

IMPLEMENTASI YOU ONLY LOOK ONCE v8 DALAM DETEKSI MAKANAN WARUNG TEGAL UNTUK SISTEM PERHITUNGAN HARGA OTOMATIS

Niko Purnomo¹, Windu Gata², Muhammad Romadhona Kusuma³,
Riadi Marta Dinata⁴, Modesta Binti Husna⁵

^{1,2,5} Teknologi Informasi, Ilmu Komputer, Universitas Nusa Mandiri, Jakarta, Indonesia

³ Teknologi Informatika, Informatika, Universitas Nusa Mandiri, Jakarta, Indonesia

Jl. Margonda Raya No. 545, Depok

⁴ Teknik Informatika, Institut Sains dan Teknologi Nasional Jakarta
nicopurnomo07@gmail.com

ABSTRAK

Warung Tegal (Warteg) merupakan usaha kuliner yang populer di Indonesia, tetapi sistem perhitungan harga makanannya masih manual, yang dapat menyebabkan kesalahan transaksi. Penelitian ini bertujuan mengembangkan sistem deteksi makanan otomatis menggunakan YOLO v8 untuk mengotomatisasi perhitungan harga. Dataset terdiri dari berbagai lauk warteg yang diproses dengan teknik augmentasi seperti pemotongan, rotasi, dan pencahayaan guna meningkatkan kinerja model. Hasil penelitian menunjukkan bahwa dalam pengujian terbaik dengan dataset 70:30 (20 epochs, batch size 16, learning rate 0.001), model YOLO v8 mencapai *precision* 0.602, *recall* 0.176, *F1-score* 0.32, *box_loss* 1.756, dan *mAP@0.5* 0.229. Tantangan utama meliputi keterbatasan dataset, kompleksitas latar belakang, dan kurangnya perbandingan dengan dataset publik. Meskipun dalam beberapa kondisi akurasi mencapai 75%- 100%, diperlukan dataset lebih besar dan perbandingan model lain untuk meningkatkan akurasi. Sistem ini berpotensi mendukung digitalisasi industri kuliner dan meningkatkan efisiensi transaksi di warteg.

Kata kunci : YOLO v8, Deteksi objek, Pengolahan citra, Warung tegal, Otomatisasi harga

1. PENDAHULUAN

Sejak 1960-an, Jakarta berkembang pesat sebagai kota metropolitan, memicu pertumbuhan warung makan sederhana seperti Warung Tegal (Warteg) [1]. Warteg adalah UMKM kuliner yang menawarkan berbagai menu dengan harga terjangkau, namun pembeli sering kesulitan memilih makanan karena keterbatasan visual dan informasi harga [2]. Warteg, yang berasal dari Tegal, kini dikenal sebagai warung sederhana dengan makanan khas Jawa Tengah dan Jawa Timur [3].

Visualisasi data mempermudah analisis dan membantu memahami pola penjualan [4]. Citra sebagai representasi objek dua dimensi memainkan peran penting dalam berbagai bidang, termasuk pengolahan citra [5]. Teknologi pengenalan citra berbasis *deep learning* (DL) dapat membantu identifikasi makanan warteg untuk mengurangi kesalahan perhitungan harga dan meningkatkan efisiensi transaksi [6]. Penelitian ini mengimplementasikan deteksi menu dan harga makanan warteg menggunakan YOLO v8, pengembangan dari *Convolutional Neural Network* (CNN). Berbagai teknik augmentasi data diterapkan untuk meningkatkan akurasi identifikasi lauk [7]. YOLO v8 diharapkan memberikan solusi otomatisasi perhitungan harga makanan dengan presisi tinggi, menghasilkan metrik evaluasi seperti *accuracy*, *precision*, *recall*, *F1-score*, *Box loss*, dan *mean Average Precision* (mAP) yang lebih baik. Dengan otomatisasi ini, pembeli dan penjual dapat mengetahui total harga secara visual dengan lebih mudah.

2. TINJAUAN PUSTAKA

Beberapa penelitian telah mengaplikasikan YOLO dalam berbagai bidang. Muhammad Farrel Golfantara menggunakan YOLOv8 untuk mendeteksi 14 jenis rempah-rempah dengan akurasi 86% [7]. Lusiana Rahma et al. menerapkan YOLOv3 untuk mendeteksi 31 jenis makanan khas Palembang, mencapai akurasi 96% [8]. G.C. Utami et al. menggunakan YOLOv5 untuk estimasi nutrisi makanan khas Indonesia dengan akurasi 98,61% [9].

Yanto et al. mengembangkan sistem deteksi masker berbasis YOLOv8 dengan akurasi hingga 97% dan waktu komputasi 17 milidetik [10]. Klasifikasi citra terbagi menjadi tiga jenis: *Recognition*, *Detection*, dan *Segmentation* [11]. *Object Recognition* adalah teknik *Computer Vision* yang mengajarkan mesin mengenali objek dalam gambar dan video seperti manusia [12]. *Object Detection* berfokus pada identifikasi dan lokasi objek dalam gambar menggunakan kotak pembatas, sering digunakan dalam pengawasan dan diskriminasi gambar [13]. *Object Segmentation* memberikan label kategori pada setiap piksel gambar untuk memastikan objek dengan kategori yang sama memiliki label seragam [11].

Pra-pemrosesan dataset citra merupakan langkah krusial untuk meningkatkan kualitas data sebelum digunakan dalam pelatihan model. Data yang tidak terorganisir dengan baik dapat berdampak pada akurasi dan prediksi yang tidak optimal, sehingga diperlukan teknik *preprocessing* yang tepat untuk meningkatkan generalisasi model [12].

Salah satu tujuan utama *preprocessing* adalah mencegah *overfitting* dan *underfitting*, yang dapat terjadi jika model tidak sesuai dengan data pelatihan atau kurang mampu mengklasifikasikan data baru dengan baik. *Overfitting* dapat dikurangi dengan validasi model dan penambahan citra pada dataset, sementara *underfitting* sering kali terjadi akibat pemilihan model yang kurang tepat [13].

Beberapa teknik *preprocessing* yang umum digunakan dalam klasifikasi citra meliputi *resizing* untuk memastikan keseragaman dimensi gambar, anotasi untuk menandai objek dalam gambar, serta auto-orientasi guna mempertahankan posisi objek selama augmentasi. Selain itu, augmentasi data digunakan untuk meningkatkan jumlah dan variasi data pelatihan melalui berbagai transformasi, seperti rotasi, *flipping*, *cropping*, *zoom*, translasi acak, dan perubahan kontras. Teknik - teknik ini membantu model *deep learning* lebih adaptif terhadap variasi pencahayaan, orientasi, dan skala objek dalam citra [14].

Dengan menerapkan *preprocessing* dan augmentasi data yang optimal, dataset dapat diperluas tanpa perlu mengumpulkan lebih banyak data nyata, sehingga meningkatkan performa model dalam mengidentifikasi dan mengklasifikasikan objek secara lebih akurat.

Machine learning (ML) memungkinkan komputer belajar tanpa diprogram secara eksplisit [15]. ML meniru kemampuan manusia dalam mengenali pola secara objektif, terutama untuk analisis data besar atau kompleks serta otomatisasi proses analisis [16].

Deep Learning (DL) adalah cabang ML yang menggunakan jaringan saraf tiruan (*Neural Network*) untuk menyelesaikan masalah tertentu. DL terdiri dari beberapa lapisan antara *input* dan *output*, dengan hasil pemrosesan non-linear digunakan untuk pembelajaran fitur dan klasifikasi pola [17].

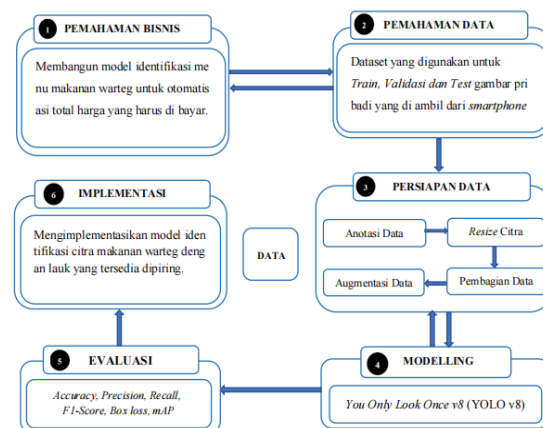
Lapisan tersembunyi memungkinkan transformasi data non-linear menjadi linear, yang menjadi keunggulan utama DL [18].

Algoritma DL diterapkan dalam pengenalan gambar, suara, teks, serta analisis pola dan klasifikasi. Salah satu pengembangan terbaru untuk deteksi objek adalah dengan menggunakan *deep learning* yaitu *You Only Look Once* (YOLO). YOLO adalah algoritma deteksi objek yang menggunakan arsitektur *Convolutional Neural Network* (CNN) yang dirilis pada tahun 2015 oleh Joseph Redmon dan Ali Farhadi. CNN adalah jenis jaringan saraf tiruan yang sangat efektif dalam menangani data gambar, di mana setiap lapisan konvolusi berfungsi untuk mengekstraksi fitur-fitur penting dari citra input [19].

3. METODE PENELITIAN

CRISP-DM adalah kerangka kerja dalam *data mining* yang digunakan untuk analisis dan pengembangan model prediksi berbasis *deep learning*. Metode ini membantu dalam deteksi makanan warteg

dengan membimbing proyek melalui enam tahap utama, memastikan proses yang sistematis dan hasil yang bisa diandalkan.



Gambar 1. Diagram Alur.

3.1. Bussiness Understanding (Pemahaman Bisnis)

Penelitian ini bertujuan mengatasi kesulitan dalam mengidentifikasi makanan warteg secara visual untuk mengetahui harga pesanan. Masalah ini dapat membingungkan pembeli dalam menentukan total pembayaran. Oleh karena itu, penelitian ini mengimplementasikan model YOLO v8 untuk identifikasi lauk dan otomatisasi perhitungan harga.

3.2. Data Understanding (Pemahaman Data)

Dalam implementasi YOLO v8, *Data Understanding* sangat penting untuk memastikan akurasi deteksi makanan warteg. Dataset harus mencakup berbagai jenis makanan dengan variasi pencahayaan, sudut pengambilan gambar, serta karakteristik seperti ukuran, bentuk, warna, dan tekstur. Penumpukan lauk dapat mempengaruhi akurasi, sehingga data harus disiapkan dengan cermat. Selain itu, proses pembersihan diperlukan untuk menghindari gangguan seperti keburaman. Jumlah dan variasi data yang memadai juga diperlukan agar model mampu mengenali makanan secara akurat dalam berbagai kondisi.

3.3. Data Preparation (Persiapan Data)

Tahap *Data Preparation* memastikan data siap untuk pelatihan YOLO v8. *Resize* citra dilakukan ke 640×640 piksel tanpa mengubah aspek rasio agar informasi tetap terjaga. Anotasi citra menggunakan *LabelImg* menambahkan *bounding box* dan label pada setiap objek. Data kemudian dibagi menjadi *training set*, *validation set*, dan *test set* untuk pelatihan, penyetasan *hyperparameter*, serta evaluasi model. Terakhir, augmentasi data seperti rotasi, *flipping*, dan perubahan kecerahan diterapkan untuk meningkatkan variasi data, meningkatkan generalisasi model, serta mencegah *overfitting*.

3.4. Modelling (Pemodelan)

Tahap Pemodelan bertujuan mengembangkan model deteksi lauk warteg menggunakan YOLO v8. Model dilatih dengan data berlabel dan diuji dengan data baru untuk mengukur kemampuannya dalam mengenali pola. *Epoch* menentukan jumlah siklus pelatihan, di mana parameter diperbarui melalui *backpropagation*. Pemilihan jumlah *epoch* yang optimal penting untuk mencegah *underfitting* atau *overfitting*, memastikan model bekerja secara efektif.

3.5. Evaluation (Evaluasi)

Tahap evaluasi menilai efektivitas YOLO v8 dalam mendeteksi makanan warteg menggunakan *Precision*, *Recall*, *F1-Score*, *Box Loss*, dan *mAP*. *Precision* mengukur ketepatan prediksi positif, sedangkan *Recall* menilai sejauh mana model mengenali objek yang benar. *F1-Score* menyeimbangkan *Precision* dan *Recall* untuk deteksi yang akurat. *Box Loss* menggunakan *Intersection over Union* (IoU) untuk mengukur kesalahan *bounding box*, dengan nilai rendah menunjukkan prediksi lebih baik. *mAP* (*mean Average Precision*) menilai akurasi model dalam mendeteksi berbagai kategori makanan secara konsisten.

3.6. Deployment (Implementasi)

Tahap Deployment bertujuan mengidentifikasi citra makanan warteg dan menampilkan harga secara otomatis. Tantangan utama meliputi rendahnya kontras, pencahayaan, dan tumpang tindih lauk, yang menyebabkan model YOLO v8 kesulitan membedakan dan mengidentifikasi makanan dengan akurasi tinggi.

4. HASIL DAN PEMBAHASAN

Dalam penelitian ini, penulis menggunakan algoritma YOLO V8 untuk mengidentifikasi dan mengklasifikasikan menu makanan warteg. Tujuannya adalah mengembangkan model deteksi yang efektif. Proses pemodelan meliputi penggunaan *bounding box*, *pre-processing*, augmentasi, serta pembagian data untuk pelatihan dan validasi. Akurasi diperoleh dari data validasi, dan pemodelan dilakukan di *Visual Studio Code*. Dengan detail pembahasan sebagai berikut:

4.1. Business Understanding

Pada tahap Business Understanding bertujuan mengembangkan sistem klasifikasi citra makanan warteg menggunakan YOLO v8 untuk otomatisasi perhitungan harga berdasarkan gambar pesanan pelanggan. Sistem ini diharapkan dapat meningkatkan efisiensi operasional, mengurangi kesalahan manusia, dan mempercepat layanan tanpa intervensi manual.

4.2. Data Understanding

Tahap *Data Understanding* menggunakan foto menu makanan warteg yang diambil dengan smartphone. Dataset terdiri dari 400 gambar awal dan

dikumpulkan selama dua bulan. Untuk meningkatkan variasi, augmentasi dilakukan hingga total 1.680 gambar. Dataset dibagi menjadi data latih (1.680 gambar), validasi (120 gambar), dan pengujian (20 gambar). Semua gambar berukuran 640×640 piksel dalam format JPG.

Tabel 1. Dataset Makanan Warteg

No	Sumber Data	Jenis Data	Jumlah Data	Ukuran Citra	Format Data
1	Data diambil dari Smartphone	Data Latih	400	640 x 640	JPG
2	Python	Data Setelah Augmentasi	1.680	640 x 640	JPG
3	Data diambil dari Smartphone	Data Validasi	120	640 x 640	JPG
4	Data diambil dari Smartphone	Data Testing	20	640 x 640	JPG

4.3. Data Preparation

Tahap *Data Preparation* mencakup *resize* gambar (640×640 piksel), anotasi manual dengan *LabelImg*, pembagian dataset (70% pelatihan, 30% validasi), dan augmentasi data (*rotasi*, *flip*, *noise*, *blur gaussian*). Langkah ini memastikan YOLO v8 dilatih dengan data berkualitas untuk deteksi makanan warteg dalam berbagai kondisi.

4.4. Resize Citra

Resize citra dilakukan untuk memastikan bahwa data yang diolah konsisten, ukuran gambar diubah menjadi 640 x 640 piksel. Dimulai dengan membuka setiap file gambar.JPG dari folder input yang dipilih menggunakan library PIL (*Python Imaging Library*).



Food01.jpg

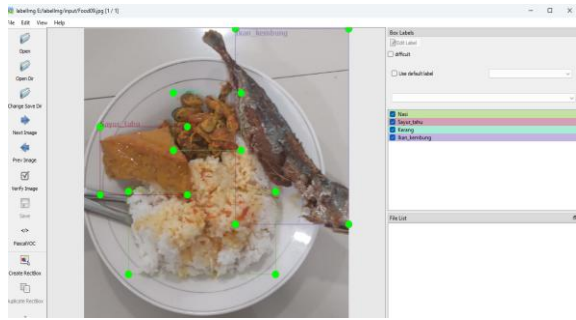


Food02.jpg

Gambar 2. Hasil *Resize* Citra 640x640 Piksel

4.5. Anotasi Citra

Proses anotasi citra dilakukan menggunakan *LabelImg* untuk menandai area makanan warteg secara manual dengan kotak pembatas dan label yang sesuai. Hasil anotasi disimpan dalam format yang kompatibel dengan YOLO v8, seperti XML, yang mencakup informasi tentang nama file, ukuran citra, dan koordinat kotak pembatas. Setiap gambar dalam dataset memiliki file XML terpisah yang membantu dalam deteksi objek. Metode ini memastikan data terstruktur dengan baik, sehingga meningkatkan akurasi model YOLO v8 dalam mengenali makanan warteg.



Gambar 3. Hasil LabelImg Citra

Setelah anotasi dengan LabelImg, hasilnya disimpan dalam format .txt yang sesuai untuk YOLO v8. Setiap gambar memiliki file berisi kelas objek (0 untuk "food"), serta koordinat dan ukuran kotak pembatas dalam format (id_kelas, x_center, y_center, width, height) yang dinormalisasi.

```
<class> <x> <y> <width> <height>
```

Konversi label memastikan model YOLO v8 dapat membaca dan memproses data dengan optimal.

```
0 0.4820833333333333 0.3020746887966805 0.36083333333333334
0.21908713692946058
7 0.4254166666666667 0.6448132780082988 0.4675 0.375103734439834
1 0.775 0.3788381742738589 0.19 0.3809128630705394
2 0.24083333333333334 0.37302904564315353 0.10333333333333333
0.14522821576763487
```

4.6. Pembagian Dataset

Dataset yang telah dianotasi kemudian dibagi menjadi dua kelompok terpisah: data latih (*training*) dan data uji (*testing*). Pembagian ini sangat penting untuk *mengevaluasi* secara objektif kinerja model. Digunakan rasio pembagian 70% *training* dan 30% *testing*. 400 gambar digunakan untuk dataset awal, dengan 280 gambar digunakan sebagai data latih dan 120 gambar untuk data uji.

4.7. Augmentasi Data

Setelah labeling dan pembagian data, augmentasi dilakukan dengan teknik seperti rotasi, *flip*, dan perubahan kecerahan menggunakan *imgaug*, meningkatkan data latih dari 400 menjadi 1.680 gambar. Proses ini membantu model mengenali pola lebih luas, mencegah *underfitting*, dan mengurangi *overfitting*, sehingga meningkatkan akurasi deteksi pada data baru.

Tabel 2. Dataset Makanan Warteg

No	Augmentasi	Hasil
1	Flip	
2	Rotasi	

3	Blur	
4	Noise	

4.8. Modelling

Dalam penelitian ini, YOLO v8 dilatih dengan dataset yang disiapkan untuk mendeteksi dan mengklasifikasikan makanan secara akurat. Proses ini melibatkan *transfer learning* dan *fine-tuning* guna mengoptimalkan parameter, meningkatkan akurasi, dan mempercepat deteksi. Penulis melakukan percobaan dengan mengubah pengaturan YOLO v8 untuk mendapatkan hasil terbaik. Mereka juga menggunakan optimizer Adam untuk menangani data besar dan kompleks. Karena keterbatasan komputer untuk menjaga model stabil, kami mencoba ukuran batch 16, yang umum digunakan, dan ukuran batch 32. Pemilihan ukuran batch ini sangat penting untuk menyeimbangkan kecepatan pelatihan, kualitas model, dan penggunaan sumber daya. Setelah pelatihan, kinerja model diuji dengan beberapa metrik, dengan kedua ukuran batch. Hasilnya disajikan dalam Tabel 3.

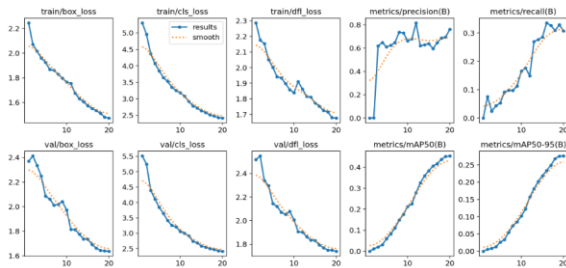
Tabel 1. Perbandingan Kinerja Model YOLO v8

Epo ch	Batch Size	Ir	Box_ loss	mAP @0.5	Precision	Rec all	F1-Score
5	16	0.001	2.060	0.013	0.355	0.030	0.02
10	16	0.001	1.922	0.086	0.611	0.074	0.11
15	16	0.001	1.825	0.152	0.577	0.121	0.20
20	16	0.001	1.756	0.229	0.602	0.176	0.32
5	16	0.01	3.126	0.000	0.080	0.001	0.00
10	16	0.01	2.809	0.000	0.140	0.005	0.00
15	16	0.01	2.831	0.008	0.053	0.043	0.01
20	16	0.01	2.793	0.012	0.081	0.061	0.02
5	32	0.001	3.126	0.000	0.080	0.001	0.00
10	32	0.001	2.811	0.001	0.080	0.006	0.00
15	32	0.001	2.835	0.005	0.072	0.033	0.01
20	32	0.001	2.798	0.013	0.061	0.071	0.02
5	32	0.01	3.126	0.000	0.080	0.001	0.00
10	32	0.01	2.809	0.002	0.047	0.012	0.00
15	32	0.01	2.834	0.004	0.091	0.030	0.01
20	32	0.01	2.796	0.013	0.071	0.062	0.01

Penelitian ini menemukan model YOLO v8 terbaik yang memiliki 20 *epochs*, *batch size* 16, dan *learning rate* 0.001. Meskipun ada ruang untuk *perbaikan*, model ini menunjukkan deteksi lauk makanan yang baik dengan *mAP@0.5* 0.229, *precision* 0.602, dan *recall* 0.176. Dengan *F1-score* 0.32, keseimbangan antara akurasi dan *recall* ditunjukkan, dan kehilangan *box loss* 1.756 menunjukkan akurasi dalam menentukan posisi lauk. Model ini memiliki akurasi yang signifikan dengan *batch size* 16 dan waktu pelatihan 6.427 jam.

Model YOLO v8 terpilih untuk deteksi lauk warteg *dievaluasi* setelah eksperimen dan pemilihan konfigurasi optimal dengan 20 *epochs*, *batch size* 16, dan *learning rate* 0.001. Analisis mencakup *kurva*

loss, metrik performa, dan stabilitas model selama pelatihan dan validasi.

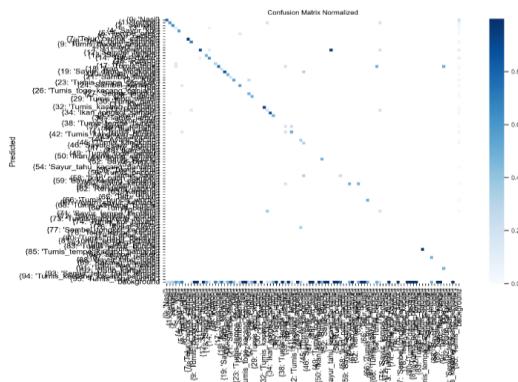


Gambar 3. Matrix Evaluasi Pelatihan YOLO v8

Gambar 3 menunjukkan metrik evaluasi pelatihan model YOLO v8 menggunakan optimizer Adam. Loss pada pelatihan dan validasi menurun secara konsisten, sementara *precision*, *recall*, dan *mAP* meningkat seiring bertambahnya *epoch*. Hal ini menandakan bahwa optimizer Adam berhasil mengoptimalkan model, menjadikannya pilihan yang tepat untuk penelitian ini.

4.9. Evaluation

Setelah model dilatih, tahap evaluasi dilakukan untuk menilai kinerjanya dalam mendeteksi makanan warteg. *Confusion matrix* digunakan untuk mengevaluasi performa klasifikasi dan mengidentifikasi kesalahan prediksi. Sebagai tabel matriks, *confusion matrix* menggambarkan kinerja model pada dataset uji, memberikan informasi perbandingan klasifikasi, serta dikenal sebagai *error matrix*. Matriks ini menjadi alat yang berguna dan komprehensif dalam menilai efektivitas model klasifikasi.



Gambar 4. Confusion Matrix

Hasil *confusion matrix* menunjukkan bahwa model mampu mengklasifikasikan sebagian besar makanan warteg dengan baik, meskipun masih terdapat beberapa kesalahan prediksi pada kelas yang memiliki kemiripan visual. Distribusi nilai pada matriks mengindikasikan bahwa model memiliki akurasi yang cukup baik, namun masih dapat ditingkatkan melalui optimasi lebih lanjut. Pengukuran kinerja *confusion matrix* menggunakan 4 istilah yang akan menunjukkan hasil dari proses klasifikasi yaitu:

- TP (True Positive): Jumlah kasus positif yang diprediksi benar oleh model.
- TN (True Negative): Jumlah kasus negatif yang diprediksi benar oleh model.
- FP (False Positive): Jumlah kasus negatif yang diprediksi salah sebagai positif oleh model.
- FN (False Negative): Jumlah kasus positif yang diprediksi salah sebagai negatif oleh model.

a. Accuracy

Accuracy untuk mengukur proporsi prediksi yang benar dari keseluruhan prediksi yang dibuat oleh model. *Accuracy* ini adalah metrik evaluasi yang paling dasar.

Rumus *Accuracy*:

$$Accuracy = \frac{TP + TN}{TN + TP + FN + FP} \quad (1)$$

b. Precision

Precision adalah metrik evaluasi untuk mengukur ketepatan prediksi positif dalam klasifikasi biner. *Precision* tinggi menunjukkan bahwa prediksi positif model memiliki probabilitas tinggi untuk benar. Metrik ini sangat penting dalam kasus dengan biaya *False Positive* (FP) yang tinggi.

Rumus *Precision*:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

c. Recall

Recall adalah matrik evaluasi yang digunakan dalam klasifikasi biner untuk mengukur kemampuan model dalam mengidentifikasi semua kasus positif yang ada.

Rumus *Recall*:

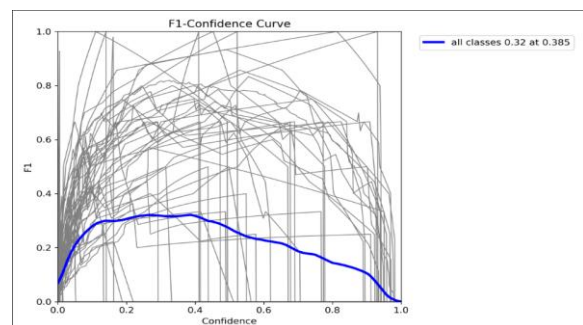
$$Recall = \frac{TP}{TP + FN} \quad (3)$$

d. F1-score

F1-score adalah matriks evaluasi yang digunakan untuk mengukur keseimbangan antara *precision* dan *recall* dalam klasifikasi. *F1-score* adalah rata-rata harmonis dari *precision* dan *recall*, memberikan satu angka yang mencerminkan *trade-off* antara kedua metrik tersebut. *F1-score* sangat berguna ketika perlu mempertimbangkan keseimbangan antar *false positive* dan *false negative*.

Rumus *F1-score*:

$$F1 - Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (4)$$

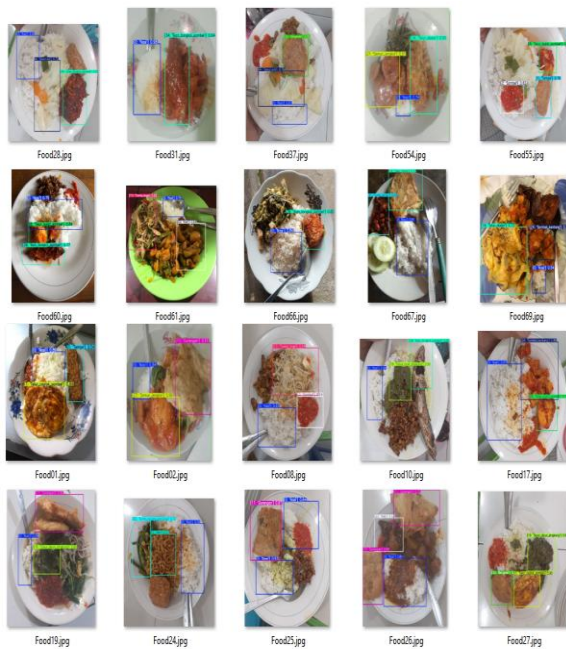


Gambar 5. F1-Confidence Curve

F1-Confidence Curve menunjukkan bahwa nilai *F1-score* tertinggi yang dicapai adalah 0.32 pada *confidence threshold* 0.385. Grafik ini mengindikasikan bahwa model masih memiliki keterbatasan dalam menyeimbangkan *presisi* dan *recall*, sehingga dapat dilakukan perbaikan dengan optimasi *threshold* atau peningkatan kualitas data latih.

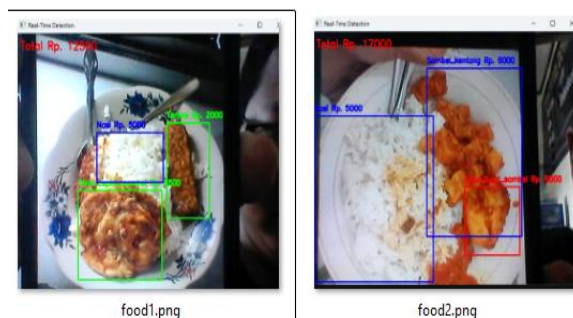
4.10. Implementasi

Model yang dikembangkan mampu mengidentifikasi lauk di piring menggunakan YOLO v8. Sistem ini diharapkan dapat mengurangi kesalahan transaksi, mempercepat pencatatan, dan meningkatkan efisiensi operasional warteg.



Gambar 6. Hasil Prediksi Model

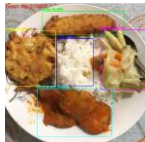
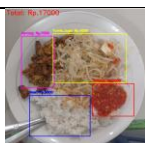
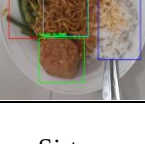
Hasil pengujian model menunjukkan bahwa dengan menambahkan 20 dataset acak dengan variasi latar belakang dan bentuk, model masih mengalami kesulitan dalam mengklasifikasikan beberapa gambar. Hal ini mengindikasikan bahwa model perlu ditingkatkan untuk menangani variasi yang lebih kompleks.



Gambar 7. Hasil Prediksi Real-time

Hasil pengujian menunjukkan bahwa sistem mampu mendeteksi dan mengidentifikasi makanan warteg secara *real-time* dengan akurasi yang baik. Sistem berhasil mengenali berbagai jenis makanan pada piring serta menghitung harga total secara otomatis. Deteksi makanan ditampilkan dengan *bounding box* berwarna dan label harga pada gambar. Implementasi ini membuktikan bahwa sistem dapat membantu mempercepat proses transaksi dan meningkatkan efisiensi di warteg.

Tabel 4. Pengujian berbasis website

Gambar Hasil Prediksi	Teks Hasil Prediksi	Akurasi (%)
	Nasi Telur ceplok sambel Tempe	75.00
	Nasi Gorengan Tempe Sayur kol Semur_jengkol	100.00
	Nasi Kerang Tumis_toge Sambel	100.00
	Nasi Sambel_ kentang Telur bulat sambel	75.00
	Nasi Tumis kacang panjang Mie Tempe	100.00

Sistem deteksi makanan warteg berbasis YOLOv8 berhasil di-deploy dengan fitur utama "Capture Food Real-time" untuk identifikasi dan perhitungan harga otomatis. Sistem menampilkan *bounding box* dengan nama makanan dan harga, serta menyimpan gambar untuk pelatihan ulang. Berdasarkan tabel hasil pengujian yang diberikan, terdapat hasil deteksi akurat terhadap menu makanan dengan persentase akurasi yang tertera di kolom terakhir. Berikut adalah penjelasan untuk setiap baris hasil pengujian:

- Nasi Telur Ceplok Sambel Tempe (Harga Rp. 12500, Akurasi: 75%) Pengujian menunjukkan bahwa nasi telur ceplok sambel tempe dapat ditemukan dengan akurasi 75%. Hasil ini menunjukkan bahwa sistem dapat mengenali dengan baik bagian utama hidangan, tetapi

- kemungkinan kesalahan pada bahan pelengkap atau penyajian sangat kecil.
- Nasi Gorengan Tempe Sayur Kol Sambel Jengkol (Harga Rp. 21.000, Akurasi: 100%) Deteksi pada nasi gorengan tempe sayur kol sambel jengkol menunjukkan akurasi sempurna sebesar 100%. Ini menunjukkan bahwa sistem mengenali dengan sangat baik semua bagian menu, termasuk nasi goreng, tempe, sayur kol, sambel, dan jengkol, secara akurat.
 - Nasi Kerang Tumis Toge Sambel (Harga: Rp.17.000, Akurasi: 100%) Hasil pengujian untuk nasi kerang tumis toge sambel juga menunjukkan akurasi seratus persen. Hasil ini menunjukkan bahwa sistem dapat dengan akurat mengidentifikasi semua komponen hidangan, termasuk nasi, kerang, toge, dan sambel.
 - Nasi Sambel Kentang dan Telur Bulat Sambel (Harga: Rp.17.000, Akurasi: 75%) Deteksi terhadap nasi sambel kentang dan telur bulat sambel memiliki akurasi 75%, yang menunjukkan hasil yang cukup baik. Namun, mungkin ada kesalahan yang terjadi pada komponen sambel atau bagian lainnya.
 - Nasi Tumis Kacang Panjang Mie Tempe (Harga: Rp.15.000, Akurasi: 100%) Uji coba terakhir menunjukkan bahwa nasi tumis kacang panjang mie tempe memiliki akurasi seratus persen. Ini menunjukkan deteksi yang sangat akurat terhadap setiap bagian hidangan, memberikan hasil yang ideal untuk transaksi di warung.

Secara keseluruhan, pengujian menunjukkan bahwa sistem dapat mengenali sebagian besar menu dengan sangat baik dengan beberapa menu dengan akurasi 100% dan beberapa dengan akurasi 75% yang menunjukkan bahwa sistem tersebut cukup efektif dalam membantu proses transaksi di warung.

Tabel 5. Harga Menu Makanan

No	Menu Makanan	Harga (Rp)	No	Nama Makanan	Harga (Rp)
1	Nasi	5000	59	Ikan Patin Sambel	12500
2	Tempe	2000	60	Sayur Kacang Panjang	4500
3	Sambel	1000	61	Sayur Kentang	5500
4	Mie	3000	62	Ikan Patin Sayur	13000
5	Sayur Kol	3000	63	Rendang Daging	15000
6	Tempe Orek	4000	64	Kentang	4500
7	Telur Ceplok	5000	65	Bihun	5000
8	Telur Ceplok Sambel	5500	66	Telur Bulat	5500
9	Telur Bulat Sambel	6000	67	Tumis Buncis Wortel	12500

No	Menu Makanan	Harga (Rp)	No	Nama Makanan	Harga (Rp)
10	Tumis Oncom Cepokak	4500	68	Ayam Sayur Kentang	5500
11	Sayur Timun	3000	69	Tumis Kentang Buncis	5000
12	Gorengan	2000	70	Tumis Buncis	2500
13	Sambel Tempe	4000	71	Tahu	5500
14	Sayur Bayam	3000	72	Sayur Tempe Kentang	5500
15	Telur Dadar	6000	55	Sayur Tahu Kacang Panjang	5000
16	Rawon	10000	56	Tumis Tahu	4500
17	Kerang	7000	57	Tumis Oncom	5000
18	Tumis Toge	4000	58	Tumis Sawi	4500
19	Ikan Kembung	8000	59	Ikan Patin Sambel	12500
20	Sayur Daun Singkong	3500	60	Sayur Kacang Panjang	4500

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan dataset baru dengan 400 gambar makanan warteg dan menerapkan YOLOv8 untuk deteksi makanan dalam satu piring. Model menunjukkan hasil positif dengan *precision* 0.602, *recall* 0.176, *F1-score* 0.32, *box_loss* 1.756, dan *mAP@0.5* 0.229. Teknik augmentasi data diterapkan untuk meningkatkan akurasi deteksi. Selain itu, sistem otomatisasi perhitungan harga berbasis deteksi objek dikembangkan untuk mengurangi kesalahan transaksi manual di warteg dan mendukung digitalisasi UMKM kuliner. Akurasi model berkisar antara 75% hingga 100% dalam berbagai kondisi. Penelitian ini dapat menjadi referensi dalam pengembangan sistem deteksi makanan berbasis *deep learning*, dengan peluang pengembangan lebih lanjut pada peningkatan dataset dan eksplorasi model arsitektur lain untuk meningkatkan akurasi dan efisiensi sistem.

DAFTAR PUSTAKA

- [1] Rohadi, "Transformasi Visual Fasad Warung Tegal di Jakarta," *Jurnal Senirupa Warna*, vol. 6, no. 2, pp. 141–147, 2021.
- [2] A. H. Sukma and I. Pranawukir, "Perencanaan dan Strategi Komunikasi Jaringan Franchise Warung Tegal Kharisma Bahari," *WACANA: Jurnal Ilmiah Ilmu Komunikasi*, vol. 19, no. 2, pp. 274–284, 2020.
- [3] S. Supiyandi, M. A. Mujib, K. Azis, R. Abdillah, dan S. N. Iskandar, "Penerapan teknologi pengolahan citra dalam analisis data visual pada

- tinjauan komprehensif," *Jurnal Kendali Teknik dan Sains*, vol. 2, no. 3, pp. 179-187, 2024.
- [4] Hutahaean, H. D., B. D. Waluyo, and M. A. Rais, "Teknologi Identifikasi Objek Berbasis Drone Menggunakan Algoritma Sift Citra Digital," *Jurnal Teknik Informatika UNIKA Santo Thomas*, vol. 4, no. 2, pp. 202–207, 2019.
- [5] W. Min, S. Jiang, L. Liu, Y. Rui, and R. Jain, "A survey on food computing," *ACM Computing Surveys (CSUR)*, vol. 52, no. 5, pp. 1–36, 2019.
- [6] C. Kiourt, G. Pavlidis, and S. Markantonatou, "Deep learning approaches in food recognition," in *Machine Learning Paradigms: Advances in Deep Learning-Based Technological Applications*, pp. 83–108, 2020.
- [7] M. F. Golfantara, "Penggunaan Algoritma Yolo V8 Untuk Identifikasi Rempah-Rempah," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3S1, 2024.
- [8] L. Rahma, H. Syaputra, A. H. Mirza, and S. D. Purnamasari, "Objek deteksi makanan khas Palembang menggunakan algoritma YOLO (You Only Look Once)," *Jurnal Nasional Ilmu Komputer*, vol. 2, no. 3, pp. 213-232, 2021.
- [9] D. I. Mulyana, "Optimasi Penerapan Algoritma Yolo dan Data Augmentasi dalam Klasifikasi Pakaian Tradisional Kebaya," *Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi)*, vol. 8, no. 1, pp. 188-193, 2024.
- [10] Y. Yanto, F. Aziz, and I. Irmawati, "YOLO-V8 peningkatan algoritma untuk deteksi pemakaian masker wajah," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 3, pp. 1437–1444, 2023.
- [11] T. I. Z. M. Putra, S. Suprpto, and A. F. Bukhori, "Model Klasifikasi Berbasis Multiclass Classification dengan Kombinasi Indobert Embedding dan Long Short-Term Memory untuk Tweet Berbahasa Indonesia," *Jurnal Ilmu Siber dan Teknologi Digital*, vol. 1, no. 1, pp. 1-28, 2022.
- [12] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, and Y. Miao, "Review of image classification algorithms based on convolutional neural networks," Nov. 01, 2021, *MDPI*. doi: 10.3390/rs13224712.
- [13] Y. Liu, P. Sun, N. Wergeles, and Y. Shang, "A survey and performance evaluation of deep learning methods for small object detection," *Expert Syst Appl*, vol. 172, Jun. 2021, doi: 10.1016/j.eswa.2021.114602.
- [14] R. Yang and Y. Yu, "Artificial Convolutional Neural Network in Object Detection and Semantic Segmentation for Medical Imaging Analysis," Mar. 09, 2021, *Frontiers Media S.A.* doi: 10.3389/fonc.2021.638182.
- [15] S. Hao, Y. Zhou, and Y. Guo, "A Brief Survey on Semantic Segmentation with Deep Learning," *Neurocomputing*, vol. 406, pp. 302–321, Sep. 2020, doi: 10.1016/j.neucom.2019.11.118.
- [16] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.
- [17] C. Bu and Z. Zhang, "Research on Overfitting Problem and Correction in Machine Learning," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Dec. 2020. doi: 10.1088/1742-6596/1693/1/012100.
- [18] S. Hadiani, F. Aziz, D. Nur Sulistyowati, D. Riana, R. Saputra, and Kurniawantoro, "Identification of Potato Plant Pests Using the Convolutional Neural Network VGG16 Method," *Journal Medical Informatics Technology*, pp. 39–44, Jun. 2024, doi: 10.37034/medinftech.v2i2.37.
- [19] L. Rahma, H. Syaputra, A. H. Mirza, and S. D. Purnamasari, "Objek deteksi makanan khas Palembang menggunakan algoritma YOLO (You Only Look Once)," *Jurnal Nasional Ilmu Komputer*, vol. 2, no. 3, pp. 213-232, 2021.