menggunakan algoritma YOLO sehingga menjadi model siap pakai [5]. Sistem tersebut dikembangkan menggunakan *AI project cycle*, merupakan sebuah metode yang dilakukan untuk membuat proyek berbasis AI secara utuh. Metode ini akan memandu secara terorganisir bagaimana pembuatan dan penyelesaian proyek AI [6]. Berikut merupakan langkah-langkah metode tersebut:



Gambar. Metode AI Project Cycle

### 3.1 Problem Scopping

Dalam tahapan ini dilakukan untuk menilai seberapa besar permasalahan yang ada dengan penekanan pada strategi penyelesaian. Langkah pertama adalah memahami dan menganalisis masalah secara menyeluruh, kemudian menetapkan tujuan untuk mengatasi masalah secara efektif. Pendekatan 4Ws digunakan untuk memandu penentuan ruang lingkup masalah dan mempermudah dalam penyelesaian masalah dengan Mengidentifikasi *who*, *what*, *where* dan *why*. membantu menjelaskan aspek-aspek penting. *Who* menjelaskan keterlibatan masalah yang relevan, *what* mengidentifikasi sifat masalah, *where* kapan masalah itu terjadi, dan *why* itu terjadi serta solusi yang dibutuhkan. Dengan menjelaskan setiap elemen 4W secara rinci, kita akan menganalisis masalah secara menyeluruh dan merumuskan solusi khusus untuk mengatasi masalah tersebut. Berikut scope dalam pengembangan aplikasi ini:

Tabel. Scope pengembangan proyek

Who? Pemangku	Petani apel: Ingin memastikan bahwa apel mereka memenuhi standar kualitas untuk mendapatkan harga terbaik di pasar.
------------------	---

Kepentingan Utama	Produsen dan distributor apel: Membutuhkan proses otomatis untuk menilai kualitas dan kuantitas apel secara cepat dan akurat untuk efisiensi rantai pasokan.
	Pengecer: Memerlukan apel dengan kualitas konsisten untuk memenuhi ekspektasi pelanggan.
	Konsumen: Mengharapkan apel berkualitas tinggi saat membeli dari pengecer.
	Tim pengembang teknologi: Bertanggungjawab atas pengembangan, implementasi, dan pemeliharaan sistem otomatis ini.
What? Deskripsi Masalah	Masalah yang ada: Deteksi kualitas dan penghitungan apel secara manual saat ini memakan waktu, tidak konsisten, dan rentan terhadap kesalahan manusia.
	Solusi yang dicari: Mengembangkan sistem otomatis yang menggunakan teknologi pengenalan citra dan pembelajaran mesin untuk mendeteksi kualitas apel (berdasarkan warna, ukuran, dan cacat) dan menghitung jumlah apel secara akurat.
Where? Lokasi Implementasi	Ladang dan kebun apel: Dimana apel dipanen dan pertama kali diperiksa.
	Fasilitas pengolahan dan pengemasan: Tempat apel diperiksa, diklasifikasikan, dan dikemas untuk distribusi lebih lanjut.
	Pusat distribusi: Dimana apel diperiksa sebelum dikirim ke pengecer.
	Toko ritel: Dimana kualitas apel perlu dipastikan sebelum sampai ke tangan konsumen.
Why? Alasan Pengembangan	Efisiensi proses: Mengurangi waktu dan tenaga yang dibutuhkan untuk memeriksa dan menghitung apel secara manual.

	Akurasi dan konsistensi: Meningkatkan akurasi deteksi kualitas apel dan konsistensi hasil dibandingkan dengan metode manual.
	Pengurangan biaya: Mengurangi biaya tenaga kerja dan potensi kerugian dari kesalahan manusia dalam proses penilaian kualitas.
	Peningkatan kepuasan pelanggan: Memastikan bahwa apel yang sampai ke konsumen memiliki kualitas tinggi, meningkatkan kepercayaan dan kepuasan pelanggan.
	Keunggulan kompetitif: Memberikan keunggulan kompetitif bagi produsen dan pengecer melalui penggunaan teknologi canggih dalam proses produksi dan distribusi.

### 3.2 Data Acquisition

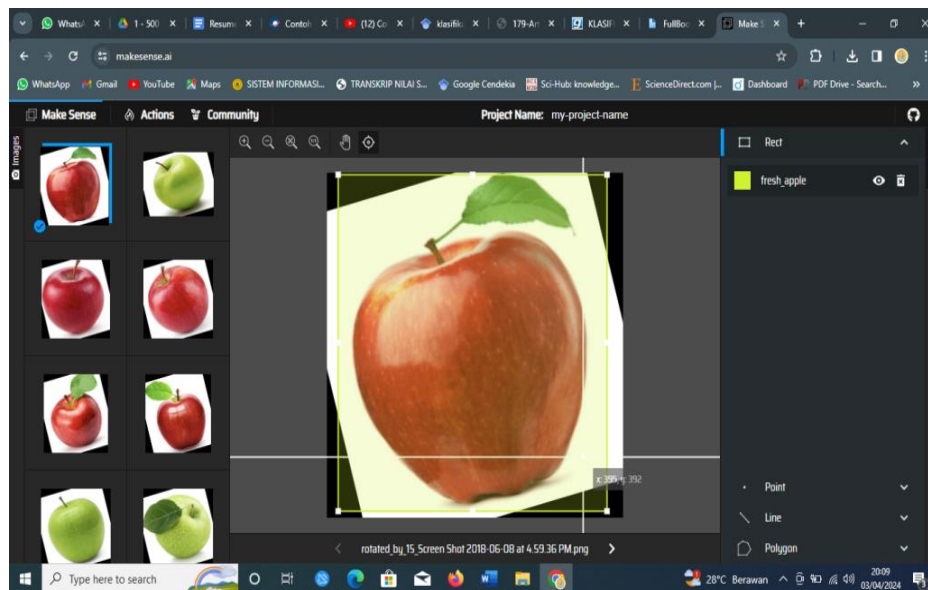
Setelah menetapkan batasan masalah dan solusinya, langkah selanjutnya merupakan akuisisi data. Pengumpulan data dilakukan untuk memperoleh bahan yang dibutuhkan dalam rangka mencapai tujuan proyek. Dataset yang digunakan pada aplikasi ini merupakan dataset "*Fresh and Stale Images of Fruits and Vegetables*" yang diperoleh dari situs Kaggle. Dataset ini berisi gambar buah-buahan yang dikelompokkan menjadi beberapa *class*. Dalam proyek ini tidak menggunakan semua *class* gambar pada dataset tersebut, melainkan hanya menggunakan 2 kategori yaitu *Apple Fresh* dan *Apple stale*. Total keseluruhan gambar yang digunakan yaitu 3000 data dengan masing-masing setiap kelas berjumlah 1500 data. Keseluruh citra pada dataset ini memiliki ukuran 512x512 piksel seperti pada gambar 1.



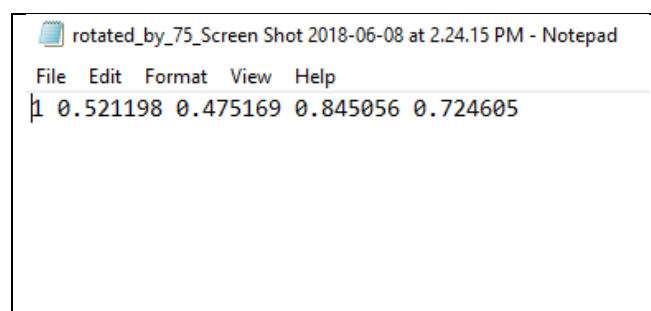
Gambar. Dataset yang digunakan

### 3.3 Data Exploration

Tahap exploration dilakukan explorasi pada dataset yang sudah diperoleh ditahap sebelumnya. Proses anotasi pada setiap gambar dilakukan menggunakan software “Make Sense AI” seperti pada gambar 5. Hasil akhir anotasi citra YOLO akan tersimpan menggunakan format file .txt dengan nama yang sama untuk setiap file gambar dalam direktori yang sama. Setiap file .txt berisi anotasi untuk file gambar terkait, yaitu kelas objek, koordinat objek, tinggi dan lebar contoh format anotasi citra YOLO pada file .txt seperti ditunjukkan pada Gambar .



Gambar. Proses anotasi menggunakan Make Sense AI



Gambar. Format anotasi YOLO

Hasil pada tahap eksplorasi struktur dataset yang sudah dianotasi akan terbagi menjadi data testing, data validation dan data training yang tersimpan pada sebuah folder dengan folder label seperti ditunjukkan pada Gambar.



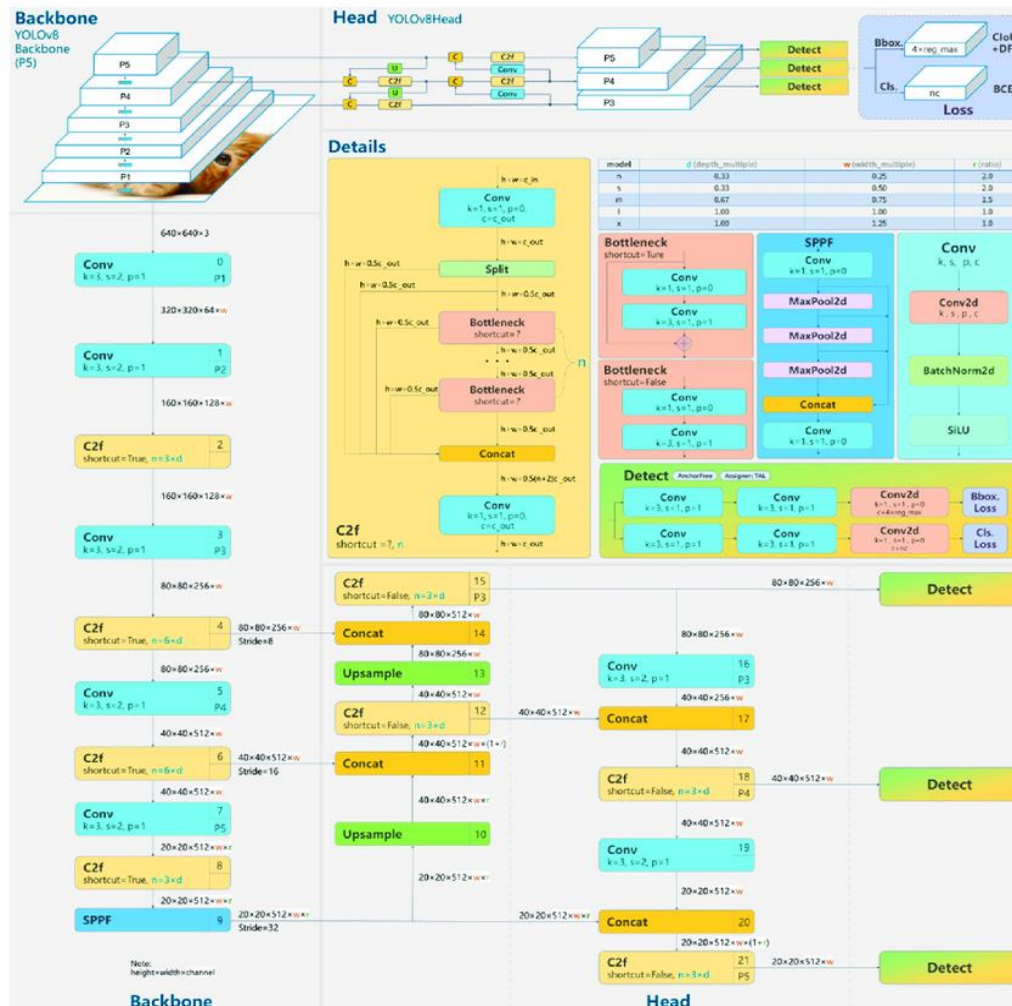
Gambar. Struktur splitting dataset

### 3.4 Modelling

Proyek ini menggunakan algoritma deteksi objek *real-time* yolo (you only look once) yang dikembangkan oleh Joseph Redmon dan Ali Farhadi pada tahun 2015. Digunakannya yolo dalam project ini bertujuan untuk mengembangkan sistem yang dapat mendeteksi kualitas buah apel secara otomatis dan menghitung jumlahnya. Yolo memproses seluruh citra dalam satu lintasan maju melalui *Convolutional Neural Network* (CNN) dan memprediksi kotak pembatas objek serta probabilitas kelas secara *real-time*[7]. Yolo membagi citra input menjadi kisi  $s \times s$  (Yanto et al. 2023). Jika pusat objek jatuh ke dalam sel kisi, sel kisi tersebut bertanggung jawab untuk mendeteksi objek tersebut. setiap sel kisi memprediksi kotak pembatas dan hasil *confidence* untuk kotak tersebut. Hasil *confidence* ini mencerminkan seberapa yakin model bahwa kotak tersebut berisi objek dan seberapa akurat perkiraan kotak tersebut menurutnya [2]. Yolo memprediksi beberapa kotak pembatas per sel kisi.

Yolov8 adalah peningkatan dari versi sebelumnya yang termasuk meningkatkan kinerja, membuat model lebih cepat, akurat, dan mudah digunakan. *Backbone model* Yolov8 tetap menggunakan modul CSP dari Yolov5. Modul C2f digunakan untuk mengekstraksi fitur visual. Pada Yolov8, struktur konvolusi Cbs  $1 \times 1$  di tahap up-sampling PAN-FPN pada Yolov5 dihapus, dan modul C3 digantikan oleh modul C2f [4]. Pada Yolov8, kepala terpisah menggunakan dua konvolusi masing-masing untuk klasifikasi dan regresi, serta memanfaatkan konsep DFL. Pembaruan terpenting pada model Yolov8 adalah adopsi metode tanpa jangkar (*Anchor-Free*) dan penggunaan pembelajaran penyelarasan tugas untuk menyelaraskan tugas klasifikasi (cls) dan regresi (reg). Jangkar yang selaras seharusnya dapat diposisikan dengan akurat. Model Yolov8 menggunakan metrik penyelarasan jangkar yang baru. Metrik penyelarasan jangkar diperoleh dengan mengalikan skor cls dan IoU antara bingkai prediksi

dan bingkai nyata. Metrik Penyelarasan Ini Diintegrasikan Dalam Pembagian sampel dan fungsi kerugian untuk mengoptimalkan prediksi setiap jangkar secara dinamis. Model Yolov8 menggunakan kerugian VFL untuk klasifikasi dan DFL + kerugian CIoU untuk klasifikasi. Arsitektur model Yolov8 memiliki 105 layer seperti pada gambar 8.



Gambar. Arsitektur odel YOLOv8

## DAFTAR PUSTAKA

- Asiati, D., Bayu Aji, G., Ningrum, V., Edi Kurniawan, F., Luther Aruan, N., & Astrelina Purba, Y. (n.d.). RESEARCH SUMMARY EMPLOYMENT OPPORTUNITIES AND HUMAN RESOURCES DEVELOPMENT IN DIGITAL ERA: A Case Study in Industrial Sector.
- Kapach, K., Barnea, E., Mairon, R., Edan, Y., & Ben-Shahar, O. (2012). Computer vision for fruit harvesting robots - state of the art and challenges ahead. *International Journal of Computational Vision and Robotics*, 3(1-2), 4–34.  
<https://doi.org/10.1504/IJCVR.2012.046419>
- J. Terven, D.-M. Córdova-Esparza, and J.-A. RomeroGonzález, <A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS,= Machine Learning and Knowledge Extraction, vol. 5, no. 4, pp. 1680–1716, Nov. 2023, doi: 10.3390/make5040083
- Arivazhagan, S., Shebiah, R. N., Selva, N. V., & Ganesan, L. (2019). Detection of fruit quality using image processing techniques. *Journal of Agriculture and Food Technology*, 6(3), 45-52.
- Bargoti, S., & Underwood, J. P. (2017). Deep fruit detection in orchards. *Robotics and Autonomous Systems*, 93, 120-130.
- Jocher, G., et al. (2023). YOLOv8: Advances in real-time object detection. *arXiv preprint arXiv:2302.03612*.
- Li, X., Zhang, Y., & Wang, Z. (2021). Video-based fruit detection and tracking for robotic harvesting systems. *Computers and Electronics in Agriculture*, 187, 106243.
- Mahesh, H. M., Kumar, J. K., & Ramesh, K. S. (2020). Fruit maturity and freshness detection using deep learning. *International Journal of Advanced Computer Science and Applications*, 11(4), 341-349.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779-788.
- Sa, I., Ge, Z., Dayoub, F., Upcroft, B., & Perez, T. (2018). Deepfruits: A fruit detection system using deep neural networks. *Sensors*, 16(8), 1222.

