



## SISTEM PENGENALAN BENIH PADI MENGGUNAKAN METODE LIGHT CONVOLUTIONAL NEURAL NETWORK PADA RASPBERRY PI 4 B

Indra Hermawan<sup>1</sup>, Defiana Arnaldy<sup>2</sup>, Maria Agustin<sup>3</sup>, M. Farishanif Widyono<sup>4</sup>, David Nathanael<sup>5</sup>, Meutia Tri Mulyani<sup>6</sup>

<sup>1,2,3,4,5</sup> Teknik Multimedia dan Jaringan, Politeknik Negeri Jakarta

<sup>6</sup> Teknik Informatika dan Komputer, Politeknik Negeri Jakarta

Depok, Jawa Barat, Indonesia

indra.hermawan@tik.pnj.ac.id, defiana.arnaldy@tik.pnj.ac.id, maria.agustin@tik.pnj.ac.id,  
muhammad.farishanifwidyono.tik18@mhs.w.pnj.ac.id, david.nathanael.tik19@mhs.w.pnj.ac.id,  
meutia.trimulyani.tik19@mhs.w.pnj.ac.id

### Abstract

*Recently, Deep learning methods with Convolutional Neural Networks (CNNs) have been widely used for image classification tasks. CNN has an unrivaled advantage in extracting discriminatory image features. However, many existing CNN-based methods are designed to go deeper and more significant with more complex layers that make them challenging to implement on mobile devices or real-time devices that use microcontrollers like raspberry pi, Arduino, and immediately. This is overcome by using a Light Convolutional Neural Network (LCNN), so it needs to experiment to test the difference in LCNN performance on a personal computer and a raspberry pi four microcontrollers with a Raspbian operating system. Experiments will be carried out using several performance measures: accuracy, F-1 score, recall, precision, and time to get performance results from deep learning. As such, the results and model architecture will confirm performance differences across individual devices and show how the model performs on resource-constrained or real-time devices. Tests show that the performance of the raspberry pi, which is a tool with limited resources, does not affect the quality of image recognition but affects the recognition processing time because the raspberry pi requires a longer processing time to perform one data or photo recognition process. This will accumulate the time required for processing many data, so it can conclude that the raspberry pi and tools with limited resources are not very practical for conducting recognition training and carrying out a recognition process that contains a lot of data or photos in one process.*

**Keywords:** Raspberry Pi, Deep learning, Artificial Neural Network, LCNN, Paddy

### Abstrak

Baru-baru ini, metode pembelajaran mendalam dengan *Convolution Neural Network* (CNN) telah banyak digunakan untuk tugas klasifikasi gambar. CNN memiliki keunggulan yang tak tertandingi dalam mengekstraksi fitur gambar diskriminatif. Namun, banyak metode berbasis CNN yang ada dirancang untuk lebih dalam dan lebih besar dengan lapisan yang lebih kompleks. Sehingga membuatnya sulit untuk diterapkan pada perangkat seluler atau pada perangkat waktu nyata yang menggunakan mikrokontroler seperti raspberry pi, Arduino, dan lain sebagainya. Hal tersebut diatasi dengan menggunakan *Light Convolution Neural Network* (LCNN), maka perlu dilakukan percobaan untuk menguji seberapa besar perbedaan kinerja LCNN pada *Personal Computer* (PC) dan pada mikrokontroler raspberry pi 4 dengan sistem operasi Raspbian. Eksperimen dilakukan dengan menggunakan beberapa parameter kinerja yaitu *accuracy*, *F-1 Score*, *recall*, *precision*, dan waktu dari pengujian klasifikasi untuk mendapatkan hasil performa dari pembelajaran mendalam. Oleh karena itu, hasil dan arsitektur model akan mengkonfirmasi perbedaan kinerja di masing-masing perangkat dan menunjukkan bagaimana performa model pada perangkat yang dibatasi sumber daya atau berjalan secara waktu nyata. Pengujian menunjukkan bahwa kinerja pada raspberry pi yang merupakan alat dengan sumber daya terbatas tidak mempengaruhi kualitas pengenalan gambar, tetapi mempengaruhi waktu pemrosesan pengenalan, dikarenakan raspberry pi membutuhkan waktu proses yang lebih lama untuk melakukan satu proses pengenalan data atau foto. Hal tersebut akan mengakumulasi waktu yang dibutuhkan untuk pemrosesan data yang banyak, sehingga dapat disimpulkan bahwa raspberry pi dan alat dengan sumber daya terbatas sangat tidak efektif untuk melakukan pelatihan pengenalan dan melakukan proses pengenalan yang berisi banyak data atau foto dalam sekali prosesnya.

**Kata kunci:** Raspberry Pi, Pembelajaran Mendalam, Jaringan Saraf Tiruan, LCNN, Padi

## 1. PENDAHULUAN

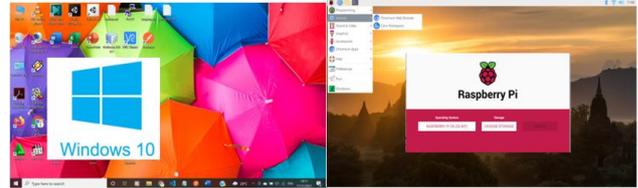
*Internet of Things* (IoT), yang juga dikenal sebagai *Internet of Everything* atau Industrial Internet, merupakan pandangan pengembangan teknologi baru yang merupakan sebuah jaringan mesin global dan perangkat yang mampu berinteraksi satu sama lain [1]. IoT diakui sebagai salah satu bidang yang paling penting dari teknologi masa depan dan mendapatkan perhatian yang luas dari berbagai industri. Pengembangan pada sistem IoT ini juga semakin banyak dikembangkan dengan pembelajaran mesin yang juga semakin berkembang menjadi *deep learning*.

Sebagian besar model pembelajaran mendalam modern didasarkan pada jaringan saraf tiruan, khususnya *Convolution Neural Network* (CNN), meskipun mereka juga dapat menyertakan rumus proposisional atau variabel laten yang diatur secara berlapis pada model generatif yang dalam. Pembelajaran mendalam memungkinkan model komputasi yang terdiri dari beberapa lapisan pemrosesan untuk mempelajari representasi data dengan berbagai tingkat abstraksi. Metode-metode tersebut telah meningkatkan pengetahuan dan pandangan terhadap pengembangan teknologi dalam pengenalan suara, pengenalan objek visual, deteksi objek, penyaringan, terjemahan mesin, bioinformatika dan banyak domain lainnya seperti penemuan obat dan genomik, di mana mereka memberikan hasil yang sebanding dan dalam beberapa kasus lebih unggul dari hasil manusia berpengalaman [2].

Pembelajaran mendalam menemukan struktur rumit dalam kumpulan data besar dengan menggunakan algoritma backpropagation untuk menunjukkan bagaimana mesin harus mengubah parameter internalnya yang digunakan untuk menghitung representasi di setiap lapisan dari representasi di lapisan sebelumnya. Dalam pembelajaran mendalam, setiap level belajar untuk mengubah data inputnya menjadi representasi yang sedikit lebih abstrak dan komposit. Salah satu pembelajaran mendalam yang sudah banyak diteliti ada jaringan saraf tiruan konvolusional digunakan untuk memproses data-data kimia [3][4]. Dalam aplikasi pengenalan gambar, input mentah dapat berupa matriks piksel, lapisan representasional pertama dapat mengabstraksi piksel dan mengkodekan tepi, lapisan kedua dapat menyusun dan mengkodekan pengaturan tepi, lapisan ketiga dapat mengkodekan bentuk dan warna, dan lapisan keempat dapat mengenali bahwa gambar tersebut berupa padi. Proses pembelajaran yang mendalam dapat mempelajari fitur mana yang ditempatkan secara optimal di levelnya. Ini tidak sepenuhnya menghilangkan kebutuhan untuk penyetelan secara manual misalnya, jumlah lapisan dan ukuran lapisan yang bervariasi dapat memberikan tingkat abstraksi yang berbeda. Juga dapat disertakan rumus proposisional atau variabel laten yang diatur secara berlapis dalam model generatif yang dalam.

Maka untuk dapat melihat seberapa baik kinerja pembelajaran mendalam pada sebuah perangkat IoT

dilakukan perbandingan kinerja pada alat PC dengan sistem operasi windows 10 dengan alat IoT, yaitu mikrokontroller Raspberry pi 4 dengan sistem operasi raspbianOS. RaspbianOS merupakan system operasi berbasis kernel linux, dengan Noobs sebagai manajer instalasi [5]. Tampilan sistem operasi dari kedua alat dapat dilihat pada Gambar 1.



Gambar 1. Sistem Operasi Windows dan Raspbian

Pada penelitian ini dilakukan perbandingan kinerja sebuah arsitektur pembelajaran mendalam pada kedua alat menggunakan algoritma berdasarkan LCNN yang diimplementasikan menggunakan python, torch, dan *Open Source Computer Vision Library* (OpenCV) untuk melakukan kegiatan klasifikasi gambarnya. Torch adalah kerangka kerja komputasi numerik serbaguna dan perpustakaan pembelajaran mesin. Jika dibandingkan dengan *framework* lain seperti Tensorflow, Torch memiliki keunggulan berupa kesederhanaan dalam pembuatannya [6]. OpenCV merupakan perangkat lunak berlisensi BSD sehingga dapat digunakan untuk kepentingan akademis atau profesional secara gratis. OpenCV dapat digunakan pada Bahasa pemrograman C, C++, Python dan banyak lagi. OpenCv dirancang untuk melakukan komputasi *real time*, OpenCv dapat memaksimalkan penggunaan beberapa *core* dan dapat memaksimalkan akselerasi perangkat keras [7].

Tujuan dilakukannya perbandingan kinerja pada dua perangkat yang berbeda adalah untuk menyediakan lingkungan yang fleksibel untuk merancang dan melatih pembelajaran mesin. Fleksibilitas diperoleh melalui Lua, bahasa skrip yang sangat ringan. Performa tinggi diperoleh melalui OpenMP/SSE yang efisien dan Implementasi CUDA dari rutinitas numerik tingkat rendah [8].

Uji coba arsitektur pembelajaran mendalam dilakukan pada microcontroller Raspberry Pi 4 Model B untuk menangani hal ini. Raspberry Pi 4 Model B adalah produk terbaru dari seri Raspberry Pi 4, pertama kali diperkenalkan pada tahun 2019, diperdaya dengan prosesor 64-bit quadcore Arm Cortex-72 berkecepatan 1.5GHz, WLAN dual-band 2.4GHz dan 5GHz, Bluetooth 5.0/BLE, Gigabit ethernet, dan PoE (*Power Over Ethernet*) [9]. Jejak mekanis Raspberry Pi 4 Model B masih sama seperti Raspberry Pi 3 Model B+ maupun Raspberry Pi 3 Model B. Kemudahan pembuatan kode, banyaknya sensor dan modul yang ada di pasaran, serta besarnya kemampuan *board* ini untuk dimodifikasi membuat Raspberry Pi 4 B+ menjadi mikrokontroler yang penulis pilih untuk menjadi alat IoT yang akan diteliti.

Dengan meningkatnya penelitian pembelajaran mesin dan pembelajaran mendalam dalam pengembangan teknologi pengenalan suara, pengenalan objek visual, yang memanfaatkan peralatan IoT semakin banyak [10]. Maka semakin dibutuhkannya informasi performa pembelajaran mendalam pada alat IoT beberapa penelitian terkait seperti penelitian pembelajaran mesin pada alat-alat penerima akhir [11] atau penelitian di dalam pembelajaran mesin itu sendiri [8][12]. Maka penulis melakukan penelitian performa pembelajaran mendalam dengan alat IoT yang saat ini sedang populer yaitu, raspberry pi 4 model B. Tampilan fisik raspberry pi 4 model B dapat dilihat pada Gambar 2.



Gambar 2. Raspberry pi 4 model B

Selanjutnya pada bab metode penelitian dijelaskan kegiatan dan proses dalam perealisasiian pengujian dan penelitian, dengan dilakukannya studi literatur dan penelitian performa pada masing-masing alat, pembuatan pembelajaran mendalam, dan pelatihatannya serta menentukan parameter-parameter yang ditentukan dalam pengujian [13]. Pada bab hasil dan pembahasan menerangkan mengenai hasil dari pengujian dan fakta performa masing-masing alat sesuai dengan parameter-parameter yang diteliti. Pada bab kesimpulan dan saran dijelaskan kesimpulan sesuai hasil pengujian yang didapatkan.

## 2. METODE PENELITIAN

Penulis menerapkan sebuah arsitektur pembelajaran mendalam yang memanfaatkan algoritma jaringan saraf konvolusional ringan. Dengan tujuan untuk dapat mendapatkan kinerja pembelajaran yang baik tanpa memakan banyak sumber daya pada alat yang kedepannya dapat dikembagkan kembali dengan penambahan sensor-sensor IoT yang membutuhkan kinerja yang baik untuk alat yang bekerja secara waktu nyata.

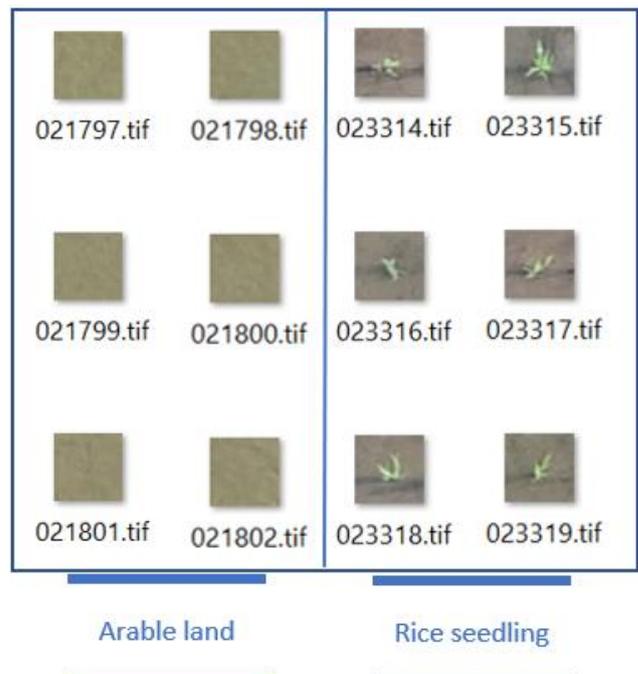
### 2.1. Dataset

Pembelajaran mendalam yang dilakukan oleh penulis menggunakan *dataset* benih padi yang memiliki jumlah sampel sebanyak 54.628 yang nantinya akan dibagi menjadi

tiga set, yaitu set pelatihan sebanyak 43.703, set pengujian sebanyak 1.093, dan set validasi sebanyak 9.832 [14].

Benih padi dipilih sebagai objek penelitian karena salah satu komponen penting dalam produksi padi [15]. Padi sendiri merupakan komoditas pangan utama di Indonesia, tingkat produksi ataupun konsumsi padi selalu menempati urutan pertama [13]. Oleh karena itu menjaga tumbuh kembang benih padi perlu dilakukan. Pendeteksian dan pengklasifikasian benih padi dengan objek selain benih padi (gulma) bisa menjadi Langkah antisipasi untuk menjaga Kesehatan benih padi. Gulma yang menyerang area pertanian dapat menurunkan produktivitas hasil panen sebesar 17% [16].

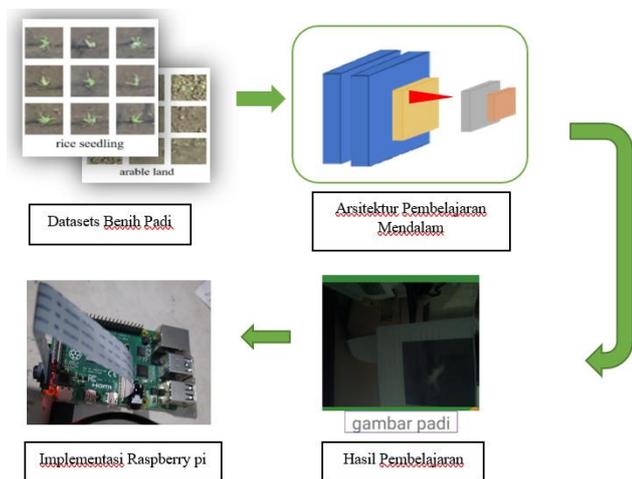
Penulis menggunakan format gambar berjenis Tiff. Tiff merupakan format gambar yang sangat fleksibel, pada umumnya tiff digunakan untuk menyimpan gambar lossless, itu artinya tidak ada kompresi dalam format tiff sehingga menghasilkan gambar yang memiliki detail yang tinggi [17]. Tampilan *dataset* yang digunakan penulis ditunjukkan oleh Gambar 3.



Gambar 3. Dataset pembelajaran mendalam

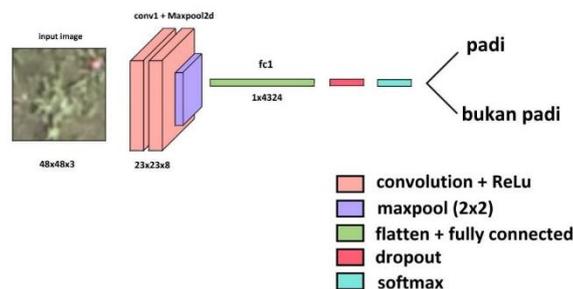
### 2.2. Tahapan Penelitian

Untuk dapat melakukan pengujiannya pada alat akhir dilakukan penelitian mengenai pemahaman sistem kerja komputer personal dan kemampuan alat IoT yang akan digunakan untuk penelitian [18], serta studi mengenai arsitektur, algoritma, peningkatan kinerja, dan pengujian pada model pembelajaran dengan cara mencari berbagai sumber pustaka, referensi dan berbagai jurnal penelitian. Diagram penelitian pembelajaran mendalam dapat dilihat pada Gambar 4.



Gambar 4. Diagram penelitian pembelajaran mendalam

Penulis menerapkan sebuah arsitektur jaringan saraf konvolusional ringan sederhana yang memiliki kemampuan pembelajaran mendalam yang sudah cukup baik dengan melakukan proses pelatihan pembelajaran mendalam yang penulis buat akan dilakukan pada komputer personal dengan sistem operasi Windows 10 dan dengan memanfaatkan dataset pengujian mesin yang telah digunakan pada salah satu penelitian pembelajaran mendalam lain [19]. Tampilan arsitektur pembelajaran mendalam ditunjukkan oleh Gambar 5.



Gambar 5. Arsitektur pembelajaran mendalam pada penelitian

Arsitektur yang akan digunakan adalah arsitektur LCNN dengan satu lapisan konvolusi, satu lapisan *max pooling* dan satu lapisan tersambung penuh. Pembelajaran dilakukan dengan melakukan penyimpanan bobot dengan hasil pembelajaran terbaik dengan mengatur beberapa *hyper parameter* yang dimiliki pembelajaran mendalam, seperti berapa kali pelatihan (*epoch*), kecepatan belajar (*learning rate*), algoritma pengoptimalisasian (*optimizer*), algoritma kriteria pengambilan bobot (*criterion*) dan ukuran pengambilan gambar (*batch size*). Dengan melakukan berbagai percobaan dengan *hyper parameter* yang ada, didapatkan pembelajaran mendalam dengan pembacaan yang memiliki akurasi sebesar 96%. Untuk mendapatkan pembacaan tersebut bobot akan disimpan dengan memanfaatkan fungsi dari perpustakaan torch, yaitu *save* untuk menyimpan data bobot ke dalam file yang nantinya bisa diberikan ke dalam arsitektur pembelajaran mendalam yang sama.

### 2.3. Perangkat Pengujian

Perangkat yang digunakan untuk pengujian adalah PC dan perangkat raspberry pi 4 model B. Spesifikasi perangkat keras yang digunakan pada pengujian ditunjukkan pada Tabel 1.

Tabel 1. Spesifikasi perangkat perangkat keras untuk pengujian

Alat	CPU	Frekuensi	Cores	Memori	OS
Windows 10	AMD Ryzen 5 4500U	2.38 GHz	6	8Gb	Windows 10 Home 21H1
Raspberry pi 4	Cortex-A72	1.5 GHz	4	4Gb	Raspbian OS 5.10

### 2.4. Evaluasi Kinerja

Pengujian dilakukan dengan memanfaatkan bobot simpanan hasil pembelajaran sebelumnya dengan memanfaatkan fungsi dari perpustakaan torch, yaitu *load* untuk mendapatkan bobot dari file yang menyimpan bobotnya. Kemudian penulis akan memberikan kedua alat pembelajaran mendalam dengan bobot dan arsitektur yang sama. Kode pengujian yang dibuat akan dijalankan di perangkat lunak jupyter notebook. Penulis memanfaatkan perangkat lunak jupyter notebook untuk membuat lingkungan kode pengujian yang sama pada masing-masing alat. kode pengujian melakukan perhitungan untuk mendapatkan beberapa parameter-parameter yang akan ditinjau yaitu, *accuracy*, skor F-1, *recall*, *precision*, dan waktu proses untuk akhirnya menentukan performa dari pembelajaran mendalam yang berjalan [20].

#### 1. Precision

*Precision* adalah prediksi rasio dari pengidentifikasian yang benar dengan jumlah pengidentifikasian yang di prediksi benar pada kelas yang spesifik. Presisi yang rendah mengindikasikan banyaknya jumlah kesalahan pengidentifikasian yang dinyatakan benar. *Precision* dapat direpresentasikan sebagaimana yang ditunjukkan pada rumus (1).

$$precision = \frac{TP}{TP + FP} \tag{1}$$

dimana TP = *true positive*, FP = *false positive*.

#### 2. Recall

*Recall* merupakan rasio prediksi dari pengidentifikasian yang benar dengan jumlah pengidentifikasian yang sesungguhnya memang benar. *Recall* yang tinggi mengindikasikan sedikit jumlah sampel yang diidentifikasi salah. *Recall* dapat direpresentasikan sebagaimana yang ditunjukkan pada rumus (2).

$$recall = \frac{TP}{TP + FN} \tag{2}$$

dimana TP = true positive, FN = false negative.

### 3. Accuracy

Accuracy merupakan rasio prediksi benar (berhasil dan gagal) dari keseluruhan data yang didapatkan, dan dapat dibuat formula yaitu jumlah dari klasifikasi benar dibagi dengan jumlah seluruh klasifikasi seharusnya. Akurasi dapat direpresentasikan sebagaimana yang ditunjukkan pada rumus (3).

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{3}$$

dimana TP = true positive, TN = true negative, FP = false positive, dan FN = false negative.

### 4. F1-Score

F1-Score mengkuantifikasi rata-rata harmonik antara presisi dan akurasi. Metrik ini biasanya menyajikan kekokohan tugas klasifikasi, dan dapat dihitung dengan menggunakan rumus (4).

$$F1 = \frac{2 * precision * recall}{precision + recall} \tag{4}$$

### 5. Waktu

Waktu digunakan untuk melihat kecepatan performa pembelajaran mendalam berjalan disetiap pembelajarannya dengan mengambil data waktu mulainya pendalaman belajar dan waktu selesainya pembelajaran mendalam.

## 3. HASIL DAN PEMBAHASAN

Pada kedua alat dilakukan pengujian pembelajaran mendalam dengan menggunakan cpu untuk menjalankan proses penentuannya. Kedua alat memiliki program pengujian yang sama dengan arsitektur dan bobot yang pembelajaran mendalam yang sama, serta kedua pengujian dilakukan perangkat lunak bantuan yang sama, yaitu dengan jupyter notebook. Lingkungan pengujian kedua alat disamakan untuk memberikan hasil yang kompeten. Lingkungan pengujian dapat dilihat pada Gambar 6.

```

n_epochs = 10
test_acc = []
val_precision = []
val_f1 = []
val_recall = []
val_acc = []
val_kappa = []
test_time = 0.0

acc_chart = []
precision_chart = []
recall_chart = []
kappa_chart = []

for epoch in range(n_epochs):
    model.eval()
    start_test = time.time()
    for _, _ in enumerate(test_loader, 0):
        data_test, target_test = next(iter(test_loader))

        start_test_model_only = time.time()

        with torch.no_grad():
            output = model(data_test)

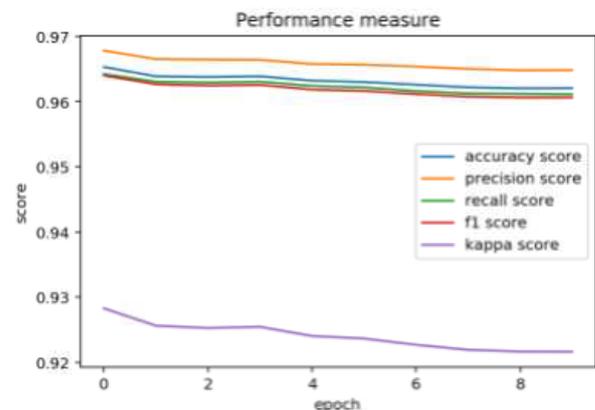
        end_test_model_only = time.time()

        test_time += (end_test_model_only - start_test_model_only)

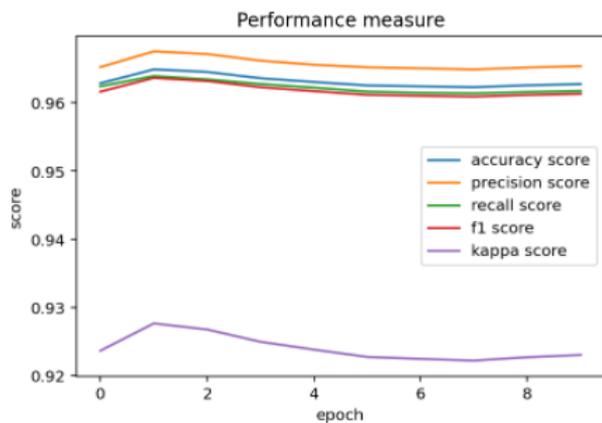
    ps = torch.exp(output)
    top_p, top_class = ps.topk(1, dim=1)
    preds = top_class.squeeze(dim=1)
    targets = target_test
    
```

Gambar 6. Tampilan Lingkungan Pengujian Jupyter Notebook

Dari hasil sepuluh kali pengujian telah didapatkan bahwa performa kerja perangkat keras tidak mempengaruhi hasil pengujian pengenalan gambar dari pembelajaran mendalam, dan lebih dipengaruhi dari ukuran dan banyaknya gambar atau data tes yang digunakan. Hal tersebut dikarenakan terbatasnya sumber daya tidak mempengaruhi kualitas pengenalan, dan telah dibuktikan dengan mengamati rata-rata hasil accuracy, precision, recall, skor kappa dan f1-score yang tidak terlalu jauh berbeda. Hasil dari pengujian accuracy, precision, recall, skor kappa dan f1-score dapat dilihat pada grafiknya yang dapat dilihat dalam Gambar 7 dan Gambar 8.



Gambar 7. Grafik Akurasi, Presisi, Recall, Skor Kappa, dan Skor F-1 pada Windows 10



**Gambar 8.** Grafik Akurasi, Presisi, *Recall*, Skor Kappa, dan Skor F-1 pada Raspberry Pi 4

Tetapi pada kinerja pengolahan banyaknya data dan ukuran gambar sangat mempengaruhi kecepatan pengujian pada raspberry pi 4 yang memiliki sumber daya yang terbatas, pengolahan gambar tes yang berisi dari 9.832 file gambar membuat raspberry pi 4 membutuhkan waktu yang cukup lama dalam melakukan pengujian pembelajaran mendalam. Hasil dari pengujian kecepatan dapat dilihat pada Tabel 2.

Hasil data berikut terjadi dikarenakan sumber daya raspberry pi yang terbatas membuat kemampuan pengolahan data yang terbatas, sehingga terlihat bahwa raspberry pi membutuhkan waktu pemrosesan yang cukup lama dikarenakan banyaknya data yang perlu dilakukan pengujian. Karena sumber daya komputasinya yang terbatas, seperti kemampuan cpu banyaknya *cores* dan besarnya memori lah yang mempengaruhi proses pengenalan gambar.

**Tabel 2.** Pengujian Kinerja Waktu Load Data

Pengujian Waktu	Windows 10	Raspberry Pi
1.	25s	2m 28s
2.	12s	2m 53s
3.	12s	3m 13s
4.	12s	2m 46s
5.	12s	2m 56s
6.	12s	2m 57s
7.	13s	2m 55s
8.	12s	2m 50s
9.	13s	2m 51s
10.	13s	2m 51s

#### 4. KESIMPULAN

Pada penelitian ini dilakukan pengujian pengenalan gambar pada kedua alat pc dengan sistem operasi windows 10 dan raspberry pi 4 dengan pembelajaran mendalam yang sama untuk melakukan perbandingan performa kinerja pada masing-masing alat. Didapatkan bahwa perangkat keras/alat tidak mempengaruhi performa ke akuratan atau hasil

kualitas pengenalan gambar, tetapi mempengaruhi kecepatan performa pengolahan data, ditunjukkan dengan lamanya waktu yang dibutuhkan raspberry pi saat mengolah banyaknya data gambar yang akan diuji dari banyaknya data yang disiapkan. Hal tersebut menunjukkan bahwa hanya kemampuan komputasi alat yang mempengaruhi proses pengenalan gambar.

#### DAFTAR PUSTAKA

- [1] I. Lee and K. Lee, "The Internet of Things (IoT): Applications, investments, and challenges for enterprises," *Bus. Horiz.*, vol. 58, no. 4, pp. 431–440, 2015, doi: 10.1016/j.bushor.2015.03.008.
- [2] H. A. Shiddieqy, F. I. Hariadi, and T. Adiono, "Implementation of deep-learning based image classification on single board computer," *2017 Int. Symp. Electron. Smart Devices, ISESD 2017*, vol. 2018-Janua, pp. 133–137, 2017, doi: 10.1109/ISESD.2017.8253319.
- [3] N. Yanti and M. Ulfah, "Aplikasi Jaringan Syaraf Tiruan Untuk Clustering Polutan Kimia Penyebab Pencemaran Udara," *JTT (Jurnal Teknol. Terpadu)*, vol. 3, no. 2, pp. 68–74, 2015, doi: 10.32487/jtt.v3i2.82.
- [4] N. Yanti and F. Z. Rachman, "Neural Network Application For The Analysis Of The Nutrition And Environment Effect To Microbial Growth Rate On Fermented Soybean Patty (Tempe) Fermentation," *JTT (Jurnal Teknol. Terpadu)*, vol. 4, no. 2, p. 100, 2016, doi: 10.32487/jtt.v4i2.171.
- [5] M. Nayak and P. Dash, "Original Research Paper Smart Surveillance Monitoring System," no. June, pp. 16–19, 2018.
- [6] L. Jiang and Z. Zhang, "Research on image classification algorithm based on pytorch," *J. Phys. Conf. Ser.*, vol. 2010, no. 1, 2021, doi: 10.1088/1742-6596/2010/1/012009.
- [7] L. Enciso, R. Yaguana, R. Jaramillo, and E. Zelaya-Policarpo, "IoT in services for communities in the recognition of plates with OpenCV-python," *WEBIST 2018 - Proc. 14th Int. Conf. Web Inf. Syst. Technol.*, no. Webist, pp. 430–437, 2018, doi: 10.5220/0007235004300437.
- [8] V. Kovalev, A. Kalinovskiy, and S. Kovalev, "Deep Learning with Theano, Torch, Caffe, TensorFlow, and Deeplearning4J: Which One Is the Best in Speed and Accuracy?," *Proc. 13th Int. Conf. Pattern Recognit. Inf. Process. (PRIP 2016)*, vol. 1, no. July 2017, pp. 99–103, 2016.
- [9] S. Dokrimare, "IoT based Smart Mirror using Raspberry Pi," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 7, no. 4, pp. 1431–1434, 2019, doi: 10.22214/ijraset.2019.4257.
- [10] B. Balon and M. Simic, "Using Raspberry Pi Computers in Education," *2019 42nd Int. Conv. Inf. Commun. Technol. Electron. Microelectron.*, no. March 2018, pp. 671–676, 2019, doi: 10.23919/mipro.2019.8756967.
- [11] X. Zhang, Y. Wang, and W. Shi, "PCamp: Performance comparison of machine learning

- packages on the edges,” *USENIX Work. Hot Top. Edge Comput. HotEdge 2018, co-located with USENIX ATC 2018*, vol. 2, no. 1, 2018.
- [12] D. Fitriati, “Perbandingan Kinerja CNN LeNet 5 dan Extreme Learning Machine pada Pengenalan Citra Tulisan Tangan Angka,” *J. Teknol. Terpadu*, vol. 2, no. 1, pp. 10–16, 2016.
- [13] T. W. Saleh, N. Buri, and A. A. Saragih, “Keragaan Hama, Penyakit Dan Musuh Alami Pada Budidaya Beberapa Varietas Padi Gogo Di Lahan Sawah,” *Researchgate.Net*, no. 270, pp. 163–170.
- [14] T. Yu and H. Zhu, “Hyper-Parameter Optimization: A Review of Algorithms and Applications,” pp. 1–56, 2020.
- [15] J. Firdaus and U. Ahmad, “Deteksi Benih Varietas Padi Menggunakan Gelombang Near Infrared dan Model Jaringan Saraf Tiruan Detection of Rice Seed Varieties Using Near Infrared Spectroscopy,” 2017.
- [16] S. Syarifah, I. Apriani, and R. H. T. Amallia, “IDENTIFIKASI GULMA TANAMAN PADI (*Oryza sativa* L. var. Ciherang) SUMATARA SELATAN,” *J. Biosilampari J. Biol.*, vol. 1, no. 1, pp. 40–44, 2018, doi: 10.31540/biosilampari.v1i1.52.
- [17] M. K. Linga Murthy, D. Bhargavi, V. Nagamani, S. Divya, and P. Jagadeesh, “Novel approach for image cryptography,” *J. Adv. Res. Dyn. Control Syst.*, vol. 12, no. 2, pp. 721–726, 2020, doi: 10.5373/JARDCS/V12I2/S20201088.
- [18] D. J. Norris, *Machine Learning with the Raspberry Pi: Experiments with Data and Computer Vision*. 2020.
- [19] M. Der Yang, H. H. Tseng, Y. C. Hsu, C. Y. Yang, M. H. Lai, and D. H. Wu, “A UAV open dataset of rice paddies for deep learning practice,” *Remote Sens.*, vol. 13, no. 7, 2021, doi: 10.3390/rs13071358.
- [20] V. Gonzalez-Huitron, J. A. León-Borges, A. E. Rodriguez-Mata, L. E. Amabilis-Sosa, B. Ramírez-Pereda, and H. Rodriguez, “Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4,” *Comput. Electron. Agric.*, vol. 181, no. October 2020, 2021, doi: 10.1016/j.compag.2020.105951.