

## **BAB II**

### **TINJAUAN PUSTAKA**

#### 2.1 Deteksi Objek

Deteksi objek merupakan proses identifikasi lokasi objek dalam suatu gambar dan penetapan setiap objek ke kelas yang sesuai (Xiao dkk., 2020). Teknologi pengenalan objek di pada gambar digital ini telah banyak berkembang pesat dan akan terus lebih maju lagi seiring dengan kemajuan *Artificial Intelligence*. Salah satu pemanfaatan teknologi *AI* dalam pengolahan citra yaitu komputer dapat mendeteksi objek-objek yang berada pada gambar maupun video (Tessa Ningrum dkk., 2024).

Manfaat deteksi objek ini banyak sekali sesuai dengan tujuan pembuatannya dan bidangnya. Misalnya dalam bidang industri, deteksi objek digunakan dalam proses sortir untuk mengenali dan memilah produk yang sesuai standar dan yang tidak secara otomatis. Seperti yang telah dilakukan dalam penelitian klasifikasi kematangan buah sawit dengan *Deep Learning* (Kurniawan dkk., 2023).

Pemanfaatan teknologi deteksi dalam bidang keamanan juga telah dikembangkan seperti deteksi pelanggaran peraturan lalu lintas dengan kamera di jalan seperti yang sudah banyak diterapkan di sekitar lampu merah di kota-kota besar. Beberapa penelitian lainnya juga telah banyak mengembangkan model deteksi objek seperti deteksi emosi melalui ekspresi wajah, deteksi usia, dan deteksi jenis kelamin (Siam dkk., 2022), (Hassouneh dkk., 2020), (Sachi dkk., 2022).

### 2.1.1 Deteksi Wajah

Deteksi wajah (*face detection*) merupakan bagian dari teknologi deteksi objek yang berfokus pada menentukan lokasi dan ukuran wajah manusia dalam gambar digital (Kumar dkk., 2019). Pengembangan *face detection* juga memungkinkan komputer dalam mendeteksi ekspresi sampai emosi manusia.

Ekspresi wajah adalah perubahan pada wajah individu yang muncul sebagai reaksi terhadap emosi yang mereka rasakan, niat yang mereka miliki, atau interaksi sosial yang sedang berlangsung (Sidik dkk., 2021). Model deteksi emosi manusia yang ada saat ini sudah banyak diterapkan untuk mendeteksi banyak emosi pada manusia secara umum yaitu marah, sedih, senang, takut, terkejut, jijik, dan lainnya (Azhari & Fitriyani, 2020), (Hassouneh dkk., 2020).

### 2.1.2 Deteksi Tingkat Antusiasme

Antusiasme merupakan perasaan positif yang menunjukkan ketertarikan seseorang terhadap sesuatu (Suciati, 2018). Pemahaman mengenai tingkat antusiasme atau ketertarikan dapat diperoleh dari emosi yang muncul sebagai respons otomatis manusia yang terlihat pada wajah mereka (Manizar, 2016).

Deteksi antusiasme ini termasuk deteksi emosi dari ekspresi wajah manusia. Perbedaannya adalah pada pengelompokan emosi yang dijadikan beberapa kelas sesuai dengan tingkat ketertarikannya. Antusiasme dalam pembelajaran daring tingkat ini dilihat dari seberapa tertariknya seseorang ke arah depan atau kamera saat sedang menyimak materi pembelajaran. Kusumastuti dkk., 2020 mengelompokkan tingkat ketertarikan ini dikelompokkan menjadi 3 kelas dalam

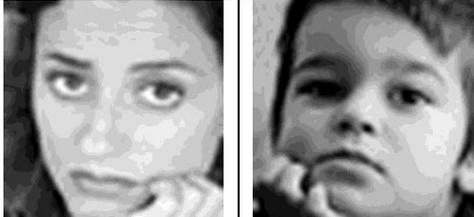
penelitiannya tentang klasifikasi ketertarikan anak PAUD melalui ekspresi wajahnya pada tabel 2.1.

Tabel 2.1 Daftar Kelas Ketertarikan Anak PAUD (Kusumastuti dkk., 2020)

Nama Kelas	Contoh Gambar
Tertarik (T)	
Cukup Tertarik (CT)	
Tidak Tertarik (TT)	

Penelitian ini akan menambahkan 2 kelas lagi dengan tujuan agar kelas lebih kompleks dan lebih efektif dalam klasifikasi preferensi seseorang yaitu dengan menggunakan skala 5 kelas (Manderscheid & Lee, 2023). Kelas yang ditambahkan yaitu kelas ‘kurang tertarik’ dan ‘sangat tertarik’ sehingga jika diurutkan dari positif ke negatif maka akan menjadi ‘sangat tertarik’, ‘tertarik’, ‘cukup tertarik’, ‘kurang tertarik’, dan ‘tidak tertarik’ yang dapat terlihat pada tabel 2.2.

Tabel 2.2 Kelas Tingkatan Antusiasme

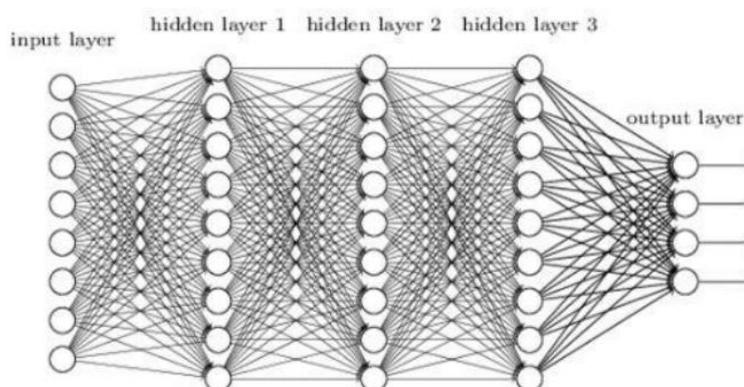
Nama Kelas	Contoh Gambar		Deskripsi
Sangat Tertarik (1)			Ditunjukkan ekspresi suka dengan tatapan menghadap ke dapan layar
Tertarik (2)			Ditunjukkan ekspresi netral dengan tatapan menghadap ke dapan layar
Cukup Tertarik (3)			Ditunjukkan ekspresi kurang suka tetapi tatapan masih menghadap ke dapan layar
Kurang Tertarik (4)			Ditunjukkan ekspresi kurang suka dengan wajah menghadap ke depan tetapi pandangan mata ke arah lain
Tidak Tertarik (5)			Ditunjukkan ekspresi kurang suka dengan wajah yang bahkan tidak menghadap ke layar

## 2.2 Convolutional Neural Network (CNN)

*Convolutional Neural Network* (CNN) merupakan salah satu algoritma *deep learning* jenis *neural network* yang biasa digunakan untuk analisis data visual, seperti gambar ataupun video. *Deep Learning* merupakan cabang dari *machine learning* yang berfokus pada pengembangan dan penggunaan arsitektur jaringan

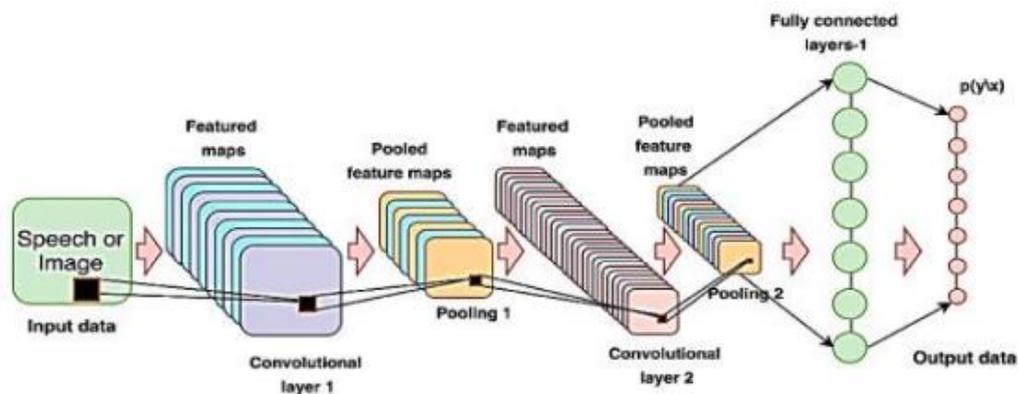
saraf yang sangat dalam dan terdiri dari beberapa lapisan (*layer*) yang saling terhubung. *Deep learning* berusaha untuk mempelajari abstraksi tingkat tinggi dalam data dengan menggunakan arsitektur hierarkis (Guo dkk., 2016). Pendekatan *deep learning* dapat mengidentifikasi pola yang rumit dan mendalam dalam data, memungkinkan komputer untuk mengambil keputusan atau membuat prediksi yang lebih kompleks dan akurat.

*Deep Learning* memungkinkan komputer agar dapat belajar dari data-data yang tidak terstruktur dan sangat kompleks kemudian dapat mengerjakan tugas atau membuat keputusan yang membutuhkan pemahaman tingkat tinggi dari data-data tersebut. *Deep Learning* banyak digunakan dalam deteksi maupun klasifikasi objek dari gambar ataupun video.



Gambar 2.1 *Deep Neural Network* (Charli dkk., 2020)

Algoritma *Convolutional Neural Network* (CNN) terdiri dari lapisan-lapisan konvolusi yang bertanggung jawab untuk mengekstrak fitur-fitur hierarkis dari input, diikuti oleh lapisan-lapisan pooling yang digunakan untuk mengurangi dimensi data (Nurchahyo & Iqbal, 2022). Jaringan ini terdiri dari neuron-neuron berukuran kecil yang terdapat pada setiap lapisan dalam arsitektur model yang dirancang untuk memproses data input dalam bentuk bidang reseptif.



Gambar 2.2 Arsitektur Jaringan Dasar CNN (Nurchahyo & Iqbal, 2022)

Gambar 2.2 menunjukkan arsitektur berlapis dari jaringan CNN dasar. Filter bertindak sebagai koneksi lokal yang diaplikasikan pada input dan berbagi parameter yang sama (bobot  $W_i$  dan bias  $n_i$ ) untuk menghasilkan peta fitur ( $z_i$ ), yang masing-masing memiliki ukuran  $a - b - 1$ . Lapisan konvolusi menghitung produk titik antara bobot dan input yang diberikan. Parameter untuk bobot  $W_i$  dan bias  $n_i$  untuk pembuatan peta  $z_i$  untuk fitur  $i$  dengan ukuran  $a - b - 1$  dapat diberikan sebagai:

$$z^i = g(W^i * r + n^i) \quad (1)$$

Output dari lapisan konvolusi didapatkan dengan fungsi aktivasi  $f$  atau metodologi non-linear. Subsampling dilakukan pada setiap lapisan untuk menghasilkan peta dan mengurangi jumlah parameter dalam jaringan yang bertujuan untuk membantu mengontrol *overfitting* dan meningkatkan proses pelatihan. Proses pooling dilakukan pada elemen  $pp$  (juga dikenal sebagai ukuran filter) untuk menyatukan semua peta fitur yang berdekatan. Tahap akhir lapisan terhubung sepenuhnya seperti pada jaringan saraf lainnya. Lapisan ini mengambil fitur tingkat rendah dan tingkat menengah sebelumnya, dan menghasilkan abstraksi

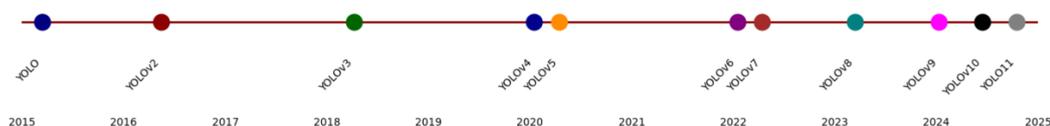
tingkat tinggi dari data input. Lapisan terakhir, yang juga dikenal sebagai SVM atau *Softmax*, digunakan untuk menghasilkan skor klasifikasi dalam bentuk probabilitas untuk terkait dengan kelas tertentu.

### 2.3 YOLO (You Only Look Once)

YOLO (*You Only Look Once*) merupakan algoritma deteksi objek yang diperkenalkan pada tahun 2015 oleh Joseph Redmon dan Ali Farhadi. Algoritma ini memproses seluruh gambar dalam satu kali pengolahan melalui jaringan saraf *convolutional* (CNN), dan secara *real-time* memprediksi kotak pembatas objek serta probabilitas kelas. Sistem deteksi ini bekerja dengan memanfaatkan *classifier* atau *localizer* yang disesuaikan untuk melakukan deteksi. Model diterapkan pada gambar di berbagai lokasi dan skala. Daerah dengan skor tertinggi pada gambar akan dianggap sebagai hasil deteksi (Rahma dkk., 2021).

#### 2.3.1 Perkembangan Algoritma YOLO

Algoritma YOLO dari awal diperkenalkan hingga saat ini telah mengalami perkembangan pesat. Meskipun penulis awalnya Joseph Redmon telah menghentikan penelitiannya di bidang visi komputer setelah merilis YOLO-v3, konsep inti YOLO yang efektif dan potensial telah terus dikembangkan oleh para peneliti lainnya.



Gambar 2. 3 Perkembangan Algoritma YOLO (Sapkota dkk., 2024)

Berikut pengenalan singkat perkembangan algoritma YOLO:

1) YOLOv1 (2015):

YOLOv1 dikembangkan oleh Joseph Redmon dan Ali Farhadi. YOLOv1 ini memperkenalkan deteksi objek dalam satu lintasan maju melalui jaringan saraf, memungkinkan deteksi yang cepat dan efisien.

2) YOLOv2 (2016):

YOLOv2 dikenal juga sebagai YOLO9000. YOLOv2 memperkenalkan *anchor boxes* dan *batch normalization* sehingga model mampu mendeteksi lebih banyak kelas objek dengan peningkatan akurasi dan kecepatan.

3) YOLOv3 (2018):

YOLOv3 ini menggunakan arsitektur Darknet-53 yang lebih dalam dan kuat. Meningkatkan deteksi objek kecil dan menawarkan peningkatan signifikan dalam akurasi secara keseluruhan.

4) YOLOv4 (2020):

YOLOv4 menambahkan berbagai teknik terbaru seperti *mish activation* dan *mosaic data augmentation*. Meningkatkan performa deteksi dalam hal kecepatan dan akurasi.

5) YOLOv5 (2020):

YOLOv5 dikembangkan oleh Ultralytics. Fokus pada kemudahan penggunaan dan integrasi dengan *framework* yang berbeda. Menawarkan berbagai varian model untuk berbagai kebutuhan komputasi.

6) YOLOv6 & YOLOv7, (2022):

Setiap versi YOLOv6 & YOLOv7 membawa perbaikan dalam kecepatan, akurasi, dan efisiensi komputasi. YOLOv7 dikenal karena optimisasi yang lebih baik untuk aplikasi *real-time*.

7) YOLOv8,

YOLOv8 membawa peningkatan pada arsitektur dan kemampuan deteksi. Model ini menggabungkan *backbone convolutional* dan *Feature Pyramid Network* (FPN) untuk mendukung deteksi multi-skala yang lebih baik, yang sangat penting dalam mendeteksi objek dengan berbagai ukuran dalam gambar.

8) YOLOv9

YOLOv9 menggabungkan beberapa modul penting, termasuk CSPNet, ELAN, dan GELAN, untuk meningkatkan integrasi fitur dan efisiensi komputasi. CSPNet (*Cross Stage Partial Network*) meminimalkan redundansi komputasi dengan membagi aliran data melalui dua jalur, sehingga meningkatkan efisiensi tanpa mengorbankan akurasi deteksi. ELAN (*Enhanced Linear Aggregation Network*) membantu dalam penggabungan fitur yang lebih baik dari berbagai lapisan, memperkuat kemampuan model dalam mengenali pola objek di berbagai skala. Modul GELAN (*Generalized ELAN*) membawa peningkatan lebih lanjut dengan memaksimalkan pemrosesan dan kombinasi fitur dari lapisan yang lebih kompleks.

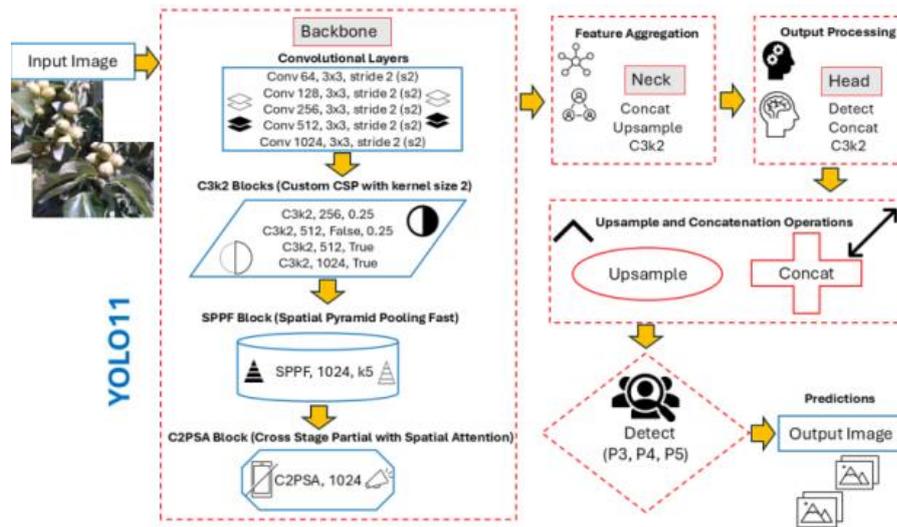
9) YOLOv10

YOLOv10 memperkenalkan arsitektur baru dengan strategi penetapan label ganda dan *Path Aggregation Network* (PAN) untuk meningkatkan presisi dalam lokalisasi dan klasifikasi objek. Strategi penetapan label ganda memungkinkan

model untuk secara adaptif memberikan label pada objek melalui dua pendekatan yang berbeda, membantu mengurangi kesalahan prediksi, terutama pada objek yang sulit dibedakan. Sementara itu, *Path Aggregation Network* (PAN) meningkatkan integrasi fitur dari berbagai level dalam model, sehingga informasi dari lapisan yang lebih rendah dan lebih tinggi dapat digabungkan secara efektif.

#### 10) YOLOv11

Kerangka kerja YOLOv11 merevolusi deteksi objek dengan memperkenalkan jaringan saraf terpadu yang secara bersamaan menangani regresi *bounding box* dan klasifikasi objek, menghilangkan kebutuhan akan beberapa tahap pemrosesan terpisah dan memungkinkan pelatihan *end-to-end*. Arsitektur YOLOv11 terdiri dari tiga komponen utama: *backbone*, *neck*, dan *head*. *Backbone*, yang biasanya terdiri dari jaringan saraf konvolusional, berfungsi sebagai ekstraktor fitur utama, mengubah data gambar mentah menjadi peta fitur multi-skala. *Neck* kemudian memproses peta fitur ini dengan lapisan yang dirancang untuk menggabungkan dan meningkatkan representasi fitur di berbagai skala. Terakhir, *head* menghasilkan prediksi akhir untuk lokasi dan klasifikasi objek berdasarkan peta fitur yang telah diproses. Berdasarkan fondasi ini, YOLO11 memperkenalkan penyempurnaan arsitektur dan optimasi parameter, meningkatkan kinerja dan akurasi deteksi dibandingkan versi sebelumnya (Khanam & Hussain, 2024). Arsitektur YOLOv11 dapat dilihat pada Gambar 2.7.



Gambar 2. 4 Arsitektur YOLOv11 (Sapkota dkk., 2024)

Kemudian komparasi/perbandingan dari algoritma YOLOv1 sampai dengan YOLOv11 disajikan pada Tabel 2.3.

Tabel 2. 3 Perkembangan Algoritma YOLO (Khanam & Hussain, 2024)

Algoritma YOLO	Tahun Rilis	Tugas	Kontribusi	Kerangka Kerja
YOLO	2015	<i>Object Detection, Basic Classification</i>	<i>Single-stage object detector</i>	Darknet
YOLOv2	2016	<i>Object Detection, Improved Classification</i>	<i>Multi-scale training, dimension clustering</i>	Darknet
YOLOv3	2018	<i>Object Detection, Multi-scale Detection</i>	<i>SPP block, Darknet-53 backbone</i>	Darknet
YOLOv4	2020	<i>Object Detection, Basic Object Tracking</i>	<i>Mish activation, CSPDarknet-53 backbone</i>	Darknet

Algoritma YOLO	Tahun Rilis	Tugas	Kontribusi	Kerangka Kerja
YOLOv5	2020	<i>Object Detection, Basic Instance Segmentation (via custom modifications)</i>	<i>Anchor-free detection, SWISH activation, PANet</i>	PyTorch
YOLOv6	2022	<i>Object Detection, Instance Segmentation</i>	<i>Self-attention, anchor-free OD</i>	PyTorch
YOLOv7	2022	<i>Object Detection, Object Tracking, Instance Segmentation</i>	<i>Transformers, E-ELAN reparameterisation</i>	PyTorch
YOLOv8	2023	<i>Object Detection, Instance Segmentation, Panoptic Segmentation</i>	<i>GANs, anchor-free detection</i>	PyTorch
YOLOv9	2024	<i>Object Detection, Instance Segmentation</i>	<i>PGI and GELAN</i>	PyTorch
YOLOv10	2024	<i>Object Detection</i>	<i>Consistent dual assignments for NMS-free training</i>	PyTorch
YOLOv11	2024	<i>Object Detection, Instance Segmentation, and Pose Estimation</i>	<i>Spatial Pyramid Pooling - Fast (SPPF), Cross Stage Partial with Spatial Attention (C2PSA), C3k2 block</i>	PyTorch

#### 2.4 Optimasi Hyperparameter

Masalah umum yang sering dihadapi dalam penggunaan algoritma CNN adalah potensi terjadinya *overfitting*. *Overfitting* terjadi ketika model CNN secara

berlebihan mempelajari fitur-detail dari data pelatihan yang mungkin tidak dapat digeneralisasi dengan baik pada data yang belum pernah dilihat sebelumnya yang disebabkan oleh kemampuan model CNN untuk mengingat fitur-detail yang spesifik pada data pelatihan, yang dapat menyebabkan kinerja yang buruk pada data uji atau data yang berbeda (Sanjaya & Ayub, 2020).

Salah satu upaya untuk membangun model deteksi yang memiliki tingkat akurasi yang tinggi dan menghindari terjadinya *overfitting* adalah dengan melakukan optimasi *hyperparameter*. *Hyperparameter* ini merupakan parameter yang ditetapkan sebelum proses pelatihan model dan tidak diperbarui selama pelatihan. Parameter ini sangat penting karena mempengaruhi bagaimana model belajar dan seberapa baik kinerjanya dalam mendeteksi objek. Optimasi *hyperparameter* ini telah dilakukan dalam beberapa penelitian seperti dalam mendeteksi kesalahan printer 3D menggunakan YOLOv8 dengan optimasi *hyperparameter* yang mendapat nilai rata-rata presisi tertingginya yaitu 89,7% setelah mencoba melatih model beberapa kali dengan pengaturan *hyperparameter* yang berbeda (Karna dkk., 2023).

Tabel 2.4 Perbandingan Model dengan Optimasi *Hyperparameter* (Karna, 2023)

Parameter		Improved Model (w/additional module) mAP <sub>val</sub> (50-95)				
Image Size	Batch Size	YOLOv8n	YOLOv8s	YOLOv8m	YOLOv8l	YOLOv8x
240	4	47.3%	40.0%	54.8%	60.5%	52.5%
	8	61.5%	46.9%	65.7%	61.6%	66.6%
	12	61.2%	55.3%	64.0%	64.3%	67.2%
	16	60.3%	61.6%	64.2%	73.7%	71.2%
	32	64.3%	65.1%	<b>76.8%</b>	72.5%	75.9%
480	4	63.4%	81.5%	75.7%	73.5%	66.5%
	8	75.1%	83.2%	78.4%	78.7%	76.3%
	12	81.6%	83.9%	80.8%	79.8%	77.6%
	16	82.3%	81.6%	83.1%	81.2%	81.0%
	32	82.8%	<b>83.9%</b>	81.5%	83.1%	81.2%

Parameter		Improved Model (w/additional module) mAP <sub>val</sub> (50-95)				
Image Size	Batch Size	YOLOv8n	YOLOv8s	YOLOv8m	YOLOv8l	YOLOv8x
640	4	76.8%	59.3%	77.6%	62.0%	61.3%
	8	74.7%	77.3%	79.3%	65.6%	61.7%
	12	81.7%	77.7%	81.4%	76.0%	67.1%
	16	82.4%	<b>89.7%</b>	83.7%	79.0%	69.5%
	32	76.2%	83.2%	75.6%	72.4%	63.9%

Pengaturan *hyperparameter* yang tepat seperti banyaknya *epoch* dalam proses *training* model dapat meningkatkan kemampuan model dalam mengenali pola yang relevan pada data, sehingga dapat meningkatkan akurasi prediksi pada data uji atau data baru. Penyesuaian *hyperparameter* ini diperlukan untuk menemukan kombinasi yang paling efektif dalam meningkatkan kinerja model (Karna et al., 2023). Beberapa *hyperparameter* penting yang mempengaruhi kinerja model yaitu:

1) *Image Size*

*Image Size* merupakan ukuran input gambar yang digunakan dalam pelatihan. Ukuran gambar yang berbeda dapat mempengaruhi kecepatan dan akurasi deteksi. Contoh ukuran gambar yang umum digunakan adalah 240, 480, dan 640 piksel. Ukuran gambar yang berbeda dapat diperoleh menggunakan teknik augmentasi data yaitu teknik untuk menghasilkan data baru dengan memanfaatkan data yang sudah ada. Salah satu penerapannya pada data gambar meliputi *resize*, *flip*, *random cropping*, atau rotasi pada gambar asli. Teknik augmentasi data ini membantu mengurangi *overfitting* karena hasil augmentasi biasanya mengandung kebisingan, sehingga model tidak terlalu bergantung pada data yang terlalu ideal (Sanjaya & Ayub, 2020).

2) *Batch Size*

*Batch Size* merupakan jumlah sampel yang diproses sebelum model memperbarui bobotnya. Ukuran *batch* yang lebih besar biasanya memerlukan lebih banyak memori tetapi dapat membuat pelatihan lebih stabil dan cepat (Karna dkk., 2023). Contoh ukuran *batch* yang digunakan adalah 4, 8, 12, 16, dan 32.

### 3) *Number of Epochs*

*Number of Epochs* merupakan jumlah kali seluruh dataset dilalui oleh algoritma pelatihan. Lebih banyak epoch bisa meningkatkan akurasi, tetapi juga meningkatkan risiko *overfitting* jika terlalu banyak (Adi Nugroho dkk., 2020).

## 2.5 Penelitian Terkait

Literatur yang dijadikan referensi dalam penelitian ini mencakup jurnal-jurnal ilmiah nasional dan internasional yang berfokus pada pengembangan model deteksi terkait objek emosi ataupun metode terkait algoritma YOLO. Referensi-referensi tersebut digunakan sebagai dasar untuk mengembangkan model deteksi antusiasme dengan metode YOLO dengan tujuan mencapai nilai akurasi yang optimal. Beberapa penelitian yang paling berkaitan dan menjadi referensi utama bagi penelitian ditampilkan pada tabel 2.4.

Tabel 2.5 *State of The Art* Penelitian Terkait

<i>State of The Art – Facial Emotional Detection</i>			
<b>Penulis</b>	<b>Judul Penelitian</b>	<b>Algoritma dan Dataset</b>	<b>Hasil</b>
(Azhari & Fitriyani,	Implementasi Algoritma Convolutional	Convolutional Neural Network dengan dataset	Tingkat akurasi model mencapai 83%

<i>State of The Art – Facial Emotional Detection</i>			
<b>Penulis</b>	<b>Judul Penelitian</b>	<b>Algoritma dan Dataset</b>	<b>Hasil</b>
2020)	Neural Network Dalam Deteksi Emosi Manusia Berdasarkan Ekspresi Wajah	FER2013 dengan 7 kategori kelas emosi	
(Kusumastuti dkk., 2020)	Klasifikasi Ketertarikan Anak PAUD Melalui Ekspresi Wajah Menggunakan Metode CNN	Convolutional Neural Network dengan dataset 243 gambar dengan 3 kategori kelas ketertarikan	Tingkat akurasi tertinggi mencapai 81,66%
(Kurniawan dkk., 2023)	Klasifikasi Tingkat Kematangan Buah Sawit Berbasis Deep Learning dengan Menggunakan Arsitektur YOLOv5	YOLOv5 dengan dataset 1500 gambar dengan 3 kategori kelas kematangan	Nilai mAP model mencapai 92%, akurasi.
(Motwani & S, 2023)	Human Activities Detection using DeepLearning Technique- YOLOv8	YOLOv8 dengan 2 dataset yaitu Face Detection Dataset & Benchmark menggunakan 2845 gambar, dan Mask-Dataset dengan	Mencapai akurasi 84,6% dengan 66,5% recall dalam waktu 0,030 detik

<i>State of The Art – Facial Emotional Detection</i>			
<b>Penulis</b>	<b>Judul Penelitian</b>	<b>Algoritma dan Dataset</b>	<b>Hasil</b>
		menggunakan 5237 gambar.	
(Salma & Hidayat, 2024)	Deteksi Antusiasme Siswa dengan Algoritma Yolov8 pada Proses Pembelajaran Daring	YOLOv8 dengan dataset FER2013 menggunakan 2000 dataset dijadikan 2 kategori kelas antusiasme	Akurasi tertinggi mencapai 95,3% dan waktu inferensi 62ms.
(Dewi dkk., 2024)	Deep Learning and YOLOv8 Utilized in an Accurate Face Mask Detection System	YOLOv8 dengan dataset FMD terdiri dari 853 gambar digabungkan dataset MMD mencakup 682 gambar	Dengan akurasi rata-rata sebesar 78,4%.
(Yanto dkk., 2023)	YOLOv8 Peningkatan Algoritma Untuk Deteksi Pemakaian Masker Wajah	YOLOv8 dengan dataset terdiri dari 1422 gambar yang diperoleh sendiri, mencakup 3 kelas: mask, nomask, dan badmask.	Akurasi untuk setiap kelas yaitu 94% untuk kelas badmask, 97% untuk kelas mask, dan 95% untuk kelas nomask.
(Abuzairi dkk., 2021)	Implementasi Convolutional Neural Network	YOLOV3 dengan dataset yang terdiri dari total 600	Mencapai hasil mAP@0.5 menunjukkan

<i>State of The Art – Facial Emotional Detection</i>			
<b>Penulis</b>	<b>Judul Penelitian</b>	<b>Algoritma dan Dataset</b>	<b>Hasil</b>
	Untuk Deteksi Nyeri Bayi Melalui Citra Wajah Dengan YOLO	gambar emosi bayi, yang mencakup emosi sedih, netral, dan nyeri.	97,9% untuk emosi sedih, 99,2% untuk emosi netral, dan 96,9% untuk emosi nyeri, dengan akurasi model sebesar 70%.
(Karna dkk., 2023)	Toward Accurate Fused Deposition Modeling 3D Printer Fault Detection Using Improved YOLOv8 With Hyperparameter Optimization	YOLOv8 dengan dataset 5000 gambar mencakup 6 kelas yaitu <i>rectangle</i> , <i>cube</i> , <i>cylinder</i> , <i>stringy</i> , <i>obstacle</i> , dan <i>unstick</i> .	Hasil terbaik didapat dengan model YOLOv8s dengan ukuran <i>image input</i> 640 dan <i>batch size</i> 16, mencapai hasil 89.7%.
(Awad dkk., 2024)	Early Diagnoses of Acute Lymphoblastic Leukemia Using YOLOv8 and YOLOv11 Deep Learning Models	YOLOv8 dan YOLOv11 dengan dataset 3,256 gambar yang berisi sampel sel darah putih	Hasil akurasi terbaik didapat dengan algoritma YOLOv11s dengan mencapai akurasi 98.8%.
(Sapkota dkk., 2024)	Comprehensive Performance Evaluation of YOLO11,	YOLOv8, YOLOv9, YOLOv10, dan YOLOv11 dengan	Hasil terbaik diraih YOLOv9 gelan-base dan YOLOv9 gelan-e dengan

<i>State of The Art – Facial Emotional Detection</i>			
<b>Penulis</b>	<b>Judul Penelitian</b>	<b>Algoritma dan Dataset</b>	<b>Hasil</b>
	YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments	dataset sebanyak 1,149 gambar apel hijau	skor 93,5% dan waktu inferensi tercepat diraih YOLOv11n dengan hanya 2,4 ms.

Tabel 2.4 menunjukkan keunggulan algoritma YOLO dibanding algoritma *Convolutional Neural Network (CNN)* sebelumnya, terutama terkait dengan keakuratan hasil dan kecepatan proses deteksi. Algoritma YOLO memproses seluruh gambar dalam satu pass, mempertimbangkan konteks global, dan mampu mendeteksi banyak objek dengan berbagai ukuran secara simultan. Selain itu, pelatihan *end-to-end* dan penggunaan *bounding box regression* meningkatkan lokalisasi dan deteksi objek secara keseluruhan, menjadikannya lebih cocok untuk aplikasi yang memerlukan kinerja cepat dan responsif.