

PENGARUH ARSITEKTUR CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI PENYAKIT DAUN TOMAT

(EFFECT OF CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE FOR TOMATO LEAF DISEASE CLASSIFICATION)

Rizky Arya Kurniawan¹⁾, Andi Sunyoto²⁾, dan Asro Nasiri³⁾

^{1, 2, 3)}Magister Teknik Informatika, Universitas Amikom Yogyakarta

Yogyakarta, Indonesia

e-mail: rizky.1281@students.amikom.ac.id¹⁾, andi@amikom.ac.id²⁾, asro@amikom.ac.id³⁾

ABSTRAK

Penyakit tanaman menjadi salah satu faktor yang krusial dalam keberlangsungan hidup tanaman. Pada tanaman tomat juga dibutuhkan identifikasi awal untuk dapat menangani masalah penyakit. Salah satu organ tanaman tomat yang biasa diserang penyakit adalah daun. Dengan melakukan identifikasi sejak dini maka dapat mencegah terjadinya gagal panen. Tentu dengan adanya sistem yang terlatih dapat mengurangi biaya seorang petani dalam menangani penyakit tanpa bantuan ahli. Pada penelitian ini akan dilakukan uji kemampuan arsitektur CNN untuk mengklasifikasi citra penyakit daun tomat. Dataset yang digunakan berjumlah 4079 data citra yang terbagi menjadi 3 kelas penyakit. Dari hasil percobaan yang telah dilakukan arsitektur InceptionV3 mendapatkan hasil terbaik dengan tingkat akurasi sebesar 99,67%, ResNet50 memiliki akurasi 97,36% dan MobileNet 85,81%.

Kata Kunci: *Convolutional Neural Network, Klasifikasi, Penyakit Daun Tomat*

ABSTRACT

Plant disease is one of the crucial factors in plant survival. Tomato plants also need early help to be able to deal with disease problems. One of the organs of the tomato plant that is commonly attacked by diseases is the leaves. By providing assistance early on, it can prevent crop failure. Of course, having a trained system can reduce the cost of a farmer in dealing with diseases without expert assistance. In this research, we will test the ability of the CNN architecture to classify tomato leaf disease images. The dataset used is 4079 image data which are divided into 3 disease classes. From the results of experiments that have been carried out the InceptionV3 architecture gets the best results with an accuracy rate of 99,67%, ResNet50 has 97,36% accuracy and MobileNet 85,81%.

Keywords: *Convolutional Neural Network, Classification, Tomato Leaf Disease.*

I. PENDAHULUAN

Tanaman tomat merupakan sayuran yang paling umum dijumpai di Indonesia. Tomat memiliki kandungan antioksidan yaitu vitamin E, vitamin C dan *beta-carotene*. Pada saat terjadi perubahan iklim sering muncul berbagai penyakit pada daun tomat. Identifikasi penyakit pada daun tomat sangat penting dilakukan untuk mencegah kerugian yang besar pada hasil dan kuantitas produk pertanian [1]. Mengidentifikasi penyakit tanaman dengan cara manual akan menggunakan waktu yang lama dan tenaga ahli yang mahal [2].

Banyak penelitian yang telah dilakukan dalam beberapa tahun terakhir. Penelitian yang pernah dilakukan menggunakan algoritma *Support Vector Machine* (SVM) berhasil mendapatkan akurasi sebesar 80% [3]. Peneliti lain juga melakukan ujicoba dengan menggunakan algoritma KNN mendapatkan akurasi sebesar 78%

[4], PNN 91,88% [4], *Random Forest* 95% [5] dan *Decision Tree* 78% [6].

Teknologi *artificial intelligence* berkembang sangat pesat, sehingga pengenalan objek dapat dilakukan oleh *computer vision* dan *deep learning*. Salah satu metode yang sering digunakan adalah *convolutional neural network* (CNN). CNN merupakan *neural network* yang dapat digunakan untuk klasifikasi data citra [7]. CNN juga memiliki beberapa arsitektur yang dapat membantu proses komputasi menjadi lebih efisien. Beberapa penelitian yang menggunakan arsitektur CNN dengan mengidentifikasi 10 kelas penyakit daun tomat dengan jumlah 18.159 gambar. Didapatkan hasil arsitektur ResNet50 sebesar 57,31%, VGG16 sebesar 86,19%, inceptionV3 sebesar 94,22%, DenseNet121 sebesar 94,42% dan MobileNetV2 sebesar 98,59% [8]. Dari penelitian tersebut masih terdapat beberapa langkah yang bisa dioptimalkan seperti menggunakan *optimizer Adamax*. Beberapa tahun terakhir juga telah

dikembangkan dan dioptimalkan arsitektur CNN baru. Oleh karena itu, perlu dilakukan evaluasi kinerja dari masing-masing arsitektur CNN untuk klasifikasi penyakit daun tomat.

Berdasarkan uraian diatas, peneliti akan melakukan sebuah penelitian yang difokuskan pada klasifikasi 3 (tiga) jenis penyakit daun tomat yaitu *bacterial spot*, *early blight* dan *leaf mold*. Algoritma yang digunakan adalah CNN dengan arsitektur InceptionV3, ResNet50 dan MobileNet. Dengan membandingkan 3 arsitektur CNN, akan dilakukan evaluasi kinerja untuk mendapatkan akurasi terbaik dalam mengenali penyakit daun tomat.

II. STUDI PUSTAKA

Beberapa penelitian terdahulu yang menjadi referensi dan literature review untuk penelitian ini adalah:

Penelitian yang dilakukan oleh Gehlot dkk dengan judul “*Analysis of Different CNN Architectures For Tomato Leaf Disease Classification*” mengenai klasifikasi 10 kelas penyakit daun tomat menggunakan metode CNN. Pada penelitian ini mengusulkan model CNN dengan arsitektur Vgg-16, GoogleNet, DenseNet-121, dan ResNet-101. Dari hasil percobaan yang dilakukan didapatkan hasil akurasi sebesar 99,68% pada arsitektur ResNet-101, 99,69 pada arsitektur DenseNet-121, 97,85% pada arsitektur AlexNet, 99,19% pada arsitektur VGG-16 dan 98,52% pada arsitektur GoogleNet [9].

Penelitian lainnya yang dilakukan oleh Chrisno R. Kotta dkk melakukan klasifikasi 10 kelas penyakit daun tomat menggunakan metode CNN. *Image processing* yang dilakukan adalah mengkonversi citra menjadi 64x64 pixel. Hasil dari percobaan yang dilakukan mendapatkan hasil akurasi sebesar 94 % dan tingkat kesalahan sebesar 6%. Kemudian hasil percobaan dikonversi ke *TensorFlow Lite* model lalu dibuatkan aplikasi untuk percobaan. Setelah digunakan aplikasi tersebut memiliki akurasi sebesar 80% [10].

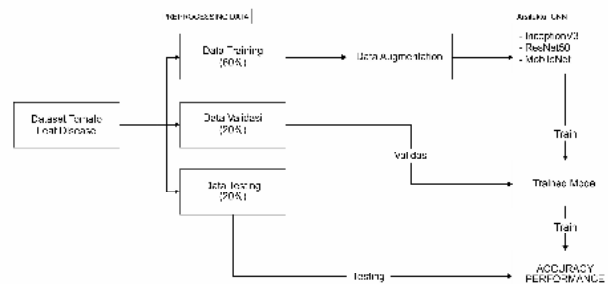
Peneliti lain mencoba menggunakan metode SVM seperti yang dilakukan oleh Debasish dkk. Dataset yang digunakan adalah data daun tomat sehat dan data daun yang terkena penyakit. Terdapat 8 kelas citra daun tomat. Jumlah *dataset* yang digunakan adalah 14000 citra. *Image processing* yang dilakukan adalah mengubah ukuran gambar dari 256x256 pixel menjadi 128x128 pixel. Pada penelitian ini menggunakan *feature extraction* yaitu *contour images*.

Percobaan tersebut menghasilkan akurasi 87,6% [11].

Penelitian yang dilakukan oleh Gaurav Langar dkk dengan judul “*Tomato Leaf Disease Detection Using Artificial Intelligence and Machine Learning*” menggunakan 4 kelas penyakit daun tomat dengan jumlah data citra 520 gambar. Peneliti menggunakan CNN-based untuk melakukan klasifikasi. *Image pre-processing* menggunakan *image enhancement*, *RGB to Lab conversion*, *filtering* dan lainnya. Dari hasil percobaan didapatkan hasil akurasi sebesar 95% [12].

III. METODE PENELITIAN

Penelitian yang dilakukan menggunakan dataset publik *plant disease* yang diunduh melalui situs Kaggle. Tahap pertama dilakukan *pre-processing* dan augmentasi data. Kemudian dilakukan *data training* menggunakan CNN dan berbagai arsitekturnya. Alur Metodologi penelitian dapat dilihat pada Gambar 1.

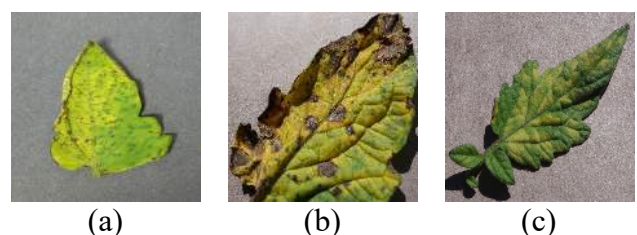


Gambar 1. Alur Metodologi Penelitian

A. Dataset

Dataset yang digunakan oleh peneliti adalah dataset penyakit daun tomat. Dataset memiliki 4079 gambar penyakit daun tomat terdiri dari 3 kelas penyakit diantaranya *bacterial spot*, *early blight* dan *leaf mold*. Dataset dapat diakses melalui situs

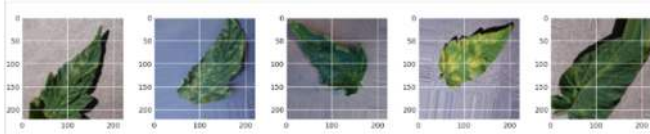
<https://www.kaggle.com/datasets/emmarex/plantdiseases>. Beberapa contoh gambar data penyakit yang digunakan dapat dilihat pada Gambar 2.



Gambar 2. Citra Penyakit Daun Tomat (a) *Bacterial Spot* (b) *Early Blight* (c) *Leaf Mold*

B. Pre-processing dan Augmentasi Data

Preprocessing data dilakukan dengan melakukan perubahan ukuran citra pada dataset yang semula 256 x 256 pixel menjadi 224 x 224 pixel. Kemudian citra dirotasi dan digeser secara *random* lalu diberi tambahan *background*. Dengan menggunakan fungsi *numpy squeeze* maka dapat menghapus entri satu dimensi dari bentuk *array*. Kemudian akan dikembalikan dengan data yang sama namun dengan dimensi yang berbeda. Hasil dari *squeeze image* dapat dilihat pada Gambar 3.



Gambar 3. Hasil Preprocessing

Setelah proses *pre-processing* dilakukan augmentasi data. Jumlah data akan disamakan menjadi 1000 gambar. Kemudian akan dibagi untuk data *training*, *validation* dan *testing*. Berikut adalah rincian dari dataset yang akan digunakan.

Tabel 1. Detail Dataset Penyakit Daun Tomat.

Nama Penyakit	Citra Asli	Training	Validation	Testing
Bacterial Spot	2127	2420	302	303
Early Blight	1000	2420	302	303
Leaf Mold	952	2420	302	303

C. Arsitektur CNN

Penelitian ini menggunakan beberapa arsitektur CNN. Arsitektur yang digunakan adalah InceptionV3, ResNet50 dan MobileNet. Berikut penjelasan secara singkat mengenai arsitektur yang digunakan.

InceptionV3 adalah model pengenalan citra yang dikembangkan oleh Google. Model inception menggunakan beberapa filter pada *layer convolutional*. Hasil dari beberapa filter tersebut dijadikan satu menggunakan *Concat Channel* [13]. Arsitektur ini dibuat untuk memenuhi *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC). Kompetisi ini mengharuskan model untuk dapat mengklasifikasikan gambar ke dalam salah satu dari 1000 kategori, termasuk hewan, objek dan pemandangan. InceptionV3 berhasil mendapatkan hasil akurasi sebesar 78,1%.

ResNet50 adalah arsitektur CNN yang digunakan untuk klasifikasi citra, deteksi objek dan

segmentasi. Pertama kali dikenalkan oleh *Microsoft Research* pada tahun 2015. ResNet50 terdiri dari 50 lapisan konvolusional dengan koneksi *shortcut* yang melewati beberapa lapisan dalam jaringan [14]. Arsitektur ini memungkinkan jaringan mempelajari lebih banyak fitur dan menghasilkan prediksi yang lebih akurat sekaligus mengurangi jumlah parameter yang perlu dioptimalkan.

MobileNet merupakan salah satu arsitektur yang dapat digunakan untuk proses komputasi data yang berlebih. MobileNet memiliki daya komputasi yang sedikit untuk menjalankan transfer learning. MobileNet dirancang agar ringan dan efisien, membuatnya ideal untuk digunakan pada perangkat seluler dengan daya komputasi dan memori terbatas. Arsitektur tersebut terbagi menjadi 2 jenis *convolution* yaitu *depthwise convolution* dan *pointwise convolution* [15].

D. Evaluasi Model

Untuk mengetahui kinerja dari model yang digunakan maka perlu dilakukan evaluasi model. *Confusion matrix* dapat digunakan untuk menguji nilai *accuracy*, *precision*, *recall* dan *f1-score* dari arsitektur yang diujikan. Contoh dari *confusion matrix multi class* dapat dilihat pada tabel 2.

Tabel 2. Confusion Matrix Multi Class.

		Prediction Class			
		Class 1	Class 2	...	Class n
Actual Class	Class 1	X ₁₁	X ₁₂	...	X _{1n}
	Class 2	X ₂₁	X ₂₂	...	X _{2n}

	Class n	X _{n1}	X _{n2}	...	X _{nn}

Dapat dilihat pada tabel 2, akan menghasilkan jumlah *true positive* (TTP) untuk semua kelas, *true negative* (TTN), *false positive* (TFP) dan *false negative* (TFN) untuk setiap kelas *i* yang dapat dihitung dengan menggunakan persamaan (1), (2), (3) dan (4) [16].

$$TPP_{all} = \sum_j^n = 1^{x_{jj}} \quad (1)$$

$$TTN_i = \sum_j^n = 1 \sum_{k=1, k \neq i}^n x_{jk} \quad (2)$$

$$TFP_i = \sum_{j=1, j \neq i}^n x_{ji} \quad (3)$$

$$TFN_i = \sum_{j=1, j \neq i}^n x_{ji} \quad (4)$$

Setelah mengetahui hasil TPP, TTN, TFP dan TFN, maka dapat dilakukan perhitungan *accuracy* (A), *Precision* (P), *recall* (R), *f-measure score* (F) dari

setiap kelas i dapat dihitung dengan persamaan (5), (6), (7) dan (8) [16].

$$A = \frac{TPP_{all}}{(Total\ Number\ of\ Entries)} \quad (5)$$

$$P_i = \frac{(TPP_{all})}{(TPP_{all} + TFP_i)} \quad (6)$$

$$R_i = \frac{(TPP_{all})}{(TPP_{all} + TFN_i)} \quad (7)$$

$$F_i = \frac{2(P_i)(R_i)}{P_i + R_i} \quad (8)$$

IV. HASIL DAN PEMBAHASAN

Pada bagian ini, kami menjelaskan hasil eksperimen dengan melakukan data *training* dan data *validation* menggunakan arsitektur CNN InceptionV3, ResNet50 dan MobileNet. Eksperimen ini bertujuan untuk mengetahui *accuracy*, *loss*, dan waktu komputasi yang dibutuhkan dari masing-masing model arsitektur.

Eksperimen ini dilakukan dengan menggunakan 50 epoch pelatihan. Hasil akurasi, *loss*, dan waktu komputasi *training* dari masing-masing arsitektur CNN dapat dilihat pada tabel 3.

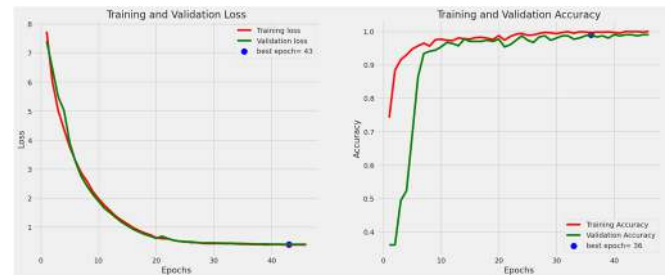
Tabel 3. Akurasi dan Waktu Komputasi

Arsitektur CNN	Train Acc(%)	Val Acc(%)	Train Loss	Val Loss	Time (Second)
InceptionV3	99,95	99,00	0,3890	0,4054	19,67
ResNet50	99,91	98,01	0,5260	0,5481	27,64
MobileNet	96,61	88,41	0,6800	0,8828	13,53

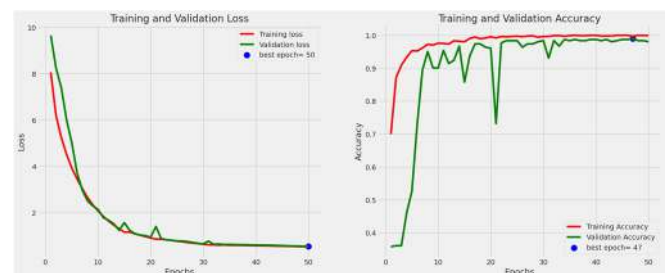
Dari hasil uji coba yang telah dilakukan, arsitektur InceptionV3 mendapatkan hasil akurasi terbaik dari semua arsitektur CNN yang diuji dengan nilai akurasi sebesar 99,95%. Kemudian arsitektur selanjutnya adalah ResNet50 dengan akurasi sebesar 99,91% dan arsitektur MobileNet dengan akurasi 96,61%. Meskipun mendapatkan tingkat akurasi yang terbaik, arsitektur InceptionV3 bukanlah arsitektur yang waktu komputasinya paling cepat. Waktu komputasi tercepat yaitu MobileNet dengan waktu 13,53 detik. Kemudian arsitektur InceptionV3 dengan waktu 19,67 detik dan terakhir arsitektur ResNet50 dengan waktu 27,64 detik.

Training loss pada tiap arsitektur juga berbeda. Arsitektur yang memiliki *training loss* paling rendah yaitu InceptionV3 dengan nilai 0,3890. Kemudian arsitektur selanjutnya adalah ResNet50 dengan nilai 0,5260 dan terakhir

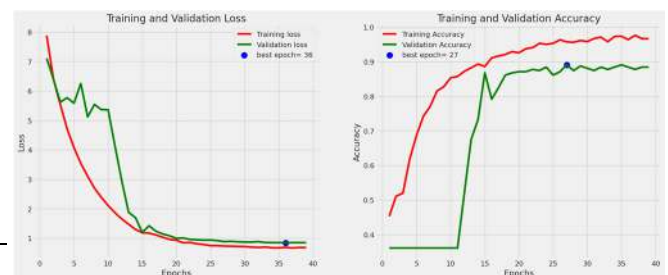
MobileNet dengan nilai 0,6800. Grafik *accuracy* dan *loss* dalam proses *training* dapat dilihat pada gambar 3-5.



Gambar 4. Accuracy dan loss training InceptionV3



Gambar 5. Accuracy dan loss training ResNet50



Gambar 6. Accuracy dan loss training MobileNet

Tabel 4 menampilkan hasil dari evaluasi *matrix* antara model yang dilatih dan data pengujian. Hasil dari percobaan yang dilakukan, Inception V3 mendapatkan hasil terbaik dari semua arsitektur yang diuji dengan akurasi 100%. Kemudian arsitektur dengan peringkat kedua adalah ResNet50 dengan akurasi 98,45% dan terakhir arsitektur MobileNet dengan akurasi 97,85%.

Tabel 4. Evaluasi Model dengan data Testing

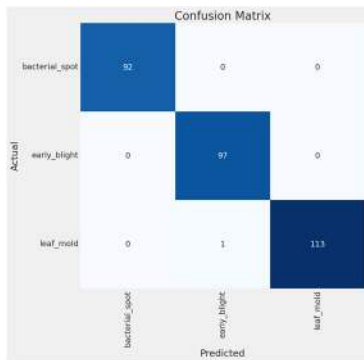
Arsitektur CNN	Test Acc(%)	Precision (%)	Recall (%)	F1-score (%)
InceptionV3	99,67	99,67	99,50	99,55
ResNet50	97,36	97,50	97,36	97,45
MobileNet	85,81	85,95	86,00	85,85

Gambar 7 adalah hasil dari *confusion matrix* arsitektur Inception V3. Dari 303 data yang diuji, terdapat 1 data yang salah dalam melakukan klasifikasi. Terjadi kesalahan pada 1 data kelas *leaf*

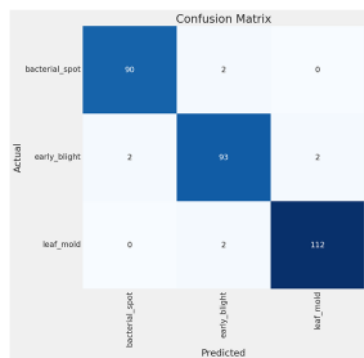
mold. Nilai *accuracy*, *precision*, *recall* dan *F1-score* dapat dilihat pada tabel 4.

Gambar 8 adalah hasil dari *confusion matrix* arsitektur ResNet50. Dari 303 data yang diuji, terdapat 8 data yang salah dalam melakukan klasifikasi. Terjadi kesalahan pada 2 data kelas *bacterial spot*, 4 data pada kelas *early blight* dan 2 data pada kelas *leaf mold*. Nilai *accuracy*, *precision*, *recall* dan *F1-score* dapat dilihat pada tabel 4.

Gambar 9 adalah hasil dari *confusion matrix* arsitektur MobileNet. Dari 303 data yang diuji, terdapat 43 data yang salah dalam melakukan klasifikasi. Terjadi kesalahan pada 9 data kelas *bacterial spot*, 25 data pada kelas *early blight* dan 9 data pada kelas *leaf mold*. Nilai *accuracy*, *precision*, *recall* dan *F1-score* dapat dilihat pada tabel 4.



Gambar 7. Hasil *Confusion Matrix* InceptionV3



Gambar 8. Hasil *Confusion Matrix* ResNet50



Gambar 9. Hasil *Confusion Matrix* MobileNet

Selanjutnya setelah dilakukan perbandingan hasil eksperimen ini dengan beberapa penelitian/penelitian sebelumnya. Dengan menggunakan arsitektur CNN, kinerja terbaik dapat diperoleh dari percobaan ini untuk mengklasifikasikan penyakit pada daun tomat.

V. KESIMPULAN

Setelah dilakukan pengujian pada masing-masing arsitektur CNN, arsitektur yang memiliki akurasi terbaik adalah arsitektur InceptionV3 dengan akurasi 99,67%. Kemudian peringkat selanjutnya adalah arsitektur ResNet50 dengan akurasi 97,36% dan terakhir arsitektur MobileNet dengan akurasi 85,81%. Hasil menunjukkan bahwa model CNN konvensional perlu digabungkan dengan arsitektur yang sudah diuji untuk dapat menghasilkan akurasi yang maksimal. Penelitian selanjutnya harus mengembangkan model CNN supaya mendapatkan waktu pelatihan dan akurasi yang lebih baik. Perlu juga untuk menambahkan kelas penyakit dan jumlah data citra penyakit pada daun tomat.

DAFTAR PUSTAKA

- [1] R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson, and R. Menaka, "Attention embedded residual CNN for disease detection in tomato leaves," *Appl. Soft Comput. J.*, vol. 86, p. 105933, 2020, doi: 10.1016/j.asoc.2019.105933.
- [2] A. Darwish, D. Ezzat, and A. E. Hassanien, "An optimized model based on convolutional neural networks and orthogonal learning particle swarm optimization algorithm for plant diseases diagnosis," *Swarm Evol. Comput.*, vol. 52, p. 100616, 2020, doi: 10.1016/j.swevo.2019.100616.
- [3] J. Trivedi, Y. Shamnani, and R. Gajjar, "Plant Leaf Disease Detection Using Machine Learning," *Commun. Comput. Inf. Sci.*, vol. 1214 CCIS, no. September, pp. 267–276, 2020, doi: 10.1007/978-981-15-7219-7_23.
- [4] Balakrishna K. and M. Rao, "Tomato Plant Leaves Disease Classification Using KNN and PNN," *Int. J. Comput. Vis. Image Process.*, vol. 9, no. 1, pp. 51–63, 2019, doi: 10.4018/ijcvip.2019010104.
- [5] M. Govardhan and M. B. Veena, "Diagnosis of Tomato Plant Diseases using Random Forest," *2019 Glob. Conf. Adv. Technol. GCAT 2019*, pp. 1–5, 2019, doi: 10.1109/GCAT47503.2019.8978431.
- [6] H. Sabrol and S. Kumar, "Intensity based feature extraction for tomato plant disease recognition by classification using decision tree," *Int. J. Comput. Sci. Inf. Secur.*, vol. 14, no. 9, pp. 622–626, 2016.
- [7] D. Hubel and T. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *J. Physiol.*, vol. 195, no. 1,

- pp. 215–243, 1968.
- [8] J. Thomkaew and S. Intakosum, “Improvement Classification Approach in Tomato Leaf Disease using Modified Visual Geometry Group (VGG)-InceptionV3,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 12, pp. 362–370, 2022, doi: 10.14569/IJACSA.2022.0131244.
- [9] M. Gehlot and M. L. Saini, “Analysis of Different CNN Architectures for Tomato Leaf Disease Classification,” *2020 5th IEEE Int. Conf. Recent Adv. Innov. Eng. ICRAIE 2020 - Proceeding*, vol. 2020, 2020, doi: 10.1109/ICRAIE51050.2020.9358279.
- [10] C. R. Kotta, D. Paseru, M. Sumampouw, T. Informatika, U. Katolik De La Salle Manado, and K. I. Kombos Manado -, “Implementasi Metode Convolutional Neural Network untuk Mendeteksi Penyakit pada Citra Daun Tomat,” *Jurnal_Pekommas_Vol._7_No.*, vol. 2, pp. 123–132, 2022.
- [11] D. Das, M. Singh, S. S. Mohanty, and S. Chakravarty, “Leaf Disease Detection using Support Vector Machine,” *Proc. 2020 IEEE Int. Conf. Commun. Signal Process. ICCSP 2020*, pp. 1036–1040, 2020, doi: 10.1109/ICCSP48568.2020.9182128.
- [12] G. Langer, P. Jain, N. Panchal, and C. Science, “and Engineering Trends Tomato Leaf Disease Detection Using Artificial,” vol. 5, no. 7, pp. 1–5, 2020.
- [13] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 2818–2826, 2016, doi: 10.1109/CVPR.2016.308.
- [14] V. Sangeetha and K. J. R. Prasad, “Deep Residual Learning for Image Recognition,” *Indian J. Chem. - Sect. B Org. Med. Chem.*, vol. 45, no. 8, pp. 1951–1954, 2016, doi: 10.1002/chin.200650130.
- [15] A. Fuadi and A. Suharso, “Perbandingan Arsitektur Mobilenet Dan Nasnetmobile Untuk Klasifikasi Penyakit Pada Citra Daun Kentang,” *JIPi (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 7, no. 3, pp. 701–710, 2022, doi: 10.29100/jipi.v7i3.3026.
- [16] P. Machart and L. Ralaivola, “Confusion Matrix Stability Bounds for Multiclass Classification,” 2012, [Online]. Available: <http://arxiv.org/abs/1202.6221>.