

## Penentuan Epochs Hasil Model Terbaik: Studi Kasus Algoritma YOLOv8

Jasen Jonathan<sup>1\*</sup>, Dedy Hermanto<sup>2</sup>

<sup>1,2</sup>Universitas Multi Data Palembang, Indonesia

<sup>1</sup>[jasenjonathan123@email.com](mailto:jasenjonathan123@email.com), <sup>2</sup>[dedy@mdp.ac.id](mailto:dedy@mdp.ac.id)



### Histori Artikel:

Diajukan: 7 September 2024

Disetujui: 20 September 2024

Dipublikasi: 30 September 2024

### Kata Kunci:

CNN; Computer Vision; deep learning; epoch; YOLOv8.

### Digital Transformation

*Technology (Digitech) is an*

*Creative Commons License This work is licensed under a*

*Creative Commons Attribution-*

*NonCommercial 4.0 International (CC BY-NC 4.0).*

### Abstrak

Salah satu pengembangan *machine learning* yaitu *deep learning* merupakan salah satu metode inti dalam *artificial intelligence* yang sedang berkembang dengan pesat, dikarenakan kemampuannya dalam mempelajari informasi dalam jumlah besar. Salah satu cabang dari *deep learning* adalah *computer vision*, dan *Convolutional Neural Network (CNN)* yang merupakan metode yang paling banyak digunakan untuk melakukan pemrosesan citra. *YOLOv8* merupakan salah satu algoritma yang menggunakan *CNN* yang telah dimodifikasi sebagai dasar, *YOLOv8* merupakan algoritma *open-source* yang paling banyak digunakan dikarenakan menghasilkan hasil pengenalan objek yang akurat, cepat, dan mudah untuk di implementasikan. Proses pelatihan model dari *YOLOv8* membutuhkan perangkat yang cukup memadai dengan jumlah *epochs* yang ditentukan secara manual. Penelitian ini bertujuan untuk mengetahui jumlah *epoch* yang dibutuhkan dalam membuat model *YOLOv8* sesuai dengan kriteria yang di tentukan pada penelitian ini. Pelatihan akan dilakukan dengan menggunakan 50 *epochs*, 100 *epochs*, 150 *epochs*, 200 *epochs*, 250 *epochs*, dan 300 *epochs*. Pelatihan akan di jalankan dengan menggunakan *dataset* citra bibit ikan lele yang terdiri dari 753 gambar bibit ikan lele yang telah di anotasi. Pelatihan dijalankan dengan menggunakan *CPU Ryzen 5 4600H*. Berdasarkan dari hasil pelatihan didapatkan bahwa 50 *epochs* memiliki waktu pelatihan tercepat dengan hasil yang kurang baik. Hasil terbaik terdapat pada 200-300 *epochs* dengan rata-rata *precision* sebesar 96% dengan waktu pelatihan yang cukup lama.

### PENDAHULUAN

*Deep learning* merupakan salah satu pengembangan dari *machine learning* dan *artificial intelligence*, merupakan teknologi yang cukup berkembang di dunia saat ini, dikarenakan kemampuannya untuk mempelajari informasi dari data dalam jumlah yang besar (Sarker, 2021) (Hermawan et al., 2023). *Deep learning* telah mengembangkan pengetahuan tentang *artificial intelligence* dengan membuka kemajuan yang besar dalam berbagai bidang, salah satunya adalah *computer vision*. Perubahan yang terjadi dibuktikan dengan meningkatkan kemampuan untuk mempelajari representasi yang kompleks dari data (Chauhan et al., 2024).

*Computer vision* umumnya digunakan untuk menentukan aktivitas yang diawasi melalui *streaming multimedia*. Banyak metode mapan yang dapat digunakan untuk memprediksi dan menganalisis yaitu *supervised learning*, *un-supervised learning*, dan *semi-supervised learning* (Khan & Al-Habsi, 2020). Umumnya *computer vision* menggunakan metode *supervised learning* dalam konteks seperti mendeteksi objek. *Supervised learning* model di latih dengan menggunakan *dataset* yang diberikan label dengan tujuan untuk memprediksikan hasil atau mengklasifikasikan data (Sharma, 2024). *Computer vision* memiliki kemajuan yang pesat dalam bidang klasifikasi citra, segmentasi semantik, deteksi objek, dan *image super-resolution reconstruction* seiring dengan perkembangan yang pesat dari *deep convolutional neural networks (CNN)* (Zhao et al., 2024).

*Convolutional Neural Network (CNN)* merupakan salah satu metode *deep learning* yang paling banyak digunakan untuk melakukan pemrosesan citra. *CNN* telah menyelesaikan masalah kategorisasi dari objek dalam citra. Metode lebih fleksibel untuk digunakan dalam berbagai objek. Namun, metode *CNN* memiliki kekurangan di mana membutuhkan perangkat keras dengan kemampuan tinggi untuk menghasilkan model yang diinginkan (Peryanto et al., 2022). Metode *CNN* memiliki banyak cabang yang dikembangkan seiring dengan berkembangnya teknologi *computer vision* dan algoritma *deep learning*, salah satu metode yang populer saat ini adalah *You Only Look Once (YOLO)*. *YOLO* memiliki kapabilitas yang terbaik untuk mengenali objek dengan akurasi yang tinggi dan kecepatan waktu deteksi (Bhavana et al., 2024).

Sebelum menggunakan perangkat lunak berbasis *CNN* dibutuhkan sebuah model yang didapatkan dari internet atau dikembangkan sesuai kebutuhan. Kualitas dari model bergantung pada jenis *dataset* dan parameter yang digunakan ketika menghasilkan sebuah model, salah satu parameter yang penting adalah *epochs*. Epoch

terdiri dari *weight* dan *bias*. Nilai-nilai di inialisasikan secara sembarang ketika memulai pelatihan. Nilai tersebut kemudian disesuaikan terhadap nilai yang memiliki hasil yang benar. *Epoch* berarti satu melewati seluruh kumpulan data, di mana nilai dari *epoch* akan di berikan ke *epoch* berikutnya (Afaq & Rao, 2020). Untuk menghasilkan sebuah model yang baik, maka dibutuhkan pemilihan epoch yang sesuai untuk mendapatkan akurasi yang cukup baik. Untuk mendapatkan epoch yang baik juga berbanding lurus dengan waktu pembuatan model

Penelitian ini bertujuan untuk mengetahui jumlah *epoch* terbaik untuk digunakan dalam *YOLOv8* terhadap *dataset* yang telah disediakan.

### STUDI LITERATUR

Penelitian yang dilakukan oleh (Ajayi et al., 2023) membahas mengenai cara untuk mempertahankan hasil produksi pertanian tinggi dalam hal efisiensi dan keberlanjutan. Penelitian diajukan untuk menguji ketahanan dalam pelatihan *epochs* terhadap *You Only Look Once (YOLO) v5s*. Pengujian di evaluasi menggunakan citra *UAV* yang dianotasikan menggunakan *bounding box* dan kemudian dilatih dengan menggunakan *google colab* terhadap 100, 300, 500, 600, 700, dan 1000 citra. Penelitian bertujuan untuk menentukan performa optimal model di mana didapatkan hasil optimal pada 100 sampai 600 *epochs*. Sementara itu, penurunan sedikit terdeteksi ketika dinaikkan hingga 700 *epochs* yang menghasilkan akurasi, presisi, dan *recall* sebesar 65%, 43%, dan 43%, di bandingkan dengan akurasi, presisi, dan *recall* sebesar 67%, 78%, dan 34% yang diperoleh pada 600 *epochs*. Penurunan terus terjadi walaupun *epochs* telah dinaikkan menjadi 1000 di mana akurasi, presisi, dan *recall* adalah 65%, 45%, dan 40%. Hasil menunjukkan bahwa 600 *epochs* adalah nilai optimal dalam model *automatic crop and weed classification*.

Penelitian kedua dilakukan oleh (Ajayi & Ashi, 2023) membahas mengenai efek dari jumlah *training epochs* terhadap akurasi klasifikasi menggunakan *Faster Region-Based Convolutional Neural Network*. Penelitian dilakukan dengan menggunakan 10.000, 20.000, 100.000, 200.000, dan 242.000 *epochs* untuk mencari titik puncak tertinggi yang mendarat dalam proses mengenali rumput. Penelitian menghasilkan model memberikan akurasi, presisi, *recall*, dan *F1-Score* sebesar 52.5%, 50%, 7.7%, dan 71.6% pada 10.000 *epochs*, 67.8% akurasi, 67% presisi, 52.4% *recall*, dan 85.9% *F1-Score* pada 20.000 *epochs*, 97.2% akurasi, 96.2% presisi, 97.5% *recall*, dan 99% *F1-Score* pada 100.000 *epochs*, 98.3% akurasi, 98.1% presisi, 99.1% *recall*, dan 99.4% *F1-Score* pada 200.000 *epochs*, terakhir 97% akurasi, 95% presisi, 99% *recall*, dan 99% *F1-Score* pada 242.000 *epochs*. Berdasarkan dari hasil pengamatan didapatkan bahwa performa meningkat dengan semakin banyak *epochs* yang digunakan, namun menjadi jenuh pada 242.000 *epochs*.

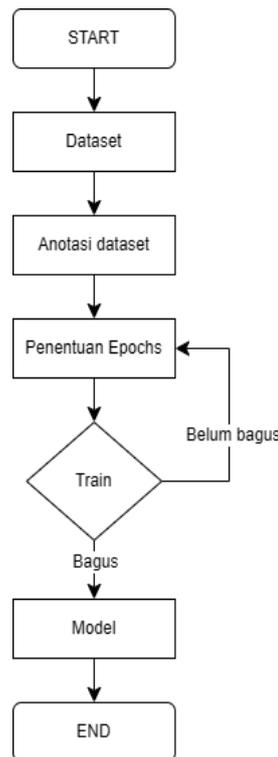
Penelitian terakhir dilakukan oleh (Ogawa et al., 2023) mengenai pengenalan tempat parkir yang kosong menggunakan *YOLOv5*. Tujuan dari penelitian tersebut adalah untuk menentukan kemampuan dari algoritma *YOLOv5* dalam mendeteksi tempat parkir kosong, serta dampak dari *training epochs* terhadap akurasi dari model. *YOLO* dipilih dikarenakan memiliki kecepatan dan akurasi deteksi yang baik. *Dataset* yang digunakan berisikan 135 citra yang telah di anotasikan. Penelitian dijalankan dengan melatih model terhadap 50 dan 300 *epochs*, dengan model yang menggunakan 300 *epochs* menghasilkan hasil yang lebih bagus dari pada 50 *epochs*. Pelatihan 300 *epochs* menghasilkan rata-rata akurasi sebesar 6.5%, sedangkan model yang menggunakan 50 *epochs* menghasilkan akurasi sebesar 41.3%. Pengamatan menghasilkan bahwa jumlah *epochs* mempengaruhi akurasi dari model *YOLOv5*.

Tinjauan pustaka berisi kajian literatur terdahulu yang dilakukan oleh peneliti lain yang relevan dengan penelitian yang dilakukan untuk menunjukkan kebaruan ilmiah artikel tersebut. Pada bagian ini diuraikan *research gap* atau *state of the art* sebagai dasar pernyataan kebaruan ilmiah dari artikel.

### METODE

#### Tahapan Penelitian

Tahapan penelitian memberikan gambaran mengenai tahapan penelitian yang dijalankan seperti yang dapat dilihat pada Gambar 1.



Gambar 1. Tahapan penelitian

**Dataset**

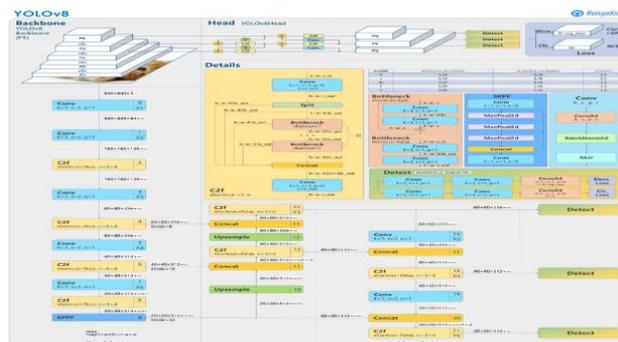
Dataset yang digunakan dalam penelitian ini dapat dilihat pada link *kaggle*: <https://www.kaggle.com/datasets/jasenjonathan/personal-baby-catfish-image-dataset>. Dataset terdiri dari 753 citra bibit ikan lele, di mana terdapat 627 citra yang telah di *crop* dan 126 citra yang tidak di *crop*. Citra diambil dengan menggunakan kamera *smartphone* Huawei Nova 5T pada pencahayaan yang terang dengan menggunakan ember plastik.

**Anotasi Dataset**

Akan dilakukan tahap *preprocessing* terhadap seluruh citra yang ada di dalam *dataset*, setiap citra akan di anotasikan secara manual dengan menggunakan *website civit.ai*. Anotasi data dilakukan untuk memudahkan *machine learning* dalam menentukan dan mengekstraksi informasi dengan menempatkan *bounding box*, segmentasi, atau *key points* pada objek yang di targetkan (S.S. Killikatt et al., 2023).

**YOLOv8**

*You Only Look Once (YOLO) v8* dibuat dan diterbitkan pada bulan Januari 2023 oleh *Ultralytics*, developer *YOLOv5*. Dalam *YOLOv8* bagian *bottleneck* pada arsitektur mirip seperti *YOLOv5*, namun *kernel* konvolusi pertama telah di naikan dari 1x1 menjadi 3x3, dan blok *C3* utama pada *backbone YOLOv5* telah di ganti dengan *C2f* blok dan jumlah *bottleneck* dalam modul *C2f* di ganti dari 3-6-9-3 menjadi 3-6-6-3, seperti yang dapat dilihat pada Gambar 2.



Gambar 2. Arsitektur YOLOv8

Arsitektur dari YOLOv8 dibagi menjadi *backbone*, *neck*, dan *head*. YOLOv8 *backbone* menggunakan CSPDarknet53 *backbone network* yang di modifikasi, yang menggunakan *cross-stage partial connections* untuk meningkatkan alur informasi antara layer dan menaikkan akurasi. *Neck*, atau yang dikenal sebagai ekstraksi fitur, menggabungkan *feature map* dari berbagai tahapan *backbone* untuk menangkap informasi dalam berbagai skala. Terakhir adalah *head*, yang bertanggung jawab atas melakukan prediksi. YOLOv8 menggunakan beberapa modul pendeteksian yang memprediksikan *bounding box*, *objectness scores*, dan *class probabilities* untuk setiap sel dalam *feature map*. Kemudian hasil prediksi tersebut akan dikumpulkan untuk mendapatkan hasil deteksi akhir (Li et al., 2023). Cara kerja dari YOLOv8 dapat disimpulkan menjadi 3 tahap. Pertama, citra *input* di bagi menjadi jaringan atau yang sering disebut sebagai *grids*. Kemudian, YOLOv8 akan memprediksi *bounding box*, *confidence*, dan *class probability* melalui rangkaian layer yang terhubung. Terakhir, dengan menggunakan algoritma *non-max supression* menghilangkan *bounding box* yang saling bertabrakan (Parico & Ahamed, 2021).

**Penentuan Epochs**

Penentuan *epochs* di lakukan untuk menentukan batas awal dan batas akhir dari *epochs* yang akan digunakan untuk melatih model YOLOv8 berdasarkan dari *dataset* yang dipakai. Pada penelitian ini pelatihan akan di mulai dari 50 *epochs* dan jumlah *epochs* akan di naikkan sebanyak 50 *epochs* hingga mencapai 300 *epochs*. Hasil dari setiap pelatihan akan ditampilkan menggunakan *chart* untuk menentukan *epoch* terbaik untuk model YOLOv8.

**HASIL**

Hasil dari penelitian ini adalah *chart* yang berisikan *precision* dan *mAP@50* yang didapatkan dari hasil pelatihan sebagai penanda kualitas dari model. Penentuan *epoch* terbaik dapat dilihat dari kestabilan *graph* setiap 50 *epochs* hingga 300 *epochs* serta nilai tertinggi dari *precision* dan *mAP@50*.

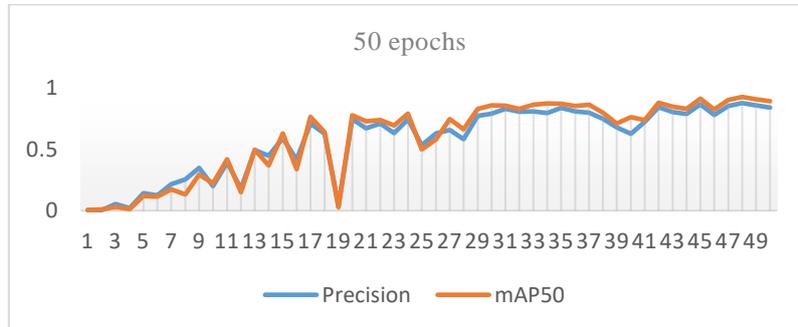
**Pelatihan**

Pelatihan dilakukan dengan menggunakan *Python language* yang di eksekusikan dengan menggunakan perangkat lunak *PyCharm*. Pada penelitian ini akan digunakan model YOLOv8 *nano yaml file* sebagai model awal dan akan di eksekusi menggunakan *CPU Ryzen 5 4600H*.

**Hasil**

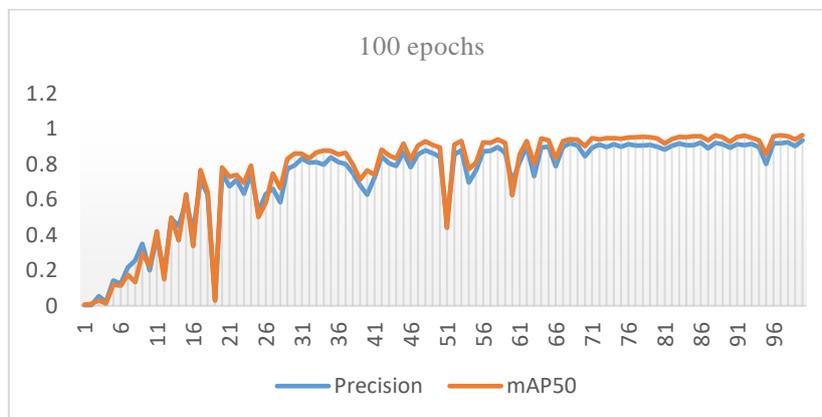
Berikut adalah *graphic* hasil dari penelitian model YOLOv8:

- a. 50 epochs
- b.



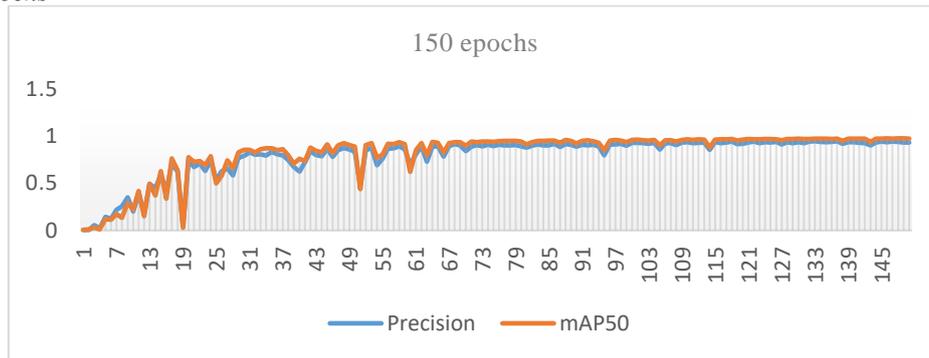
Gambar 3. Training 50 epochs

- c. 100 epochs
- d.



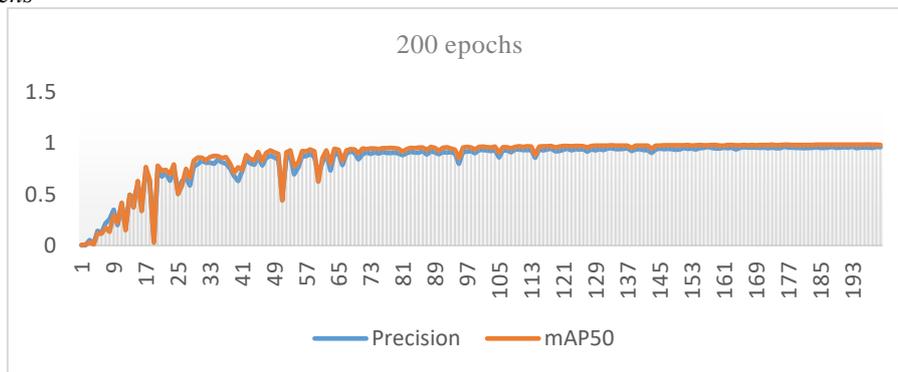
Gambar 4. Training 100 epochs

e. 150 epochs



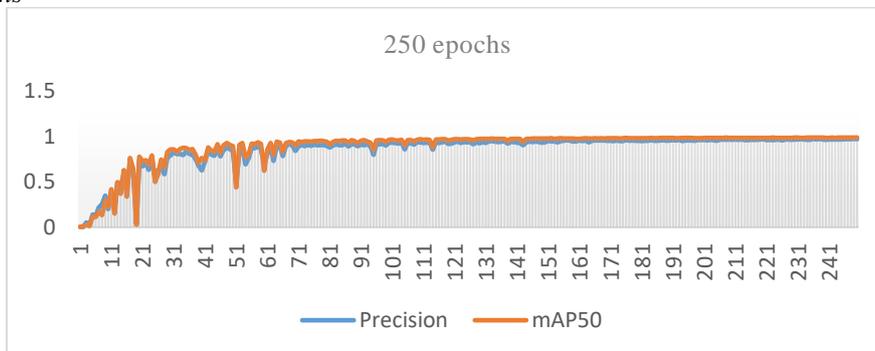
Gambar 5. Training 150 epochs

f. 200 epochs



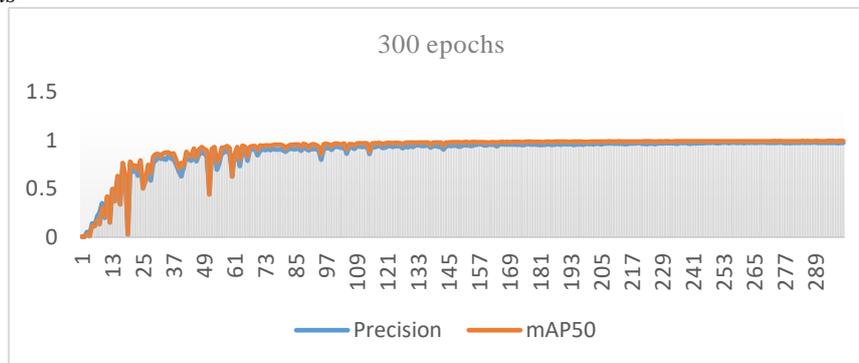
Gambar 6. Training 200 epochs

g. 250 epochs



Gambar 7. Training 250 epochs

h. 300 epochs



Gambar 8. Training 300 epochs.

## PEMBAHASAN

Berdasarkan dari penelitian yang telah dilakukan, didapatkan hasil:

**Waktu pelatihan**

Waktu pelatihan yang digunakan untuk melatih model dapat dilihat pada Tabel 1.

Tabel 1. Tabel waktu pelatihan *epochs*

No	<i>Epochs</i>	Waktu
1.	50	6 jam 52 menit
2.	100	14 jam 1 menit
3.	150	20 jam 59 menit
4.	200	27 jam 39 menit
5.	250	35 jam 51 menit
6.	300	42 jam 7 menit

**Hasil pelatihan**

Jumlah *epochs* yang digunakan dalam penelitian tersaji pada *x-axis* dalam *graphic* dan *value* dari *precision* dan *mAP@50* dapat dilihat pada *y-axis* dalam *graphic*. Penentuan hasil *epoch* terbaik dijelaskan sebagai berikut:

- Gambar 3 menampilkan *graph* hasil pelatihan dari 50 *epochs*. *Graph* menunjukkan bahwa *precision* dan *mAP@50* meningkat yang menandakan bahwa pelatihan berjalan dengan baik dan *losses* menurun dari *epoch* ke-1 sampai dengan *epoch* ke-18. Pelatihan setelah *epoch* ke-20 menunjukkan bahwa peningkatan dari *precision* dan *mAP@50* mulai melambat dengan beberapa penurunan *precision* dan *mAP@50* yang cukup besar yang menandakan bahwa *YOLO* masih belum bisa memprediksi secara akurat,
- Gambar 4 dan Gambar 5 menampilkan *graph* hasil pelatihan dari 100 *epochs*. *Graph* menunjukkan hasil yang mirip dengan *graph* 50 *epochs* di mana model menunjukkan kenaikan drastis pada *epochs* 1-50 yang menunjukkan bahwa *YOLO* masih berada pada tahap mempelajari fitur dari objek. Sebagian besar dari *precision* dan *mAP@50* terlihat merata pada *epoch* 70-95 yang menunjukkan bahwa akurasi *YOLO* telah meningkat,
- Gambar 5 sampai Gambar 8 menampilkan *graph* hasil pelatihan dari 150 *epochs*, 200 *epochs*, 250 *epochs*, dan 300 *epochs*. Pada *graphs* menampilkan kenaikan drastis dari *precision* dan *mAP@50* yang menandakan perkembangan pada model pada *epoch* ke-1 sampai *epoch* ke-50, namun perkembangan mulai melambat di atas *epoch* ke-50 dengan beberapa *training loss* yang terjadi pada *epoch* ke-50 hingga *epoch* ke-70. Pada *epoch* ke-70 hingga *epoch* ke-145 dapat dilihat bahwa sebagian besar data menjadi konstan dengan *value* dari *precision* dan *mAP@50* mulai mendekati 1, hal tersebut menandakan bahwa tingkat akurasi dari arsitektur telah meningkat. Dari *epoch* ke-200 dapat dilihat bahwa *training curve* telah mendatar dengan *precision* yang mendekati *mAP@50* yang menandakan bahwa tidak ada peningkatan lagi dari pelatihan.

Berdasarkan dari hasil yang telah diperoleh, *epochs* 50-150 masih memiliki kecenderungan hasil yang berubah-ubah, sehingga belum cukup baik untuk digunakan sebagai model, akan tetapi untuk penggunaan *epochs* 200-300 memiliki hasil yang cukup baik sehingga baik digunakan untuk model penelitian yaitu rata-rata *precision* sebesar 96%.

## KESIMPULAN DAN SARAN

Terdapat beberapa kesimpulan yang dihasilkan dari penelitian yang telah dilakukan berkaitan dengan kapan penentuan *epochs* terbaik untuk digunakan dalam sebuah model. Pertama, pelatihan terhadap 50 *epochs* menggunakan waktu yang tercepat dibandingkan dengan pelatihan terhadap 300 *epochs*, akan tetapi pelatihan model dengan menggunakan 50 *epochs* menghasilkan hasil yang kurang bagus dibandingkan dengan jumlah *epochs* yang melebihi dari 50. Kedua, Hasil pelatihan yang diperoleh untuk menentukan *epoch* terbaik untuk model yang digunakan yaitu antara 200-300 *epochs*, di mana nilai rata-rata *precision* adalah 0.969 dan *mAP@50* sebesar 0.988 tanpa terjadi penurunan yang cukup signifikan. Terdapat saran untuk penelitian ke depan yakni, dengan menggunakan metode lain seperti dengan menggunakan *GPU* untuk mengurangi waktu yang diperlukan untuk melatih model dari *YOLOv8*, atau dengan menggunakan algoritma lain untuk membandingkan perbedaan optimisasi pelatihan *epochs*.

## REFERENSI

- Afaq, S., & Rao, S. (2020). Significance Of Epochs On Training A Neural Network. *International Journal of Scientific & Technology Research*, 9, 485–488. <https://api.semanticscholar.org/CorpusID:225647672>
- Ajayi, O. G., & Ashi, J. (2023). Effect of varying training epochs of a Faster Region-Based Convolutional Neural Network on the Accuracy of an Automatic Weed Classification Scheme. *Smart Agricultural Technology*, 3,

100128. <https://doi.org/https://doi.org/10.1016/j.atech.2022.100128>
- Ajayi, O. G., Ashi, J., & Guda, B. (2023). Performance evaluation of YOLO v5 model for automatic crop and weed classification on UAV images. *Smart Agricultural Technology*, 5, 100231. <https://doi.org/https://doi.org/10.1016/j.atech.2023.100231>
- Bhavana, N., Kodabagi, M., B, M. K., Ajay, P., Kumaran, Dr. N. M., & Appathurai, A. (2024). POT-YOLO: Real-Time Road Potholes Detection using Edge Segmentation based Yolo V8 Network. *IEEE Sensors Journal, PP*, 1. <https://doi.org/10.1109/JSEN.2024.3399008>
- Chauhan, V., Zhou, J., Lu, P., Molaei, S., & Clifton, D. (2024). A brief review of hypernetworks in deep learning. *Artificial Intelligence Review*, 57. <https://doi.org/10.1007/s10462-024-10862-8>
- Hermawan, I., Agustin, M., Rosyida Zain, A., Tri Mulyani, M., & Nathanael, D. (2023). *Rice Seedling Image Classification Using Light Convolutional Neural Network* (Vol. 9, Issue 2). <https://doi.org/https://doi.org/10.23917/khif.v9i2.20401>
- Khan, A. I., & Al-Habsi, S. (2020). Machine Learning in Computer Vision. *Procedia Computer Science*, 167, 1444–1451. <https://doi.org/https://doi.org/10.1016/j.procs.2020.03.355>
- Li, P., Zheng, J., Li, P., Long, H., Li, M., & Gao, L. (2023). Tomato Maturity Detection and Counting Model Based on MHSA-YOLOv8. *Sensors*, 23, 6701. <https://doi.org/10.3390/s23156701>
- Ogawa, M., Arnon, T., & Gruber, E. (2023). Identifying Parking Lot Occupancy with YOLOv5. *Journal of Student Research*, 12. <https://doi.org/10.47611/jsr.v12i4.2280>
- Parico, A. I., & Ahamed, T. (2021). Real Time Pear Fruit Detection and Counting Using YOLOv4 Models and Deep SORT. *Sensors*, 21, 4803. <https://doi.org/10.3390/s21144803>
- Peryanto, A., Yudhana, A., & Umar, R. (2022). *Convolutional Neural Network and Support Vector Machine in Classification of Flower Images*. <https://doi.org/https://doi.org/10.23917/khif.v8i1.15531>
- Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Computer Science*, 2(6), 420. <https://doi.org/10.1007/s42979-021-00815-1>
- Sharma, S. (2024). Supervised Learning: An InDepth Analysis. *Interantional Journal Of Scientific Research In Engineering And Management*, 08, 1–5. <https://doi.org/10.55041/IJSREM35414>
- S.S. Killikatt, Avantika Patil, Sarika Pharakate, Supriya Koli, Shreya Wankhade, & Maithili Patil. (2023). Object Detection and Image Annotation Using Deep Learning. *International Research Journal of Modernization in Engineering Technology and Science*. <https://doi.org/10.56726/irjmets41723>
- Zhao, X., Wang, L., Zhang, Y., Han, X., Deveci, M., & Parmar, M. (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, 57(4), 99. <https://doi.org/10.1007/s10462-024-10721-6>