

Pengenalan Citra Makanan dan Perhitungan Kalori Otomatis Berbasis Convolutional Neural Network

Resty Annisa¹, Puput Budi Wintoro², Rio Ariestia Pradipta³, Ramadhana Komala⁴

^{1,2,3}Teknik Informatika, Universitas Lampung, Jl. Prof. Soemantri Brojonegoro, Bandar Lampung 35145

⁴Pendidikan Dokter, Universitas Lampung, Jl. Prof. Soemantri Brojonegoro, Bandar Lampung 35145

Email: ¹resty.annisa@eng.unila.ac.id, ²budi.wintoro@eng.unila.ac.id, ³rio.ariestia@eng.unila.ac.id,
⁴ramadhana.komala@fk.unila.ac.id

Abstrak. Perkembangan teknologi kecerdasan buatan, khususnya di bidang computer vision, telah membuka peluang baru dalam penerapan sistem pengenalan citra untuk mendukung program kesehatan masyarakat. Penelitian ini bertujuan untuk mengembangkan sistem cerdas yang mampu melakukan pengenalan citra makanan dan perhitungan kalori secara otomatis menggunakan algoritma Convolutional Neural Network (CNN). Sistem ini diimplementasikan sebagai bagian dari upaya mendukung Dapur Gizi, dengan tujuan membantu pemantauan asupan gizi secara objektif dan efisien. Metode penelitian mencakup tahapan pengumpulan dataset citra makanan, pra-pemrosesan data (resizing, augmentation, dan normalisasi), pelatihan model CNN untuk klasifikasi jenis makanan, serta integrasi modul perhitungan kalori berdasarkan hasil klasifikasi. Pengujian dilakukan untuk mengevaluasi tingkat akurasi model dalam mengenali citra makanan dan estimasi kalori menggunakan metrik evaluasi seperti accuracy, precision, recall, dan mean absolute error (MAE). Hasil penelitian menunjukkan bahwa model CNN yang dikembangkan mampu mengenali jenis makanan dengan tingkat akurasi sebesar 95,6%, precision 94,8%, dan recall 93,2%, dengan nilai error perhitungan kalori rata-rata (MAE) sebesar 6,7%. Nilai-nilai tersebut menunjukkan bahwa sistem dapat mengenali makanan dan menghitung estimasi kalori dengan tingkat ketepatan yang tinggi. Sistem ini menjadi solusi inovatif dalam mendukung pelaksanaan dan evaluasi program Dapur Gizi, serta berpotensi dikembangkan lebih lanjut sebagai aplikasi pemantauan gizi berbasis digital di bidang kesehatan dan pendidikan.

Kata kunci: Convolutional Neural Network, Kecerdasan Buatan, Pengenalan Citra Makanan, Perhitungan Kalori

Abstract. Recent advancements in artificial intelligence, particularly in the field of computer vision, have enabled the development of automated image recognition systems to support public health initiatives. This study presents the development of an intelligent system for automatic food image recognition and calorie estimation using a Convolutional Neural Network (CNN) architecture. The system is designed to assist the Free Nutritious Meal Program by providing an efficient and objective method for monitoring nutritional intake. The proposed approach involves several stages, including the collection of food image datasets, data preprocessing (resizing, augmentation, and normalization), CNN model training for food classification, and the integration of a calorie estimation module derived from classification results. Model performance was evaluated using standard metrics such as accuracy, precision, recall, and mean absolute error (MAE). Experimental results demonstrate that the developed CNN model achieves an accuracy of 95.6%, precision of 94.8%, and recall of 93.2%, with an average MAE of 6.7% for calorie estimation. These findings indicate that the system can effectively recognize various food types and accurately estimate their caloric content. The proposed system provides an innovative and practical solution for supporting the implementation and evaluation of free nutritious meal programs and can be further extended as a digital nutrition monitoring platform in healthcare and educational environments.

Keywords: Convolutional Neural Network, artificial intelegent, food image recognition, calorie estimation

PENDAHULUAN

Pembangunan sumber daya manusia (SDM) merupakan prioritas utama dalam Rencana Pembangunan Jangka Menengah Nasional (RPJMN) 2025–2029, dengan fokus pada peningkatan kualitas kesehatan dan gizi masyarakat sebagai bagian dari visi Indonesia Emas 2045. Dalam konteks tersebut, pemerintah menetapkan Program Makan Bergizi Gratis (MBG) sebagai salah satu Proyek



Strategis Nasional (PSN) melalui Peraturan Presiden (Perpres) No. 12 Tahun 2025. Program ini memiliki tujuan strategis untuk meningkatkan status gizi peserta didik, memperkuat ketahanan pangan lokal, dan mendukung pemerataan kesejahteraan sosial. Pemerintah berupaya memastikan bahwa setiap peserta didik memperoleh asupan makanan bergizi seimbang setiap hari guna mendukung tumbuh kembang dan konsentrasi belajar di sekolah melalui program MBG.

Pada kenyataannya, pelaksanaan program MBG di berbagai daerah masih menghadapi sejumlah kendala teknis dan administratif, terutama dalam proses pengawasan dan evaluasi gizi. Berdasarkan hasil penelitian Agustini (2025), tantangan dalam implementasi MBG meliputi ketidaksesuaian standar gizi antarwilayah, distribusi makanan yang belum merata, serta lemahnya koordinasi antar pemangku kepentingan [1]. Proses evaluasi yang dilakukan secara manual menggunakan tabel gizi dan pencatatan berbasis kertas juga menimbulkan potensi kesalahan dan inefisiensi, terutama ketika diterapkan pada skala nasional. Akibatnya, pengawasan terhadap kesesuaian asupan gizi peserta menjadi tidak optimal dan belum berbasis data digital yang terintegrasi. Dibutuhkan sistem berbasis teknologi yang mampu mendukung otomatisasi proses pemantauan gizi secara objektif, cepat, dan akurat. Salah satu pendekatan potensial yakni dengan pemanfaatan kecerdasan buatan (*Artificial Intelligence*) di bidang komputer vision menggunakan algoritma *Convolutional Neural Network* (CNN). CNN mampu mengenali pola visual dari citra digital dan mengklasifikasikan objek berdasarkan fitur spasial yang kompleks. Implementasi arsitektur YOLOv11 (You Only Look Once) pada sistem dapat mendeteksi berbagai jenis makanan dalam satu gambar secara real-time serta memperkirakan kandungan kalori dengan efisien. Penerapan teknologi ini sangat relevan dalam konteks MBG, karena memungkinkan pemantauan gizi peserta secara digital dan mendukung evaluasi berbasis data.

Urgensi penelitian ini berlandaskan pada amanat Perpres No. 12 Tahun 2025, yang menekankan bahwa pelaksanaan proyek strategis nasional harus dilakukan secara efisien, transparan, dan berbasis data digital. Oleh karena itu, pengembangan sistem pengawasan gizi berbasis CNN diharapkan dapat mendukung prinsip *good governance* serta memperkuat digitalisasi layanan publik di bidang pendidikan dan kesehatan. Sejalan dengan hal tersebut, pendekatan berbasis deep learning telah terbukti meningkatkan efektivitas pemantauan pola makan masyarakat dan akurasi analisis gizi hingga 30% dibandingkan metode manual [2].

Beberapa penelitian sebelumnya menunjukkan efektivitas pendekatan CNN dalam bidang pengenalan makanan dan perhitungan kalori. Zianka et al. [3] mengembangkan aplikasi deteksi makanan berbasis CNN untuk pencegahan stunting dengan akurasi mencapai 92,3%. Riswanto et al. [4] menerapkan metode Single Shot Multibox Detector (SSD) untuk klasifikasi makanan tradisional Indonesia dengan mean Average Precision (mAP) sebesar 89%. Dalam penelitian internasional, Goel et al. [2] memperkenalkan sistem deteksi makanan otomatis dengan multi-dish recognition yang mampu meningkatkan efisiensi klasifikasi hingga 94%. Sapkota et al. [5] menunjukkan bahwa arsitektur YOLOv11 memiliki performa terbaik dalam mendeteksi objek real-time dengan efisiensi komputasi tinggi, sedangkan penelitian klasik oleh Kawano dan Yanai [6] menunjukkan bahwa estimasi kalori berbasis citra dapat menghasilkan tingkat kesalahan rata-rata di bawah 10%.

Berdasarkan berbagai studi tersebut, penerapan CNN dengan arsitektur YOLOv11 berpotensi mendukung keberhasilan pelaksanaan Dapur Gizi. Sistem ini memungkinkan proses pemantauan dan edukasi gizi dilakukan secara cepat, objektif, dan akurat, sekaligus mendukung pencapaian sasaran RPJMN 2025–2029 dalam peningkatan kualitas gizi masyarakat. Penelitian ini dirancang untuk mengembangkan sistem cerdas yang mampu mengenali citra makanan serta menghitung estimasi kalori secara otomatis menggunakan CNN. Proses pengembangan meliputi pengumpulan dataset citra makanan bergizi, pra-pemrosesan data (resizing, augmentation, dan normalisasi), pelatihan model dengan metode transfer learning, integrasi modul perhitungan kalori berbasis data gizi nasional, serta evaluasi performa menggunakan metrik accuracy, precision, recall, dan mean absolute error (MAE).



METODOLOGI PENELITIAN

Metode Penelitian ini menggunakan pendekatan eksperimental kuantitatif dengan tujuan mengembangkan dan menguji sistem pengenalan citra makanan serta perhitungan kalori otomatis menggunakan algoritma Convolutional Neural Network (CNN). Pendekatan ini dipilih untuk memperoleh model yang mampu mengenali jenis makanan bergizi secara akurat dan memperkirakan jumlah kalornya berdasarkan data visual. Tahapan penelitian terdiri dari pengumpulan dataset, prapemrosesan data, pelatihan model CNN, integrasi perhitungan kalori, dan evaluasi model.



Gambar 1. Alur Penelitian

HASIL DAN PEMBAHASAN

Hasil penelitian meliputi proses pengolahan data, pelatihan model dengan arsitektur YOLOv11, serta integrasi modul perhitungan kalori berbasis Tabel Komposisi Pangan Indonesia (TKPI).

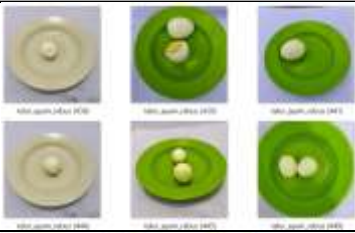
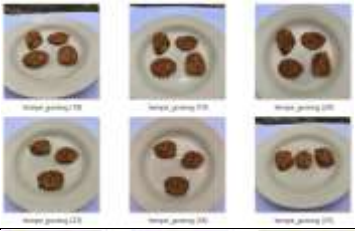
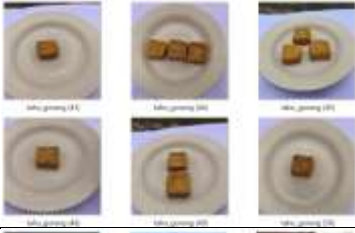
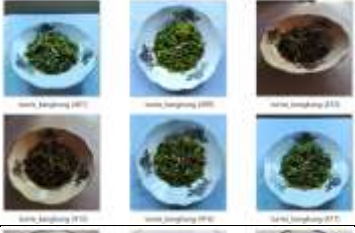
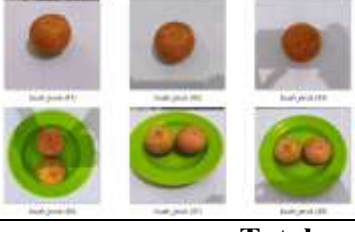
1. Pengumpulan Dataset

a. Data Primer

Data primer diperoleh melalui pengambilan gambar langsung menggunakan kamera ponsel dengan spesifikasi kamera 5 beresolusi 50 MP, aperture f/1.7, focal length 23mm (wide). Tabel 1 menyajikan gambar yang terkumpul sebagai dataset primer masing-masing kategori yaitu nasi putih, jagung rebus, ayam goreng paha bawah, telur ayam rebus, tempe goreng, tahu goreng, tumis kangkung, bayam bening, buah pisang, dan buah jeruk menyumbang 100 gambar ke dalam dataset. Keseluruhan dataset primer yang terkumpul dari tabel ini mencapai 1.000 gambar.

Tabel 1. Jumlah Dataset Primer

No.	Gambar	Jenis	Jumlah
1		Nasi Putih	100
2		Jagung Rebus	100
3		Ayam Goreng Paha Bawah	100

No.	Gambar	Jenis	Jumlah
4		Telur Ayam Rebus	100
5		Tempe Goreng	100
6		Tahu Goreng	100
7		Tumis Kangkung	100
8		Bayam Bening	100
9		Buah Pisang	100
10		Buah Jeruk	100
Total			1000

b. Data Sekunder

Data sekunder dari sumber-sumber online seperti Google, web Kaggle, dan web Roboflow. Setiap jenis makanan, yaitu nasi putih, jagung rebus, ayam goreng paha bawah, telur ayam



rebus, tempe goreng, tahu goreng, tumis kangkung, bayam bening, buah pisang, dan buah jeruk, memiliki jumlah gambar sebanyak 400. Total dataset sekunder yang terkumpul mencapai 4.000 gambar untuk mendukung keakuratan model dalam penelitian.

2. Pra-pemrosesan Data

Tahap ini meliputi beberapa proses yaitu resize data, labeling data, split data, dan augmentasi data.

a. Resizing

Tabel 2 memperlihatkan dataset sebelum dilakukan penyamaan rasio dan setelah dilakukan penyamaan rasio gambar. Keseluruhan data dipastikan memiliki rasio yang sama yaitu 1:1.

Tabel 2. Pemerataan Rasio Gambar

Sebelum			Setelah		
					
ayam	kangkung	jeruk	ayam	kangkung	jeruk
					
kangkung	nasi	pisang	kangkung	nasi	pisang

Tahap berikutnya, semua gambar diubah ukurannya menjadi 640x640 pixel menggunakan kode Python. Proses resize bisa dilihat pada Kode 1.

```
import os
import cv2
from tqdm import tqdm

# Konfigurasi path
input_dir = '/content/drive/MyDrive/PROGRAM
SKRIPSI/dataset_raw'
output_dir = '/content/drive/MyDrive/PROGRAM
SKRIPSI/dataset_resized'
target_size = (640, 640)

def process_dataset():
    for class_name in os.listdir(input_dir):
        class_input_path = os.path.join(input_dir, class_name)
        if not os.path.isdir(class_input_path):
            continue

        # Membuat folder untuk images dan labels pada setiap
        kelas
        class_output_images = os.path.join(output_dir,
class_name, "images")
        class_output_labels = os.path.join(output_dir,
class_name, "labels")
        os.makedirs(class_output_images, exist_ok=True)
        os.makedirs(class_output_labels, exist_ok=True)

        # Ambil file gambar dengan ekstensi jpg, png, jpeg
        image_files = [f for f in os.listdir(class_input_path)
if f.lower().endswith(('jpg', 'png', 'jpeg'))]
```

```

        for f in tqdm(image_files, desc=f'Processing
{class_name}') :
            img_path = os.path.join(class_input_path, f)
            img = cv2.imread(img_path)
            if img is None:
                print(f"Warning: {img_path} tidak dapat dibaca,
dilewati.")
                continue

            if img.shape[0] < target_size[1] or img.shape[1] <
target_size[0]:
                interpolation = cv2.INTER_LINEAR
            else:
                interpolation = cv2.INTER_AREA

            # Resize gambar
            resized_img = cv2.resize(img, target_size,
interpolation=interpolation)
            cv2.imwrite(os.path.join(class_output_images, f),
resized_img)

    process_dataset()

    print("\nResizing selesai!")
    print(f"Semua gambar telah diresize ke
{target_size[0]}x{target_size[1]} dan folder labels telah dibuat
untuk setiap kelas.")

```

Tabel 3. Hasil Resize Data

Sebelum	Sesudah
	
1000x1000 pixel	640x640 pixel

b. Pelabelan Gambar

Proses pelabelan gambar dilakukan menggunakan Computer Vision Annotation Tool (CVAT).



Gambar 2 Proses Pelabelan gambar

c. Pembagian Data

Dataset gambar dibagi ke dalam tiga kategori utama yaitu latih, validasi, dan uji. Komposisi jumlah data ketiga kategori tersebut bisa dilihat pada Tabel 4.

Tabel 4. Pembagian Data

Data Latih	Data Uji	Data Validasi
80% (4000)	10% (500)	10% (500)

Proporsi pembagian data yaitu data latih 80% dari total dataset (4000 gambar), data validasi 10% dari total dataset (500 gambar), dan data uji 10% dari total dataset (500 gambar) menggunakan Python untuk melakukan pemisahan data.

```
import os
import shutil
from sklearn.model_selection import train_test_split

# Konfigurasi path
INPUT_DIR = '/content/drive/MyDrive/PROGRAM
SKRIPSI/dataset_resized'
OUTPUT_DIR = '/content/drive/MyDrive/PROGRAM
SKRIPSI/dataset_split'

# Rasio split dataset
TRAIN_RATIO = 0.8
VAL_RATIO = 0.1
TEST_RATIO = 0.1

def copy_files(files, split_type, class_name,
class_images_path, class_labels_path, output_dir):
    for f in files:
        # Generate unique filename dengan prefix kelas
        base_name = f"{class_name}_{f}"
        txt_name = os.path.splitext(f)[0] + ".txt"
        base_txt_name = f"{class_name}_{txt_name}"

        # Path sumber
        img_src = os.path.join(class_images_path, f)
        lbl_src = os.path.join(class_labels_path, txt_name)

        # Path tujuan
        img_dst = os.path.join(output_dir, split_type,
'images', base_name)
        lbl_dst = os.path.join(output_dir, split_type,
'labels', base_txt_name)

        # Salin file gambar
        shutil.copy(img_src, img_dst)

        # Salin file label jika ada
        if os.path.exists(lbl_src):
            shutil.copy(lbl_src, lbl_dst)

def split_dataset():
    # Setup direktori output
    for split in ['train', 'val', 'test']:
        os.makedirs(os.path.join(OUTPUT_DIR, split, 'images'),
exist_ok=True)
        os.makedirs(os.path.join(OUTPUT_DIR, split, 'labels'),
exist_ok=True)
```



```

# Dapatkan daftar kelas yang valid (exclude hidden folders)
classes = [d for d in os.listdir(INPUT_DIR)
            if os.path.isdir(os.path.join(INPUT_DIR, d)) and
not d.startswith('.')]
print(f"Kelas yang ditemukan: {classes}")

for class_name in classes:
    class_path = os.path.join(INPUT_DIR, class_name)
    images_path = os.path.join(class_path, "images")
    labels_path = os.path.join(class_path, "labels")

    # Validasi struktur direktori
    if not os.path.exists(images_path):
        print(f"\n[Peringatan] Direktori gambar tidak
ditemukan untuk kelas {class_name}")
        continue

    # Dapatkan daftar file gambar
    image_files = [f for f in os.listdir(images_path) if
f.lower().endswith(('jpg', 'png', 'jpeg'))]

    # Skip kelas tanpa gambar
    if len(image_files) == 0:
        print(f"\n[Peringatan] Tidak ada gambar untuk kelas
{class_name}")
        continue

    print(f"\nMemproses kelas '{class_name}':
{len(image_files)} gambar ditemukan")

    try:
        # Split dataset dengan pengecekan minimal sampel
        if len(image_files) < 5: # Minimal 5 sampel untuk
split
            print(f"Kelas {class_name} memiliki terlalu
sedikit sampel. Melewati split...")
            continue

        train_files, temp_files = train_test_split(
            image_files,
            train_size=TRAIN_RATIO,
            shuffle=True,
            random_state=42
        )

        # Hitung rasio validasi yang valid
        val_test_ratio = VAL_RATIO / (VAL_RATIO +
TEST_RATIO)
        val_files, test_files = train_test_split(
            temp_files,
            train_size=val_test_ratio,
            random_state=42
        ) if len(temp_files) > 0 else ([], [])

    except ValueError as e:
        print(f"Error saat splitting kelas {class_name}:
{str(e)}")
        continue

```




```
# Salin file
copy_files(train_files, 'train', class_name,
images_path, labels_path, OUTPUT_DIR)
copy_files(val_files, 'val', class_name, images_path,
labels_path, OUTPUT_DIR)
copy_files(test_files, 'test', class_name, images_path,
labels_path, OUTPUT_DIR)

print(f"Split selesai untuk {class_name}:")
print(f" - Train: {len(train_files)}")
print(f" - Val: {len(val_files)}")
print(f" - Test: {len(test_files)}")

if __name__ == "__main__":
    split_dataset()
    print("\nProses pembagian dataset selesai!")
```

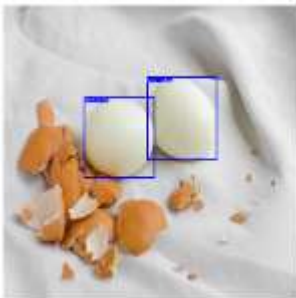
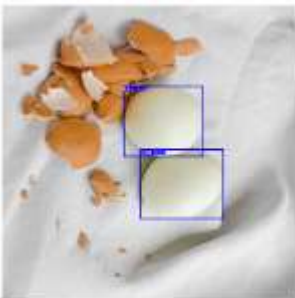
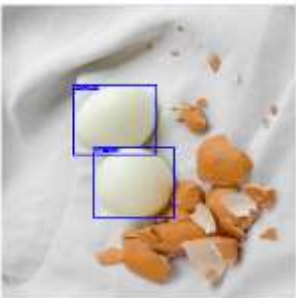
d. Augmentasi Data

Penelitian ini menggunakan dua teknik augmentasi, yaitu rotasi dan mosaic.

a) Rotasi

Gambar diputar sebesar 90° searah jarum jam dan 90° berlawanan arah jarum jam.

Tabel 5. Augmentasi Rotasi

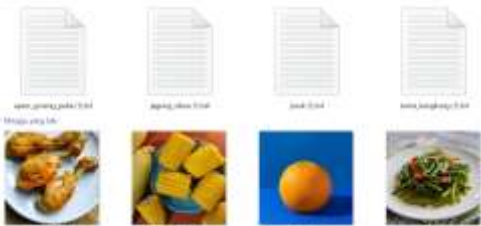
Sebelum	Sesudah	
	90 searah jarum jam	90 berlawanan arah jarum jam
		

Tabel 5 memperlihatkan hasil proses augmentasi rotasi pada sebuah citra telur ayam beserta bounding box-nya. Terlihat bahwa posisi objek dan bounding box berubah mengikuti orientasi citra, hal ini menyatakan bahwa proses rotasi tidak hanya pada gambar namun juga pada label *bounding box* dari objek yang ada di dalam gambar.

b) Mosaic

Teknik mosaic menggabungkan empat gambar berbeda menjadi satu gambar besar dalam format grid.

Tabel 6. Augmentasi *Mosaic*

Sebelum	Sesudah
	

Tabel 6 menampilkan perbandingan kondisi dataset sebelum dan sesudah dilakukan augmentasi mosaik. Empat gambar digabung menjadi satu gambar mosaik yang ditunjukkan dengan nama file “mosaic_0(1).jpg” beserta file label yang memuat informasi *bounding box* gabungan.

3. Pelatihan Model

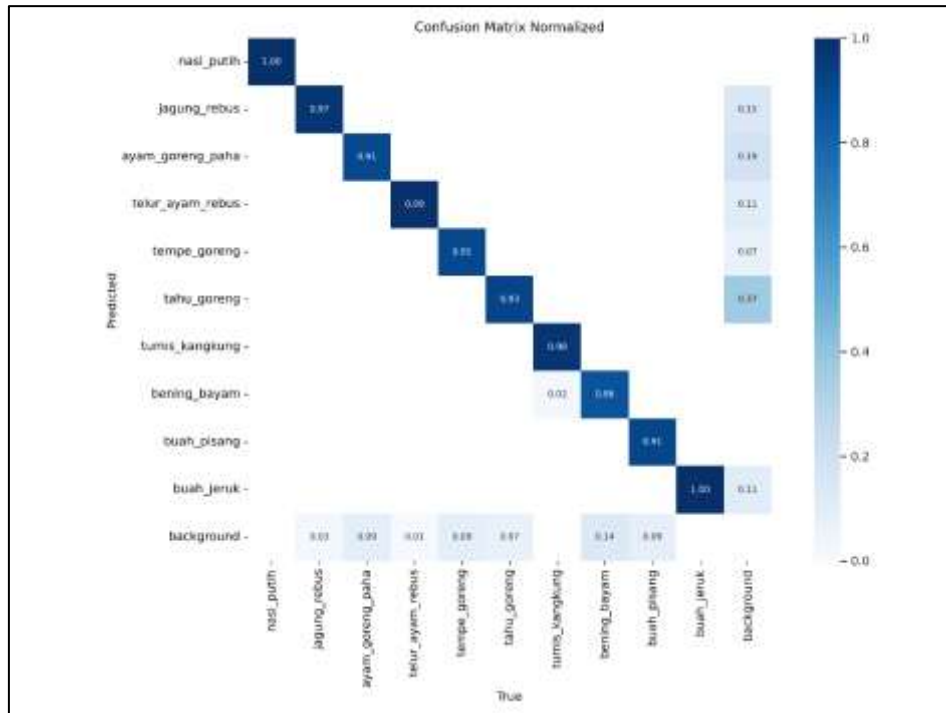
Tabel 8 menyajikan perbandingan performa dari enam model pelatihan. Evaluasi dilakukan menggunakan empat metrik utama: mAP50, mAP, Precision, dan Recall. Berdasarkan hasil yang ditampilkan pada tabel, nilai tertinggi untuk masing-masing metrik ditandai dengan warna hijau, sedangkan nilai terendah diberi warna merah. Model Pelatihan 4 dengan batch size 16 dan learning rate 0,001 mencatatkan performa terbaik secara keseluruhan dengan mAP50 sebesar 98.35%, mAP sebesar 92.72%, Precision 95.57%, dan Recall 96.46%. Sebaliknya, Pelatihan 2 menunjukkan performa paling rendah pada hampir seluruh metrik evaluasi. Sehingga peneliti memilih model 4 untuk dibawa kelangkah berikutnya yaitu pengujian.

Tabel 7. Perbandingan Hasil Pelatihan Semua Model

Model	Epoch	Batch Size	Learning Rate	mAP 50	mAP	Precision	Recall
Pelatihan 1	100	16	0,01	97.92%	89.99%	94.71%	94.69%
Pelatihan 2	100	32	0,01	97.62%	88.89%	93.53%	95.31%
Pelatihan 3	100	64	0,01	97.86%	89.66%	94.62%	95.57%
Pelatihan 4	100	16	0,001	98.35%	92.72%	95.57%	96.46%
Pelatihan 5	100	32	0,001	98.08%	90.30%	94.03%	95.59%
Pelatihan 6	100	64	0,001	97.92%	90.11%	94.95%	93.94%

Gambar 3 menampilkan confusion matrix hasil pengujian model terbaik terhadap data uji. Matrix ini telah dinormalisasi sehingga setiap nilai menunjukkan proporsi prediksi yang benar terhadap jumlah total sampel untuk masing-masing kelas sebenarnya. Berdasarkan gambar, dapat dilihat bahwa mayoritas kelas berhasil diklasifikasikan dengan sangat baik oleh model. Beberapa kelas seperti nasi_putih, telur_ayam_rebus, buah_jeruk, dan tumis_kangkung menunjukkan akurasi prediksi yang sangat tinggi, bahkan mencapai 100% untuk nasi_putih dan buah_jeruk, serta lebih dari 98% untuk kelas lainnya. Meskipun demikian, masih terdapat beberapa kelas yang menunjukkan tingkat kesalahan yang lebih tinggi dibandingkan yang lain. Misalnya, kelas bening_bayam hanya dikenali dengan benar sebesar 86%, dan mengalami kesalahan prediksi ke kelas background sebesar 14% serta sedikit ke kelas buah_pisang. Kelas tahu_goreng juga menunjukkan akurasi yang sedikit lebih rendah, yaitu 93%, dengan kesalahan prediksi ke latar belakang sebesar 7%. Kelas lainnya seperti tempe_goreng, ayam_goreng_paha, dan buah_pisang masih mempertahankan akurasi di atas 90%, meskipun terdapat kesalahan minor dalam prediksi. Kolom terakhir pada confusion matrix, yaitu background, juga menunjukkan adanya beberapa kesalahan klasifikasi dari objek nyata yang diprediksi sebagai latar belakang, seperti pada tahu_goreng dan bening_bayam.





Gambar 3. Confusion Matrix Model Terbaik

Matriks hasil pengujian model pada dataset pengujian bisa dilihat pada Tabel 8.

Tabel 8. Hasil Pengujian Model Pada Dataset Pengujian

Kelas	Precision	Recall	mAP50	mAP(50-95)
Nasi Putih	1	1	0.995	0.978
Jagung Rebus	0.968	0.968	0.963	0.925
Ayam Goreng Paha	0.929	0.897	0.938	0.899
Telur Ayam Rebus	0.979	0.993	0.992	0.958
Tempe Goreng	0.977	0.904	0.949	0.878
Tahu Goreng	0.959	0.925	0.957	0.88
Tumis Kangkung	1	0.98	0.99	0.963
Bening Bayam	0.977	0.86	0.927	0.92
Buah Pisang	1	0.915	0.957	0.858
Buah Jeruk	0.97	1	0.984	0.962
Rata-rata	0.976	0.944	0.965	0.922

KESIMPULAN

Kesimpulan dari penelitian ini Adalah sebagai berikut:

1. Model YOLOv11 berhasil dikembangkan untuk mengenali 10 jenis makanan.
2. Evaluasi kinerja model menunjukkan hasil yang sangat baik dengan mAP50 sebesar 98,35%, precision 95,57%, dan recall 96,46%, mengindikasikan kemampuan model dalam menyeimbangkan akurasi deteksi dan cakupan objek. Keberhasilan ini didukung oleh pra-pemrosesan data yang teliti, augmentasi (rotasi dan mosaic), serta penyesuaian hyperparameter (epoch 100, batch size 16, learning rate 0,001) yang optimal.
3. Implementasi Smart Nutrition tidak hanya berfungsi sebagai alat bantu deteksi dan perhitungan gizi, tetapi juga menjadi sarana edukatif bagi siswa, sekolah, dapur gizi, dan pemangku kepentingan lainnya dalam memahami nilai gizi makanan secara berbasis data sehingga



berkontribusi dalam meningkatkan literasi gizi masyarakat dan mendukung keberhasilan program Makan Bergizi Gratis (MBG) menuju generasi emas 2045.

UCAPAN TERIMA KASIH

Peneliti mengucapkan terima kasih yang sebesar-besarnya kepada Universitas Lampung atas dukungan pendanaan melalui Hibah Penelitian BLU Tahun 2025. Dukungan ini telah memungkinkan pelaksanaan penelitian berjalan dengan lancar dan berperan penting dalam mewujudkan luaran penelitian berupa sistem pemantauan gizi berbasis kecerdasan buatan yang diharapkan dapat memberikan kontribusi nyata bagi peningkatan literasi gizi masyarakat dan mendukung keberhasilan program Makan Bergizi Gratis.

DAFTAR PUSTAKA

- [1] A. Agustini, “Evaluasi Implementasi Program Makan Bergizi Gratis di Sekolah Dasar: Tantangan dan Strategi,” *Jurnal Gizi dan Kesehatan Indonesia*, vol. 7, no. 2, pp. 101–110, 2025. [Online]. Available: <https://doi.org/10.35814/jgki.v7i2.4589>
- [2] S. Goel, R. Jain, and M. Sharma, “Multi-Dish Food Recognition and Calorie Estimation Using Deep Convolutional Networks,” *IEEE Access*, vol. 12, pp. 45201–45212, 2024. [Online]. Available: <https://doi.org/10.1109/ACCESS.2024.3389012>
- [3] A. Zianka, D. Pratama, and E. Santosa, “CNN-Based Food Image Detection for Stunting Prevention Application,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 10, no. 4, pp. 557–564, 2023. [Online]. Available: <https://doi.org/10.25126/jtik.2023104567>
- [4] A. Riswanto, P. B. Wintoro, and R. A. Pradipta, “Implementasi SSD untuk Klasifikasi Makanan Tradisional Indonesia,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 5, pp. 998–1006, 2024. [Online]. Available: <https://doi.org/10.29207/resti.v8i5.4809>
- [5] D. Sapkota, J. Chae, and S. Yoon, “YOLOv11: Optimized Object Detection Architecture for Real-Time Food Recognition,” *Sensors*, vol. 24, no. 8, pp. 3587–3598, 2024. [Online]. Available: <https://doi.org/10.3390/s24083587>
- [6] Y. Kawano and K. Yanai, “FoodCam: A Real-Time Mobile Food Recognition System Employing Deep Learning,” *Journal of Information and Communication Technology*, vol. 17, no. 1, pp. 32–45, 2021. [Online]. Available: <https://doi.org/10.32890/jict2021.17.1.4>
- [7] R. Li, T. Liu, and F. Zhao, “Deep Learning-Based Nutritional Analysis Using Food Images,” *Computers in Biology and Medicine*, vol. 168, no. 12, p. 107118, 2024. [Online]. Available: <https://doi.org/10.1016/j.compbimed.2024.107118>
- [8] N. S. Hartati and M. Hidayat, “Integrasi AI untuk Pemantauan Status Gizi Anak Sekolah Berbasis Citra Makanan,” *Jurnal Teknologi dan Sistem Komputer*, vol. 12, no. 3, pp. 245–253, 2023. [Online]. Available: <https://doi.org/10.14710/jtsiskom.2023.12345>
- [9] F. Zhang, H. Chen, and Y. Xu, “Food Recognition with YOLOv5 and Calorie Estimation for Dietary Assessment,” *IEEE Transactions on Multimedia*, vol. 25, no. 4, pp. 2319–2331, 2023. [Online]. Available: <https://doi.org/10.1109/TMM.2023.3248790>
- [10] Kementerian Kesehatan Republik Indonesia, *Tabel Komposisi Pangan Indonesia (TKPI)*, Jakarta: Badan Penelitian dan Pengembangan Kesehatan, 2020. [Online]. Available: <https://tkpi.kemkes.go.id>

