Available *online* at : http://chipset.fti.unand.ac.id/

Journal on Computer Hardware, Signal Processing, Embedded System and Networking

| ISSN (Online) 2722-4422 |

Embedded System

Sistem Pendeteksi Penyakit dan Pemantauan Kondisi Tanah serta Rekomendasi Pengendaliannya pada Tanaman Cabai

Alhafiz Zhurrahman Mirza Utama[1], *Rifki Suwandi*[*2], *Lathifah Arief* [3]

[1,2] *Departemen Teknik Komputer, Fakultas Teknologi Informasi, Universitas Andalas, Limau Manis Kec. Pauh, Kota Padang, Sumatera Barat, Indonesia*

ARTICLE INFORMATION

Received: September 23, 2024
Revised: Juli 16, 2026
Available online: October 31, 2026

KEYWORDS

Deteksi Penyakit, YOLO, Raspberry Pi 4B, Tanaman Cabai, Sensor pH, Sensor Kelembaban, Pemantauan Kondisi Tanah.

CORRESPONDENCE

E-mail: rifkisuwandi@it.unand.ac.id

A B S T R A C T

The cultivation of chili plants faces challenges in early disease detection and accurate soil condition monitoring, which can potentially reduce crop yields and increase the risk of plant damage. This research aims to develop a disease detection and soil condition monitoring system based on Raspberry Pi, utilizing the You Only Look Once (YOLO) version 5 algorithm to detect diseases in chili plants from images captured by a camera. The system uses Raspberry Pi 4B as the data processing center, with a camera for capturing plant images and pH and soil moisture sensors to monitor soil conditions. The data obtained is used to provide better handling recommendations for farmers. Test results show that this system can detect plant diseases with 87.5% accuracy and an F1-score of 93.3% at a distance of 5 to 25 cm, although the accuracy decreases at distances greater than 25 cm. Furthermore, soil condition measurements provide important information about soil pH and moisture, helping farmers make more informed decisions in maintaining plant health. All disease detection and soil condition information is displayed on an LCD screen connected to the Raspberry Pi, making the system user-friendly for farmers. Therefore, this system is expected to be an effective solution for increasing crop yields and reducing losses due to diseases in chili plants, and it can be more broadly applied in the agricultural sector.

PENDAHULUAN

Cabai merupakan komoditas penting dalam sektor pertanian, mengingat nilai ekonominya yang tinggi dan dampaknya yang signifikan terhadap pasar komoditas lainnya. Data dari Badan Pusat Statistik (BPS) menunjukkan bahwa cabai menjadi salah satu komoditas hortikultura dengan kontribusi besar terhadap ekonomi lokal di banyak daerah di Indonesia. Harga jual cabai yang relatif tinggi menjadikannya komoditas strategis bagi petani untuk mencapai pendapatan yang layak. Namun, dalam praktiknya, keberhasilan produksi cabai dipengaruhi oleh berbagai faktor, seperti kondisi iklim, kualitas tanah, ketersediaan air, serta ancaman dari organisme patogen dan hama[1].

Salah satu tantangan utama yang dihadapi petani cabai adalah gangguan hama dan penyakit, yang sering kali menyebabkan kegagalan panen dan berdampak pada stabilitas harga di pasar. Kurangnya pengetahuan petani mengenai jenis penyakit tanaman cabai dan cara pengendaliannya menjadi masalah yang signifikan[2]. Akibatnya, banyak petani yang melakukan tindakan pengendalian yang tidak tepat, seperti penggunaan pupuk atau obat yang tidak sesuai dengan penyakit yang menyerang .

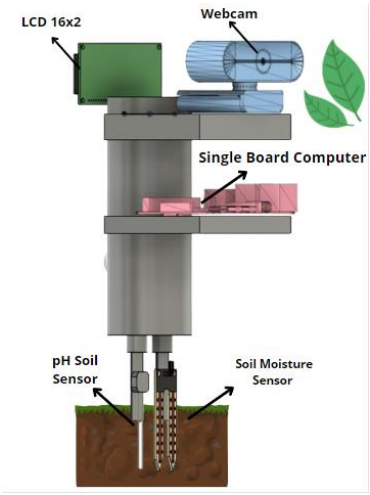
Masalah lain yang dihadapi petani adalah kesulitan dalam mengidentifikasi gejala penyakit pada tanaman mereka, yang menyebabkan penggunaan sumber daya yang tidak efisien dan meningkatkan risiko kerusakan lingkungan [3]. Selain itu, faktor penting lain seperti pemantauan kelembaban dan pH tanah sering kali diabaikan, padahal hal ini berperan penting dalam pertumbuhan tanaman cabai [4][5]. Di berbagai daerah, seperti Nagari Surantih di Pesisir Selatan dan Solok Selatan, kondisi ini menyebabkan penurunan produksi yang serius, mengingat petani lambat dalam mengambil tindakan pengendalian yang tepat terhadap hama dan penyakit [6][7].

Dalam konteks ini, diperlukan solusi inovatif yang dapat membantu petani dalam mendeteksi dini penyakit dan memantau kondisi tanah secara akurat, sehingga dapat mengoptimalkan hasil panen dan mengurangi dampak negatif terhadap lingkungan.

https://doi.org/10.25077/chipset.7.02.132-139.2026

Attribution-NonCommercial-ShareAlike 4.0 International. Some rights reserved

METODE

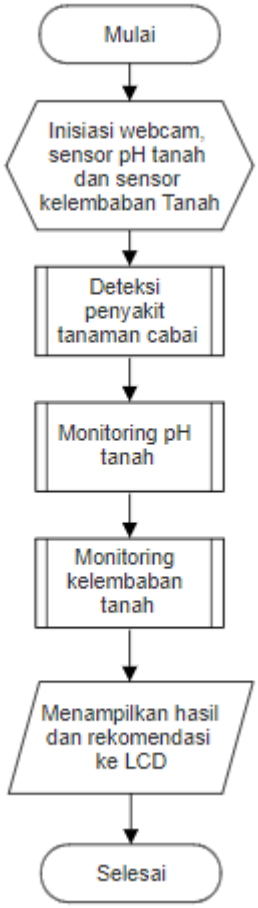


Gambar 1. Rancangan Umum Sistem

Sistem ini didesain menyerupai gimbal kamera, di mana berbagai komponen ditempatkan pada posisi strategis untuk fungsionalitas yang optimal. Webcam dan layar LCD 16x2 diletakkan di bagian atas untuk memudahkan pengambilan gambar dan tampilan informasi, sementara Raspberry Pi 4 model B dan Arduino Nano berada di bagian tengah sebagai pusat pemrosesan. Di bagian bawah, terdapat sensor pH tanah dan sensor kelembaban tanah yang dirancang agar bisa langsung ditancapkan ke tanah untuk pemantauan kondisi tanah.

Sistem ini digunakan untuk mengambil gambar daun tanaman cabai serta membaca informasi kelembaban dan pH tanah dari tanaman tersebut. Sensor pH tanah berfungsi untuk mengukur tingkat keasaman atau kebasaan tanah, sedangkan sensor kelembaban mengukur kadar air di dalam tanah. Kamera berperan dalam menangkap gambar daun tanaman cabai yang akan digunakan untuk mendeteksi penyakit.

Semua input dari sensor dan kamera diproses oleh Raspberry Pi 4, yang bertindak sebagai pusat pemrosesan. Hasil pemrosesan mencakup informasi tentang penyakit tanaman cabai, nilai kelembaban, pH tanah, serta rekomendasi pengendalian yang tepat berdasarkan informasi yang diperoleh. Output tersebut kemudian ditampilkan pada layar LCD 16x2, memungkinkan pengguna untuk mengakses informasi secara mudah dan real-time. Komponen perangkat keras yang digunakan dalam sistem ini meliputi Raspberry Pi 4 model B, Arduino Nano, webcam, sensor pH tanah, Modbus RS485, sensor kelembaban tanah, dan LCD 16x2, yang bekerja secara terpadu untuk mendukung seluruh operasi sistem.



Gambar 2. Flowchart Rancangan Proses Sistem

Pada Gambar 2, flowchart menggambarkan alur kerja sistem yang dimulai dengan inisialisasi seluruh komponen perangkat keras, termasuk webcam dan sensor-sensor yang digunakan untuk pemantauan kondisi tanaman dan tanah. Setelah proses inisialisasi selesai, sistem akan melakukan pembacaan data dari sensor pH tanah, sensor kelembaban tanah, serta pengambilan gambar tanaman melalui webcam. Data yang diperoleh dari pembacaan sensor dan hasil gambar kemudian diproses oleh Raspberry Pi untuk menganalisis kondisi tanaman dan tanah. Berdasarkan hasil analisis tersebut, sistem akan menghasilkan rekomendasi pengendalian yang sesuai dengan kondisi yang terdeteksi. Langkah terakhir dari proses ini adalah mengirimkan informasi hasil analisis dan rekomendasi pengendalian ke layar LCD, sehingga pengguna dapat melihat dan mengambil tindakan yang diperlukan berdasarkan informasi tersebut.

HASIL DAN ANALISA

Implementasi Sistem

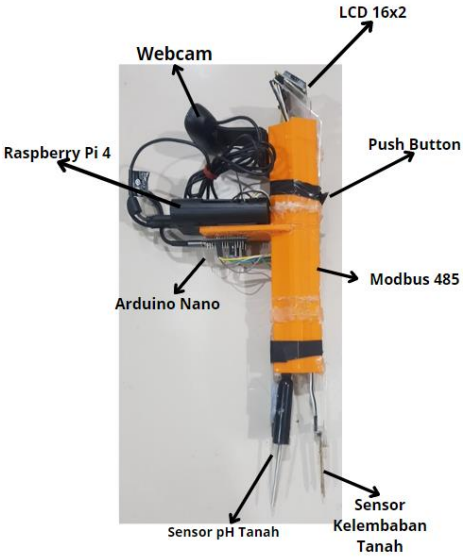
Implementasi sistem ini bertujuan untuk menguji dan memastikan bahwa sistem yang dirancang berfungsi sesuai dengan permasalahan yang diuraikan pada latar belakang, serta mencapai tujuan pembuatan sistem. Implementasi ini mencakup integrasi perangkat keras seperti Raspberry Pi 4 model B, Arduino Nano, webcam, sensor pH tanah, Modbus RS485 dan

https://doi.org/10.25077/chipset.7.02.132-139.2026

sensor kelembapan tanah. Selain itu, juga mencakup implementasi perangkat lunak, termasuk algoritma YOLO untuk deteksi penyakit dan pengelolaan data sensor. Selanjutnya, perangkat keras dan perangkat lunak diintegrasikan dalam satu kesatuan sistem.

Implementasi Perangkat Keras

Implementasi perangkat keras melibatkan perakitan komponen sesuai spesifikasi sistem, dengan desain menyerupai gimbal kamera. Webcam dan LCD 16x2 diletakkan di bagian atas, Raspberry Pi 4 model B dan Arduino Nano di bagian tengah, serta sensor pH dan kelembaban tanah di bagian bawah untuk pengukuran langsung. Komponen utama yang digunakan meliputi Raspberry Pi 4, Arduino Nano, webcam, sensor pH tanah, Modbus RS485, sensor kelembaban tanah, dan LCD 16x2, yang semuanya diintegrasikan untuk mendukung deteksi penyakit tanaman dan pemantauan kondisi tanah.



Gambar 3. Hasil Implementasi Sistem

Pada Gambar 3, perangkat keras yang diimplementasikan dalam sistem memiliki fungsi sebagai berikut:

1. Webcam: Berfungsi untuk menangkap citra digital berupa gambar sebagai input bagi Raspberry Pi.
2. Sensor pH Tanah: Membaca nilai pH tanah dengan rentang 3 hingga 9, yang digunakan untuk merekomendasikan tindakan pengendalian.
3. Modbus RS485: Mengkonversi hasil bacaan sensor tanah menjadi alamat-alamat yang dikirimkan secara serial ke Arduino Nano.
4. Sensor Kelembaban Tanah: Mengukur kadar air tanah dalam rentang 0 hingga 4095, yang dikonversi menjadi persentase 0-100%, dan digunakan sebagai dasar rekomendasi pengendalian.
5. Arduino Nano: Berfungsi sebagai kontroler yang menerima data dari sensor pH dan kelembaban tanah, menentukan rekomendasi, dan mengirimkan data secara serial ke Raspberry Pi. Arduino Nano digunakan karena sensor pH tanah memerlukan tegangan 5V, sedangkan pin GPIO Raspberry Pi hanya menyediakan 3.3V.
6. Raspberry Pi 4 Model B: Bertindak sebagai kontroler utama yang mengatur proses sistem, melakukan deteksi penyakit menggunakan YOLO, dan mengirimkan data hasil deteksi serta data dari Arduino ke LCD.
7. Push Button: Berfungsi sebagai input untuk mengontrol sistem selama operasional.
8. LCD 16x2: Menampilkan informasi hasil deteksi dari sensor pH, kelembaban tanah, dan rekomendasi pengendalian dalam bentuk simbol A-H, yang dijelaskan lebih lanjut dalam buku panduan.

Perangkat keras ini bekerja secara terintegrasi untuk mendeteksi penyakit tanaman dan memantau kondisi tanah dengan memberikan rekomendasi pengendalian yang sesuai.

Implementasi Perangkat Lunak

Implementasi perangkat lunak diperlukan agar perangkat keras dapat bekerja sesuai dengan kebutuhan sistem pada spesifikasi produk. Adapun perangkat lunak yang diimplementasikan ke dalam sistem adalah sebagai berikut :

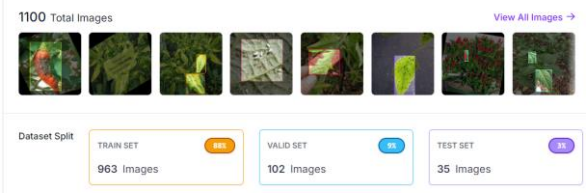
1. Library: OpenCV digunakan untuk menangkap dan memproses gambar dari kamera yang terhubung ke Raspberry Pi, kemudian digunakan sebagai input YOLO. Library serial digunakan untuk komunikasi antara Raspberry Pi dan Arduino Nano untuk menerima data sensor pH dan kelembaban tanah yang ditampilkan pada LCD.
2. Bahasa Pemrograman: Python digunakan untuk kendali sistem di Raspberry Pi, sementara Arduino (C/C++) digunakan untuk pembacaan sensor pH dan kelembaban. PyTorch digunakan untuk melatih model YOLO yang mendeteksi penyakit pada gambar dari webcam.

Deteksi Obejek Menggunakan YOLO

Pada projek ini versi YOLO yang digunakan adalah YOLOv5. Deteksi objek menggunakan YOLOv5 digunakan untuk mendeteksi adanya objek penyakit tanaman cabai pada gambar yang sudah diambil oleh webcam. YOLOv5 mampu mengenali adanya objek penyakit tanaman cabai pada gambar yang ditangkap oleh webcam dengan melalui proses training terlebih dahulu.

Proses Training YOLO

Proses training dilakukan YOLOv5 yang menggunakan framework pytorch pada platform google colab. Dengan menggunakan dataset yang sudah dikumpulkan dari public dataset. Dalam dataset ini sudah ada file test, train, dan val yang akan diuji oleh algoritma YOLOv5. Berikut adalah gambar implementasi dataset yang akan digunakan pada proses training.



Gambar 4. Dataset untuk Training

Proses training model dapat dilihat pada gambar berikut.

```
# train yolov5s on custom data for 100 epochs
# time its performance
%%time
%cd /content/yolov5/
!python train.py --img 640 --batch 16 --epochs 180 --data /content/yolov5/caps
```

```
      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
  167/179      4.64G    0.01941    0.02617   0.002028         10        640:
                 Class     Images  Instances          P          R      mAP50
                   all        102        202      0.617      0.642      0.612

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
  168/179      4.64G    0.01925    0.02596   0.001862         27        640:
                 Class     Images  Instances          P          R      mAP50
                   all        102        202      0.634      0.628      0.607
```

Gambar 5. Training dengan YOLOv5

Training YOLOv5 dimulai dengan menyiapkan dataset gambar dan anotasi dalam format yang sesuai, diikuti dengan menginstal dependensi dan mengonfigurasi model serta hyperparameter yang dibutuhkan. Setelah konfigurasi, proses pelatihan dilakukan dengan menjalankan skrip pelatihan, yang akan memproses data untuk menghitung dan meminimalkan loss serta memperbarui bobot model. Selama pelatihan, model diuji pada data validasi untuk memastikan kinerjanya dan menghindari overfitting. File-file dataset akan diakses dari sebuah file yang berfungsi untuk konfigurasi dataset yang bernama data.yaml dengan isi sebagai berikut:

path: ../yolov5/data
train: train/images
val: valid/images
test: test/images

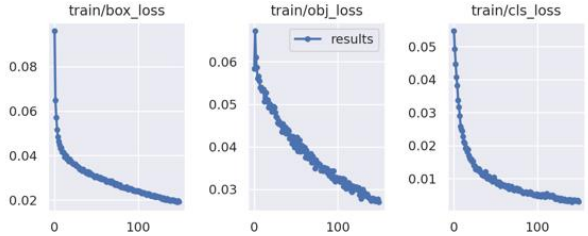
names:
0: Cabai-Sehat
1: Cabai-Terkena-Daun-Berkerut
2: Cabai-Terkena-Penyakit-Anthacnose
3: Cabai-Terkena-Penyakit-Kutu-Putih
4: Cabai-Terkena-Penyakit-Leaf-Spot
5: Cabai-Terkena-Penyakit-Yellowish

Hasil dari pelatihan atau training model pada YOLOv5 akan disimpan dalam bentuk pytorch dengan nama file best.pt. File best.pt ini merupakan model terbaik dari proses training dengan mAP tertinggi. Berikut model hasil proses training.

```
                 Class     Images  Instances          P          R      mAP50
                   all        102        202      0.625      0.611      0.632
           Cabai-Sehat        102         27      0.639      0.556      0.596
Cabai-Terkena-Daun-Berkerut        102         51      0.635      0.588      0.567
Cabai-Terkena-Penyakit-Anthacnose        102         24      0.562      0.583      0.572
0.137
Cabai-Terkena-Penyakit-Kutu-Putih        102         38      0.574      0.684      0.632
0.293
Cabai-Terkena-Penyakit-Leaf-Spot        102         39      0.696      0.645      0.682
0.328
Cabai-Terkena-Penyakit-Yellowish        102         23      0.647      0.609      0.599
```

Gambar 6. Hasil Training

Pada hasil training di atas didapatkan mAP untuk keselurahan sebesar 63.2% dengan hasil mAP yang menunjukkan peforma model yang cukup baik namun tetap butuh perbaikan pada dataset agar dapat meningkatkan peforma keseluruhan model.



Gambar 7. Grafik Hasil Training

Gambar tersebut menampilkan perubahan nilai kerugian (loss) selama proses validasi model YOLOv5, meliputi bounding box loss (val/box_loss), object loss (val/obj_loss), dan classification loss (val/cls_loss). Ketiga nilai loss ini menurun seiring bertambahnya epoch, yang menunjukkan peningkatan kinerja model dalam memprediksi bounding box, mendeteksi objek, dan mengklasifikasikan objek. Meskipun terdapat fluktuasi pada val/obj_loss, tren keseluruhan tetap menurun, menunjukkan bahwa model mampu belajar dari data latih dan menjaga kinerja yang baik pada data validasi, mengindikasikan generalisasi yang baik.

Implementasi Model YOLOv5 ke Sistem

Sistem ini diprogram menggunakan bahasa Python. Dalam implementasinya, model YOLOv5 bekerja setelah webcam menangkap gambar dan menyimpannya ke direktori. Gambar tersebut kemudian digunakan oleh YOLOv5 untuk melakukan deteksi menggunakan model yang telah dilatih sebelumnya.

```
# Jalankan deteksi objek pada gambar yang baru ditangkap
os.system(f'python /home/pi/yolov5/detect.py --weights /home/pi/Downloads/best.pt --img 640

def get_disease_and_recommendation():
    # Baca hasil deteksi dari file txt yang disimpan
    result_file = os.path.splitext(os.path.basename(image_path))[0] + '.txt'
    label_path = os.path.join('/home/pi/yolov5/runs/detect/exp/labels', result_file)

    if os.path.exists(label_path):
        with open(label_path) as f:
            result = f.read().strip().split()
    else:
        result = []
```

Gambar 8. Implementasi YOLOv5 pada Sistem

Setelah model YOLOv5 diimplementasikan, dilakukan mapping untuk memberikan rekomendasi pengendalian terhadap hasil deteksi penyakit yang sudah dijalankan sitem. Berikut gambar implementasi pemberian rekomendasi untuk hasil deteksi penyakit.

```
    # Mapping hasil deteksi ke rekomendasi
recommendation_map = {
    'Daun-Berkerut': 'Pengendalian (A)',
    'Anthacnose': 'Pengendalian (B)',
    'Kutu-Putih': 'Pengendalian (C)',
    'Leaf-Spot': 'Pengendalian (D)',
    'Yellowish': 'Pengendalian (E)'
}
```

Gambar 9. Program Menentukan Rekomendasi

Program python di atas melakukan mapping atau pencocokan antara penyakit dengan rekomendasi pengendalian yang akan diberikan oleh sistem.

Pengujian Webcam

Pengujian webcam dilakukan untuk memastikan kemampuannya menangkap gambar dengan resolusi 720p secara konsisten. Webcam ditempatkan pada posisi tetap dan digunakan untuk mengambil gambar objek statis dari berbagai sudut dan jarak. Setelah setiap pengambilan gambar, resolusi dicatat untuk memastikan gambar yang dihasilkan memenuhi standar kualitas yang diperlukan untuk deteksi objek. Pengujian ini diulang dengan berbagai objek dan kondisi, dan hasilnya dianalisis untuk menilai apakah webcam dapat mempertahankan resolusi 720p secara stabil dalam berbagai situasi.

Tabel 1. Pengujian Webcam

Percobaan ke-	**Resolusi Gambar**	**Gambar**
1	1280x720	
2	1280x720	

https://doi.org/10.25077/chipset.7.02.132-139.2026

3	1280x720	
4	1280x720	
5	1280x720	

Berdasarkan hasil pengujian, analisis menunjukkan bahwa webcam mampu menangkap gambar dengan resolusi 720p secara konsisten. Pengujian dengan objek statis maupun variasi sudut pengambilan gambar berhasil mempertahankan resolusi 720p dalam setiap kondisi tanpa penurunan kualitas. Resolusi gambar tetap stabil tanpa fluktuasi, menandakan bahwa webcam memiliki kinerja yang memadai untuk memenuhi kebutuhan pengambilan gambar pada resolusi 720p. Dengan demikian, dapat disimpulkan bahwa webcam ini dapat diandalkan untuk proses deteksi objek yang memerlukan gambar dengan resolusi tersebut.

Pengujian Sensor Kelembaban Tanah

Pengujian sensor kelembaban tanah dilakukan untuk memastikan akurasi dalam mendeteksi kondisi tanah kering atau basah. Sensor ini diujikan untuk memberikan pembacaan persentase dari 0% (tanah sangat kering) hingga 100% (tanah sangat basah), dengan fokus memastikan hasil sesuai dengan kondisi aktual. Pengujian dilakukan dengan menancapkan sensor sedalam 4 meter, dan hasil kelembaban dicatat serta dibandingkan dengan pembacaan dari Soil Moisture Tester sebagai validasi. Pengukuran kelembaban dilakukan selama 10 menit pada area yang sama untuk memastikan konsistensi hasil.

Tabel 2. Pengujian Sensor Kelembaban Tanah

No	Waktu	Sensor Kelembaban Tanah	*Soil Moisture Tester*	Selisih
1.	12.10	16%	14%	2%
2.	12.11	15%	13%	2%
3.	12.12	15%	12%	3%
4.	12.13	15%	12%	3%
5.	12.14	16%	12%	4%
6.	12.15	14%	12%	2%
7.	12.16	12%	11%	1%
8.	12.17	11%	11%	0%
9.	12.18	11%	12%	1%
10	12.19	12%	12%	0%
Rata-rata selama 10 menit		14%	12%	2%

Hasil pengujian menunjukkan bahwa sensor kelembaban tanah mencatat nilai sedikit lebih tinggi dibandingkan dengan Soil Moisture Tester selama pengujian 10 menit. Pada rentang waktu 12.10 hingga 12.19, sensor mencatat antara 11% hingga 16%, sedangkan Soil Moisture Tester mencatat 11% hingga 14%. Perbedaan lebih jelas terlihat pada 12.10 hingga 12.14, di mana sensor kelembaban memberikan nilai lebih tinggi. Namun, dari 12.15 hingga 12.19, perbedaan semakin kecil, bahkan setara pada pengukuran terakhir. Rata-rata pembacaan sensor kelembaban tanah sekitar 14%, sementara Soil Moisture Tester rata-rata 12%. Perbedaan ini mungkin akibat kalibrasi atau sensitivitas alat, namun secara keseluruhan, sensor menunjukkan performa yang andal.

Pengujian Sensor pH Tanah

Pengujian sensor pH tanah bertujuan untuk memastikan akurasi pembacaan nilai pH tanah dalam rentang 3,5 hingga 8, di mana 3,5 menunjukkan tanah sangat asam dan 8 menunjukkan tanah sangat basa. Pengujian ini dilakukan dengan mencatat hasil pembacaan sensor pH tanah, kemudian membandingkannya dengan pembacaan dari Soil pH Tester sebagai alat validasi. Pengukuran dilakukan pada area yang sama selama 10 menit, dengan fokus pada memastikan bahwa sensor pH tanah mampu memberikan hasil yang konsisten dan sesuai dengan kondisi sebenarnya.

Tabel 3. Pengujian Sensor pH tanah

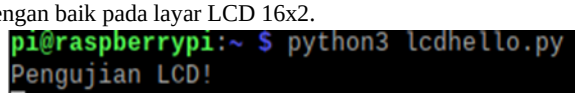
No	Waktu	Sensor pH Tanah	Soil pH Tester	Selisih
1.	11.55	9	6.5	2%
2.	11.56	6.61	6.5	2%
3.	11.57	6.37	6.3	3%
4.	11.57	6.52	6.5	3%
5.	11.58	6.47	6.5	4%
6.	11.59	6.52	6.5	2%
7.	12.00	6.51	6.5	1%
8.	12.01	6.52	6.5	0%
9.	12.02	6.52	6.5	1%
10	12.03	6.52	6.5	0%
Rata-rata selama 10 menit		6.76	6.48	0.28

Berdasarkan hasil pengujian, sensor pH tanah mencatat nilai yang sedikit lebih tinggi dibandingkan dengan Soil pH Tester selama periode pengujian 10 menit. Pada awalnya, sensor menunjukkan nilai pH 9, yang disebabkan oleh waktu penyesuaian sensor dengan lingkungannya, sekitar 15-20 detik. Setelah stabil, dari pukul 11.56 hingga 12.03, sensor pH tanah mencatat nilai pH antara 6.37 hingga 6.52, sedangkan Soil pH Tester mencatat antara 6.3 hingga 6.5. Pada menit-menit awal, sensor pH tanah mencatat nilai lebih tinggi, namun perbedaan antara keduanya berkurang dan menjadi lebih stabil di akhir pengukuran. Rata-rata pembacaan sensor pH tanah adalah 6.76, sementara Soil pH Tester mencatat rata-rata 6.48. Perbedaan ini kemungkinan dipengaruhi oleh sensitivitas alat, namun secara keseluruhan, sensor pH tanah menunjukkan kinerja yang baik dan dapat diandalkan untuk mendeteksi pH tanah.

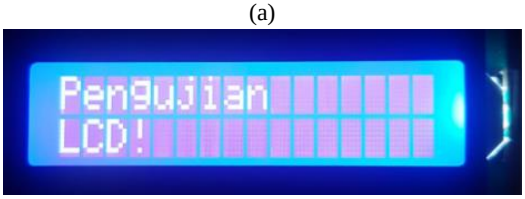
Pengujian LCD 16x2

Pengujian LCD dilakukan untuk memastikan bahwa LCD dapat menerima dan menampilkan data yang dikirim dari Raspberry Pi. Raspberry Pi diprogram menggunakan bahasa Python dan pustaka `rpi_lcd`, dengan modul I2C yang terhubung ke LCD untuk mengoptimalkan penggunaan pin GPIO pada Raspberry Pi. Pengujian ini mencakup pengaturan kontras LCD melalui kode Python agar karakter tampil dengan jelas, serta membandingkan hasil tampilan di LCD dengan output di terminal. Hasil pengujian

menunjukkan bahwa kata "Pengujian LCD" dapat ditampilkan dengan baik pada layar LCD 16x2.



(a)



(b)

Gambar 10 (a),(b) Hasil pengujian LCD 16x2

Berdasarkan hasil pengujian menunjukkan bahwa LCD dapat menerima data dari Raspberry Pi lalu menampilkannya dengan kontras cahaya yang baik sehingga dapat dibaca dengan jelas.

Pengujian Webcam dengan Model YOLOv5 Berdasarkan Jarak

Pengujian jarak webcam dengan model YOLOv5 dilakukan untuk memastikan dalam jarak berapa kamera dan model dapat mendeteksi dengan baik dan optimal. Pengujian dilakukan dengan mengambil gambar tanaman cabai dari beberapa jarak 100 cm, 50 cm, 20 cm, 10 cm, 5 cm.

Tabel 4. Pengujian Webcam dengan Model YOLOv5 Berdasarkan Jarak

Percobaan Ke-	Jarak	Gambar	Hasil
1	100 cm		Tidak Terdeteksi
2	50 cm		Tidak Terdeteksi
3	25 cm		Terdeteksi
4	15 cm		Terdeteksi
5	5 cm		Terdeteksi

Berdasarkan hasil pengujian, webcam efektif digunakan dalam rentang jarak 5-25 cm untuk menangkap citra tanaman cabai, memungkinkan deteksi objek yang baik menggunakan model YOLOv5. Pada jarak tersebut, objek masih terdeteksi dengan jelas. Namun, pada jarak lebih dari 25 cm, seperti 50 cm dan 100 cm, deteksi objek gagal, kemungkinan disebabkan oleh ukuran objek yang lebih kecil, penurunan kualitas resolusi, atau jarak fokus kamera yang tidak optimal. Ini menunjukkan bahwa performa deteksi objek sangat bergantung pada jarak antara kamera dan objek.

Pengujian Webcam dengan Model YOLOv5 Berdasarkan Intensitas Cahaya

Pengujian intensitas cahaya pada webcam dengan model YOLOv5 dilakukan untuk memastikan pada tingkat intensitas cahaya berapa kamera dan model dapat mendeteksi objek dengan baik dan optimal. Pengujian dilakukan dengan mengambil gambar tanaman cabai dengan kondisi intensitas cahaya yang berbeda, menggunakan lux meter.

Tabel 5. Pengujian Webcam dengan Model YOLOv5 Berdasarkan Intensitas Cahaya

Percobaan Ke-	Intensitas Cahaya (lux)	Gambar	Hasil
1	387		Terdeteksi (0.65)
2	1031		Terdeteksi (0.66)
3	2783		Terdeteksi (0.89)
4	5231		Terdeteksi (0.79)
5	6284		Terdeteksi (0.92)

Pengujian webcam dengan model YOLOv5 berdasarkan intensitas cahaya menunjukkan bahwa deteksi objek meningkat seiring dengan bertambahnya tingkat pencahayaan. Pada intensitas cahaya 387 lux, akurasi deteksi adalah 0.65, meningkat menjadi 0.66 pada 1031 lux, dan mencapai 0.89 pada 2783 lux. Pada intensitas 5231 lux, akurasi sedikit menurun menjadi 0.79, namun kembali optimal dengan akurasi 0.92 pada intensitas cahaya tertinggi, yaitu 6284 lux. Hasil ini menunjukkan bahwa model YOLOv5 lebih efektif dalam mendeteksi objek pada

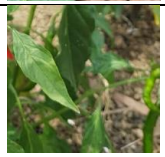
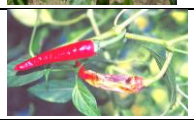
https://doi.org/10.25077/chipset.7.02.132-139.2026

pencahayaan yang lebih terang, meskipun intensitas cahaya yang sangat tinggi dapat mempengaruhi kualitas deteksi.

Pengujian Model YOLOv5

Pengujian ini bertujuan untuk mengevaluasi performa model YOLOv5 dalam mendeteksi objek penyakit pada tanaman cabai. Pengujian dilakukan dengan uji coba pada objek dengan menggunakan 4 sampel citra dari setiap kelas yang ada pada model. Total uji coba pada pengujian ini sebanyak 24 kali.

Tabel 6. Pengujian Model YOLOv5

No	Data Uji	Gambar	Hasil Prediksi (nilai akurasi)	Keterangan
1	Cabai Sehat		Sehat (0.85)	True Positive
2	Cabai Sehat		Sehat (0.85)	True Positive
3	Cabai Sehat		Sehat (0.87)	True Positive
4	Cabai Sehat		Sehat (0.76)	True Positive
5	Daun Berkerut		Daun Berkerut (0.82)	True Positive
6	Daun Berkerut		0	False Negative
7	Daun Berkerut		Daun Berkerut (0.82)	True Positive
8	Daun Berkerut		Daun Berkerut (0.8)	True Positive
9	Antachnose		Antachnose (0.76)	True Positive
10	Antachnose		Antachnose (0.70)	True Positive
11	Antachnose		Antachnose (0.75)	True Positive
12	Antachnose		Antachnose (0.79)	True Positive
13	Kutu Putih		Kutu Putih (0.71)	True Positive
14	Kutu Putih		Kutu Putih (0.91)	True Positive
15	Kutu Putih		Kutu Putih (0.84)	True Positive
16	Kutu Putih		Kutu Putih (0.88)	True Positive
17	Leaf Spot		Yellowish (0.81)	False Positive
18	Leaf Spot		Leaf Spot (0.84)	True Positive
19	Leaf Spot		Leaf Spot (0.92)	True Positive
20	Leaf Spot		Leaf Spot (0.81)	True Positive
21	Yellowish		0	False Negatif

22	Yellowish		Yellowish (0.88)	True Positive
23	Yellowish		Yellowish (0.79)	True Positive
24	Yellowish		Yellowish (0.91)	True Positive

Berdasarkan confussion mattrix dan hasil pengujian model, diketahui bahwa terdapat data uji yang terbaca dengan akurasi deteksi rendah bahkan tidak terdeteksi. Hal ini terjadi oleh beberapa faktor, yaitu kurangnya ragam atau variasi dataset saat training, sehigga ketika objek deteksi dilakukan dengan jenis atau bentuk yang berbeda dicoba, maka deteksi akan menjadi rendah. Namun secara keseluruhan, model sudah mampu mendeteksi objek penyakit pada tanaman cabai dengan cukup baik.

KESIMPULAN

Berdasarkan rancangan, implementasi, dan pengujian sistem yang telah dilakukan, dapat disimpulkan bahwa alat ini berhasil memenuhi tujuan penelitian. Alat mampu menangkap gambar dengan resolusi 720p dalam rentang jarak 5-25 cm, serta melakukan deteksi objek penyakit pada tanaman cabai dengan akurasi mencapai 87,5%. Selain itu, alat ini efektif dalam mendeteksi nilai pH tanah, dengan rata-rata selisih perbandingan sebesar 0,28 dibandingkan dengan pH tester. Proses pengukuran pH membutuhkan waktu sekitar satu menit setelah sensor ditancapkan untuk menyesuaikan dengan lingkungan. Alat juga mampu membaca kelembaban tanah dengan akurasi 98%, dengan selisih rata-rata perbandingan sebesar 2% dibandingkan dengan soil moisture tester. Selanjutnya, alat memberikan rekomendasi pengendalian yang tepat sesuai dengan penyakit dan kondisi tanah yang terdeteksi. Terakhir, output ditampilkan pada LCD dengan kontras yang jelas dan dapat dibaca oleh pengguna. Temuan ini berkontribusi pada pengembangan teknologi pertanian yang lebih efektif dalam pemantauan kesehatan tanaman dan kondisi tanah...

REFERENSI

[1] G. N. C. Tuhumury dan H. R. D. Amanupunyo, “KERUSAKAN TANAMAN CABAI AKIBAT PENYAKIT VIRUS DI DESA WAIMITAL KECAMATAN KAIRATU,” 2013.

[2] C. Prihatiningrum, A. F. Nafi’udin, dan M. Habibullah, “Identifikasi Teknik Pengendalian Hama Penyakit Tanaman Cabai di Desa,” vol. 18, 2021.

[3] Arief Muhammad Siregar Rahmadsah, “PENGGUNAAN TEKNOLOGI DRONE DALAM MONITORING,” 2022.

[4] F. Zikra, K. Usman, dan R. Patmasari, “Deteksi Penyakit Cabai Berdasarkan Citra Daun Menggunakan Metode Gray Level Co-Occurence Matrix Dan Support Vector Machine,” 2021. [5] L. Setiana Riva, “Deteksi Penyakit Tanaman Cabai Menggunakan Algoritma YOLOv5 Dengan Variasi Pembagian Data,” vol. 8, no. 3, 2023.

[5] N. Mukhayat, P. W. Ciptadi, dan R. H. Hardyanto, “Sistem Monitoring pH Tanah, Intensitas Cahaya Dan Kelembaban Pada Tanaman Cabai (Smart Garden) Berbasis IoT,” 2019.

[6] H. Yusuf, H. Fitriyah, dan S. R. Akbar, “Deteksi Penyakit pada Daun Jagung berdasarkan Hue menggunakan Metode K-Nearest Neighbor berbasis Raspberry Pi 4,” 2022. [Daring]. Tersedia pada: http://j-ptiik.ub.ac.id

[7] S. S. Zuain, H. Fitriyah, dan R. Maulana, “Deteksi Penyakit pada Daun Cabai berdasarkan Fitur HSV dan GLCM menggunakan Algoritma C4.5 berbasis Raspberry Pi 4,” 2021. [Daring]. Tersedia pada: http://j-ptiik.ub.ac.id

[8] Putra, A. D., Erlina, T., & Putri, R. E. (2021). "Sistem Cerdas Berbasis Internet of Things (IoT) untuk Pemantauan Lingkungan Pertanian." Journal of Information Technology and Computer Engineering (JITCE), 5(1), 20-27.

[9] Yendri, D., & Hersyah, M. H. (2021). "Desain Sistem Otomatisasi Irigasi Tetes Berbasis Mikrokontroler dan Sensor Kelembaban Tanah." Jurnal Nasional Teknik Elektro (JNTE), 10(2), 115-122.

[10] Maulana, R., Rahmadya, B., & Suryani, A. I. (2022). "Penerapan Single Board Computer Raspberry Pi pada Sistem Deteksi Objek Real-Time." CHIPSET: Journal on Computer Hardware, Signal Processing, Embedded System and Networking, 3(1), 30-38.

[11] Sani, A. A., Budiarti, R., & Suwandi, R. (2020). "Evaluasi Kinerja Sensor pH dan Kelembaban untuk Sistem Pertanian Presisi." Jurnal Nasional Teknik Elektro (JNTE), 9(1), 45-52.

[12] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). "You only look once: Unified, real-time object detection." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779-788.

[13] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). "YOLOv4: Optimal Speed and Accuracy of Object Detection." arXiv preprint arXiv:2004.10934.

[14] Jocher, G., et al. (2022). "ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation." Zenodo. https://doi.org/10.5281/zenodo.7347926.

AUTHOR(S) BIOGRAPHY



Alhafiz Zhurrahman

Born in Pariaman, Sumatera Barat, in 2002. He attended MAN Insan Cendekia Padang Pariaman and is currently studying at the Department of Computer Engineering, Faculty of Information Technology, Andalas University.

 https://doi.org/10.25077/chipset.7.02.132-139.2026