

## Analisis Perbandingan Performa Model Klasifikasi Kesehatan Daun Tomat menggunakan Arsitektur VGG, MobileNet, dan Inception V3

### *Analysis Tomato Leaf Health Classification Model Performance Comparison Using VGG, MobileNet, and Inception V3*

ATAM RIFA' I SUJIWANTO<sup>1</sup>, RADEN RONGGO BINTANG PRATOMO  
PRAWIRODIRJO<sup>1</sup>, PRITASARI PALUPINGSIH<sup>1,\*</sup>

#### Abstrak

Salah satu masalah dalam bidang pertanian tanaman tomat adalah adanya penyebaran penyakit pada tanaman tomat ketika ada tanaman tomat yang terkena penyakit dan terlambat diketahui serta tidak ditangani segera. Banyak penelitian tentang pengenalan dan klasifikasi pada penyakit tanaman tomat menggunakan metode *Convolutional Neural Network* (CNN). Namun, peneliti terus melakukan pembelajaran mendalam pada berbagai tugas klasifikasi objek berbasis gambar. Pada penelitian ini, dalam mengklasifikasikan kesehatan daun tanaman tomat, beberapa model diuji untuk mengidentifikasi tanaman tomat yang terjangkit penyakit. Metode yang diusulkan menggunakan pendekatan CNN dengan arsitektur VGG, MobileNet, dan Inception V3. Data citra yang digunakan berasal dari *plant disease classification merged (public dataset)*, memiliki banyak kategori gambar yang digunakan dalam karya eksperimental. Hasil eksperimen menunjukkan bahwa masing-masing model telah mencapai kinerja akurasi 98%, 93% dan 88% untuk InceptionV3, VGG, dan Mobile Net. Hasil diperoleh bahwa model dengan urutan terbaik dalam memproses data didapatkan oleh Inception V3, lalu VGG, dan Mobile Net. Meskipun demikian, mobileNet tetap memiliki efektivitas dan efisiensi saat menjalankan model yang jauh lebih baik daripada Inception V3 dan VGG.

Kata Kunci: InceptionV3, jaringan syaraf tiruan, MobileNet, pembelajaran mesin, VGG.

#### Abstract

*One of the problems in tomato farming is the spread of disease among tomato plants when affected plants are detected too late and not treated immediately. Many studies have been conducted on the identification and classification of tomato plant diseases using the Convolutional Neural Network (CNN) method. However, researchers are still intensively studying various image-based object classification tasks. In this paper, several models will be tested to classify the health of tomato plant leaves in order to identify diseases in tomato plants. The proposed method uses the CNN approach with VGG, MobileNet, and Inception V3 architectures. The image data comes from the plant disease classification merged (public dataset), which has many image categories used in the experimental work. The experimental results show that each model has achieved an accuracy performance of 98%, 93% and 88% for InceptionV3, VGG, and Mobile Net respectively. The results indicate that the best model for processing the data is Inception V3, followed by VGG and Mobile Net. However, MobileNet still has better effectiveness and efficiency when running the models compared to Inception V3 and VGG.*

*Keywords: Convolutional Neural Network, Inception V3, machine learning, MobileNet, VGG.*

---

<sup>1</sup> Program Studi Teknik Informatika, Fakultas Telematika Energi, Institut Teknologi Perusahaan Listrik Negara, Jakarta Barat 11750

\* Penulis Korespondensi : Tel/Faks: 0215-44032; Surel : pritasari@itpln.ac.id

## PENDAHULUAN

Tanaman Tomat (*Solanum lycopersicum* syn. *Lycopersicum esculentum*) merupakan tanaman yang mudah tumbuh di iklim tropis Indonesia serta memiliki nilai ekonomi yang tinggi dan menjadi salah satu bahan baku paling dibutuhkan dalam industri makanan. Menurut Badan Pusat Statistik (BPS) Indonesia, produksi tomat di Indonesia pada tahun 2021 mencapai 1.12 ton, naik sebesar 0.21% dari tahun sebelumnya sebanyak 1.11 juta ton (BPS 2022). Salah satu alasan tomat tetap menjadi komoditas panen terbesar di Indonesia karena iklim Indonesia sangat cocok untuk budidaya tanaman tomat. Tanaman tomat menghendaki tumbuh di dataran rendah atau dataran tinggi Indonesia. Namun, pertumbuhan optimal pada tomat dapat dicapai pada

ketinggian 100-2000 dpl, cocok hidup pada suhu 18-27°C pada siang hari dan 15-20°C pada malam hari serta pada keasaman tanah ideal 6-7pH (Irvan 2019). Dengan kondisi yang ideal tersebut tidak menutup kemungkinan bahwa dalam beberapa tahun ke depan tomat masih menjadi hasil panen terbanyak, serta membawa kesejahteraan bagi masyarakat Indonesia. Namun, dibalik melimpahnya hasil panen tanaman tomat, tanaman tomat juga rentan terhadap berbagai hama dan penyakit. Hal ini berpengaruh pada kesehatan tanaman tomat. Bahkan kerugian gagal panen dapat menyebabkan tomat menjadi turun sebesar 50% (Pradana 2021).

Kegagalan pada tanaman tomat tampak pada tomat yang mulai layu dan menghitam ketika terkena hama. Buruknya lagi, ketika tanaman tomat yang sudah terinfeksi dan tidak segera dibersihkan, hama dapat menyebar ke tanaman tomat lain sehingga menyebabkan semakin banyak kerugian (Hendrikus 2020). Oleh karena itu diperlukan pengamatan pada daun tomat yang mulai menghitam dan mengeluarkan bercak dengan menggunakan klasifikasi dan diagnosis yang akurat untuk membantu para petani dalam menganalisis kesehatan daun tomat dengan mengimplementasikan teknologi, agar dapat menjaga hasil panen pada tahun-tahun berikutnya.

Teknologi pembelajaran mesin (*machine learning*) dan teknik *computer vision* memiliki perkembangan yang pesat dalam metode klasifikasi. Namun, sistem klasifikasi untuk tanaman menjadi penelitian yang masih mendapatkan tantangan dan perlu dioptimalkan karena hasil data citra yang didapatkan cenderung berwarna hijau, beragamnya ciri-ciri yang tampak pada daun tomat, serta kondisi cahaya yang mempengaruhi citra gambar. Maka dari itu diperlukan ciri fisik khusus yang digunakan untuk membedakan daun tanaman yang sehat dan tidak sehat seperti warna, bentuk, dan pola tekstur daun dengan menggunakan teknik pembelajaran mesin. Baru-baru ini, *deep learning* yang merupakan sub pembelajaran dari pembelajaran mesin memungkinkan pembelajaran fitur yang kuat dan mencapai kinerja mutakhir pada berbagai tugas klasifikasi gambar (Wu *et al.* 2019). Diketahui dalam penelitian klasifikasi sebelumnya, tantangan *Visual Skala Besar ImageNet/ImageNet Large Scale Visual Classification Challenge (ILSVRC)* menandai awal dari perkembangan pesat *deep learning*. Namun, sejauh ini metode *deep learning* pada klasifikasi masih terbatas pada dataset yang terbatas (Xiaoling Xia *et al.* 2017).

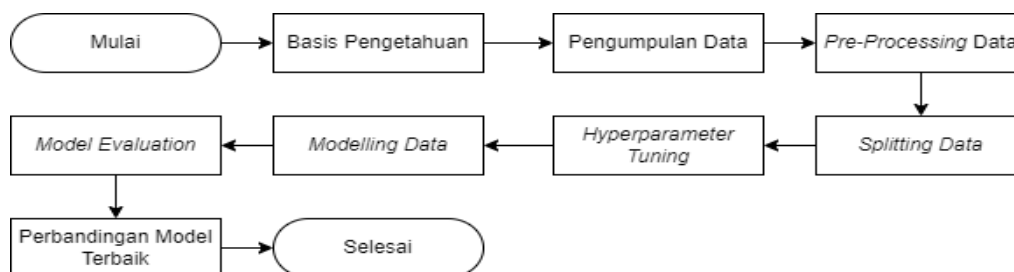
Penelitian yang dilakukan Agarwal *et al.* (2020) berhasil menerapkan algoritme CNN pada klasifikasi *dataset* daun tomat yang memiliki 10,000 gambar dengan akurasi sebesar 94.1%, kemudian dengan ResNet 50 dapat menghasilkan 88.4%. Penelitian berikutnya dilakukan Ashok *et al.* (2020) mencoba melakukan analisa pada daun tomat yang terkena penyakit dengan menggunakan algoritme CNN. Akurasi yang didapatkan adalah 99.64%. Penelitian daun tomat dengan total *dataset* sebesar 6888 gambar dengan menggunakan algoritme *Convolutional Neural Network* dengan arsitektur Resnet 101, Resnet 18 dan Resnet 150 didapatkan akurasi sebesar 98.8% serta precision sebesar 98.8% dilakukan (Kaur dan Bhatia 2019). Selanjutnya penelitian klasifikasi penyakit tomat yang dilakukan (Mamun *et al.* 2020) menggunakan delapan kelas dari penyakit tomat dengan rincian tujuh *dataset* merupakan daun yang sakit, dan satu *dataset* merupakan daun yang sehat. Dari 50 percobaan didapatkan bahwa hasil akurasi dari algoritme CNN dengan arsitektur buatan sendiri sebesar 98.77%. Kemudian pada penelitian yang dilakukan H.D. (2020) menjelaskan bahwa klasifikasi gambar daun tomat

dilakukan dengan menggunakan *Decision Tree*, *SVM*, *K-Nearest Neighbors* (KNN), serta *Naive bayes* dengan hasil terbaik diperoleh *SVM* dengan akurasi 73%. Penelitian yang dilakukan Widiyanto *et al.* (2019) berhasil mengelompokkan 5 penyakit dalam daun tomat yang terdiri dari 10,000 data dengan menggunakan *CNN* dengan akurasi sebesar 96.6%. Penelitian yang dilakukan Wang *et al.* (2019) membuat klasifikasi dengan *CNN* dengan arsitektur, *VGG-16*, *ResNet-50*, dan *ResNet-101* yang dipadukan dengan *Faster R-CNN* dan *Mask R-CNN* untuk mendapatkan *object detection*. Hasil dari penelitian ini arsitektur terbaik yang didapatkan adalah dengan menggunakan arsitektur *ResNet-101* dengan nilai akurasi sebesar 100%. Penelitian selanjutnya yang dilakukan Hong *et al.* (2020) mencoba membandingkan kinerja *CNN* dengan arsitektur *ResNet 50*, *Xception*, *MobileNet*, *ShuffleNet*, dan *DenseNet121\_Xception*. Data yang digunakan adalah data daun tomat dengan sembilan klasifikasi daun tomat. Hasil akurasi terbaik didapatkan oleh *Densenet Xception* yaitu 97.10%.

Penelitian ini akan mengusulkan tiga model untuk melakukan pengenalan pada daun tomat berbintik, daun tomat sehat, dan daun tomat yang busuk supaya dapat membantu budidaya petani dengan penyakit daun tomat yang paling sering ditemukan di Indonesia (SP. Ramlan 2019) dengan menggunakan model *VGG*, *MobileNet*, dan *Inception V3*. Alasan menggunakan Model *VGG* karena model ini dibuat oleh Universitas Oxford yang berfokus pada *Visual Geometry Group* yang memiliki 16-19 layer sehingga memiliki arsitektur yang sangat baik untuk melakukan pengklasifikasian gambar (Tammina 2019). Sedangkan *MobileNet* merupakan model yang terkenal dengan penggunaannya yang kecil serta menghemat memori dengan keunggulannya menggunakan *channel-wise filtering*, dan memiliki keunggulan *output* model ringan, serta saat melakukan training tidak membutuhkan sumber daya komputer yang rendah (Edel dan Kapustin 2022). Terakhir, *Inception V3* merupakan arsitektur yang dibuat oleh Google yang memiliki *parallel convolution* serta dapat melatih gambar secara global, uniknya dalam model ini menggabungkan teknik normalisasi dan regulasi untuk meningkatkan kinerja dan mencegah *overfitting* (Wang *et al.* 2019). Dalam penelitian ini, kinerja pendekatan kami uji dengan kumpulan data yang tersedia untuk umum pada *Kaggle* yang dikenal dengan nama *Plant Disease Classification Merged Dataset* (Saleem *et al.* 2020). Metode dalam penelitian ini berhasil mendapatkan tingkat yang lebih baik dari penelitian sebelumnya dengan rata rata 98%, dengan melihat perbandingan penggunaan *Prosesor* dan *RAM*, yang pada penelitian sebelumnya tidak dilakukan pada saat melakukan pengujian model.

## METODE

Garis besar metode penelitian ini terdiri atas beberapa langkah alur kerja yang dijelaskan pada Gambar 1.



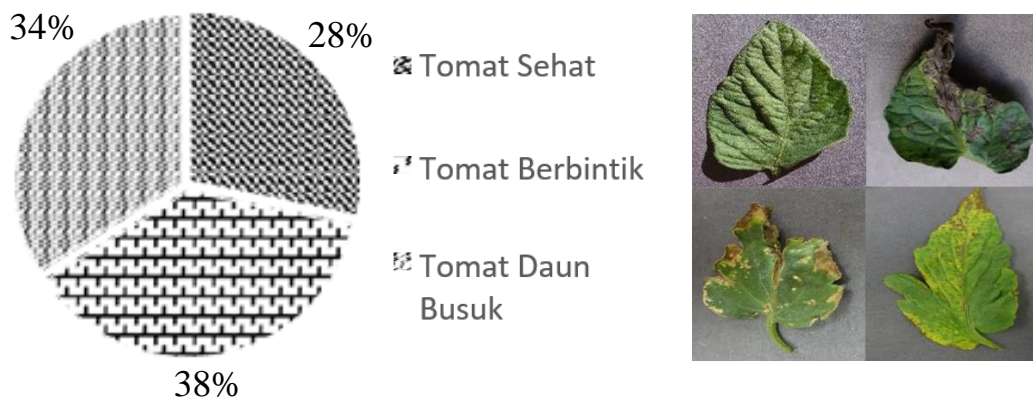
Gambar 1 Tahapan penelitian

### Basis Pengetahuan

Tahap ini melibatkan studi literatur dan penelitian terkait topik terkait morfologi pada daun tomat. Pada tahap ini, peneliti memahami landasan teori yang terkait dengan topik penelitian dan mempelajari hasil penelitian sebelumnya yang mempelajari hal serupa untuk mengidentifikasi kekurangan atau kesenjangan dalam penelitian sebelumnya dan memformulasikan pertanyaan penelitian yang tepat pada penelitian sekarang.

### Pengumpulan Data

Tahap ini melibatkan pengumpulan data yang relevan untuk menjawab pertanyaan penelitian yang telah dirumuskan. Data dapat dikumpulkan melalui sumber penelitian sebelumnya yang menjelaskan tentang klasifikasi penyakit pada daun tomat. Data yang digunakan penulis memiliki tiga *channel* (*Red, Green, Blue*) serta ukuran bervariasi dari 256 x 265 piksel sampai 1000 x 1000 piksel dengan total berjumlah 5653 gambar dalam format .png (*Portable Network Graphics*) dengan detail tomat sehat (*healthy tomato*) berjumlah 1598 sampel, tomat berbintik (*bacterial spot tomato*) berjumlah 2136 sampel dan tomat daun busuk (*late blight tomato*) berjumlah 1919 sampel. Untuk memastikan data bekerja dengan baik kami mengelompokkan data dari berbagai sudut pandang serta kualitas cahaya yang beragam.



Gambar 2 Keterangan dataset

### Preprocessing Data

Tahap ini melibatkan pemrosesan data yang telah dikumpulkan untuk memastikan bahwa data yang digunakan sesuai dengan kebutuhan analisis dan model yang akan dibangun. Pada penelitian ini, melakukan *preprocessing* data dengan membuat ukuran gambar seragam menjadi 256 x 256 pixels dengan bantuan library Keras. Selanjutnya, dilakukan proses penyeragaman gambar dengan menggunakan aplikasi pengeditan gambar secara manual untuk mengubah *lighting, clarity, color saturation* dan rotasi gambar. Selanjutnya, untuk mendapatkan data yang beragam dan meningkatkan model pelatihan data digunakan augmentasi gambar dengan menggunakan *function ImageDataGenerator* untuk membuat data lebih beragam dengan detail pada Tabel 1.

### Splitting Data

Proporsi untuk pembagian data menjadi tiga kelompok persiapan model menjadi 80% untuk *training data*, 15% untuk *testing data*, dan 5% untuk *validation data*. Saat melakukan pemisahan data kami memastikan bahwa proporsi tiap kelas data yang akan diproses memiliki pembagian proporsi jumlah data yang sama dimana kelas 0 merupakan daun tomat sehat, kelas 1 merupakan daun tomat bintik, dan kelas 2 merupakan daun tomat busuk.

Tabel 1 *Preprocessing Data*

Parameter	Value
<i>Horizontal_flip</i>	<i>true</i>
<i>Fill_mode</i>	<i>nearest</i>
<i>Shear_range</i>	0.20
<i>Zoom_range</i>	0.20
<i>Rescale</i>	1./255

Tabel 2 Persiapan data

Parameter	Data Train	Data Validation	Data Testing
<i>Target size</i>	256 x 256	256 x 256	256 x 256
<i>Color mode</i>	rgb	rgb	rgb
<i>Subset</i>	<i>training</i>	<i>validation</i>	-
<i>Batch</i>	8.00	8.00	8.00
<i>Proportion</i>	0.80	0.15	0.05

Tabel 3 Parameter *Compile*

Parameter <i>Compile</i>	Value
<i>Loss</i>	<i>sparse categorical crossentropy</i>
<i>Optimizer</i>	<i>RMSProp</i>
<i>Metrics</i>	<i>accuracy</i>
<i>Learning Rate</i>	0.0001

### **Hyperparameter Tuning**

Pada tahapan ini penulis menggunakan konfigurasi *hyperparameter* dari parameter model MobileNet, VGG, dan InceptionV3 yang telah diberikan default oleh Keras. Model tersebut dapat dilihat parameternya seperti jumlah lapisan, jumlah neuron, dan lain sebagainya. Pada tahap ini, pembuatan *hyperparameter* dilakukan pada tahap pengembangan model untuk mendapatkan model yang optimal. Tabel 3 merupakan *hyperparameter* yang digunakan untuk ketiga model.

*Loss* menggunakan *sparse categorical crossentropy* karena pada data yang digunakan memiliki klasifikasi multi kelas. *Sparse categorical crossentropy* akan membandingkan distribusi probabilitas prediksi model dengan label yang telah ditentukan, sehingga akan menghasilkan nilai yang menunjukkan seberapa dekat probabilitas prediksi dengan label yang sebenarnya. *Optimizer* menggunakan *RMSProp* karena dataset yang digunakan memiliki bobot gradien yang berbeda-beda. Sehingga *RMSProp* dapat membantu peningkatan konvergensi model dengan mengadaptasi laju pembelajaran (*learning rate*). *Metrics* dipilih *accuracy* karena dalam pelatihan ini dibutuhkan gambaran umum tentang seberapa baik model dalam melakukan klasifikasi secara keseluruhan serta dataset yang digunakan memiliki jumlah yang seimbang. *Learning rate* memiliki nilai 0.0001 karena pada pelatihan model ini mengutamakan hasil yang lebih baik sehingga model akan melalui nilai optimum atau mengalami divergensi dengan resiko pelatihan cenderung lebih lambat.

### **Modeling Data**

*Modeling data* dalam penelitian mengacu pada proses membuat model dalam *machine learning* yang dapat digunakan untuk menganalisis atau memprediksi data. Proses ini melibatkan pemilihan teknik model yang tepat, menentukan variabel yang akan dimasukkan dalam model, memilih parameter yang tepat untuk model, membuat arsitektur yang tepat, dan menguji kinerja model pada data yang tersedia. Pada praktiknya, MobileNet, VGG, dan Inception V3 memiliki karakteristik penyusunan konfigurasi model yang berbeda. Oleh karena itu, diperlukan pendekatan berbeda supaya menghasilkan model yang dapat menggambarkan hubungan antara variabel dalam data dan memberikan pemahaman yang lebih baik tentang data itu sendiri.

### Model Evaluation

*Model evaluation* atau evaluasi model pada penelitian adalah proses penilaian kinerja atau akurasi suatu model yang digunakan dalam menilai hasil model yang keluar. Untuk dapat menilai model, penulis menggunakan *convention matrix* supaya kinerja model dapat terlihat. Tujuan dari evaluasi model adalah untuk mengevaluasi sejauh mana model tersebut dapat menghasilkan hasil yang akurat dan relevan untuk tujuan yang diinginkan.

### Perbandingan Model Terbaik

Pada tahap ini kami membandingkan model dalam hasil penelitian ini akan dibandingkan berdasarkan akurasi (*accuracy*) dan presisi (*precision*). Untuk mendukung agar mendapatkan model terbaik, penulis juga memberi kondisi yang sama pada *hardware* yang digunakan pada setiap model lalu membandingkan waktu melatih data, memori yang digunakan, serta seberapa besar kinerja model yang dibebankan kepada perangkat keras.

## HASIL DAN PEMBAHASAN

Pada proses pelatihan model penulis menggunakan spesifikasi perangkat keras sebagaimana terlihat pada Tabel 4.

Tabel 4 Spesifikasi perangkat keras

Perangkat Keras	Spesifikasi
CPU	Intel(R) Xeon(R) CPU @ 2.30GHz - 2 Core
RAM	30.00 GB
STORAGE	73.10 GB
GPU	Nvidia P100 16 GB

### VGG Model

*Visual Geometry Group (VGG)* adalah salah satu model *deep learning* yang digunakan untuk tugas pengolahan citra. Model ini dikenal karena kehandalannya dalam mengatasi masalah pengenalan objek pada gambar dengan tingkat akurasi yang tinggi. VGG awalnya dikembangkan oleh grup peneliti dari Universitas Oxford pada tahun 2014 dalam sebuah kompetisi pengenalan objek yang diadakan oleh ImageNet. Model ini menggunakan arsitektur *deep Convolutional Neural Network (CNN)* yang terdiri dari beberapa layer konvolusi dan *max pooling*, yang diikuti oleh layer *fully connected* untuk klasifikasi.

Arsitektur VGG terdiri dari beberapa variasi, yang ditandai dengan jumlah layer konvolusi dan *max pooling* yang berbeda-beda. VGG16, misalnya, menggunakan 16 layer konvolusi dan *max pooling*, sementara VGG19 menggunakan 19 layer. Setiap layer konvolusi pada VGG menggunakan filter dengan ukuran kernel 3 x 3, sedangkan setiap layer *max pooling* menggunakan filter dengan ukuran 2 x 2 dan langkah sebesar 2. Tabel 5 merupakan parameter yang diberikan pada model VGG19.

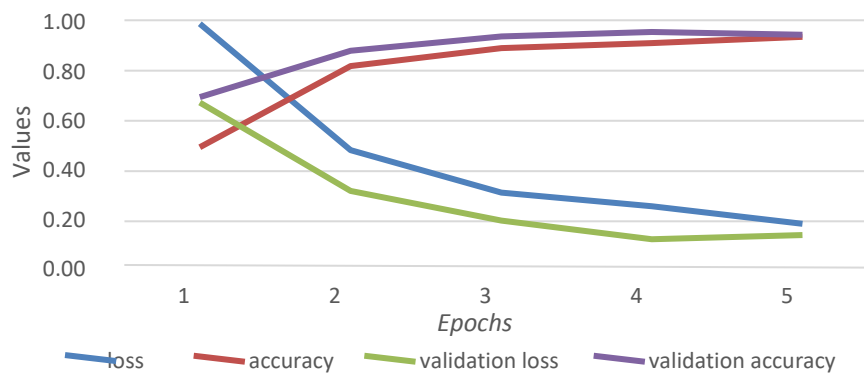
Hasil model pelatihan dengan menggunakan VGG19 dapat dilihat pada Gambar 3. Merujuk pada grafik yang dihasilkan dari model VGG19, pada iterasi pertama dan iterasi kedua terjadi penurunan drastis dari variabel *loss* dan *validation loss*. Hal ini dapat terjadi karena beberapa faktor antara lain:

1. Hal ini dapat dipengaruhi oleh banyaknya data pelatihan yang digunakan. Karena semua data yang digunakan dipisahkan pada setiap iterasi, kemungkinan akan ada pola pada iterasi pertama dan kedua yang relatif mudah untuk dipelajari oleh model. Hal ini memungkinkan model beradaptasi dengan cepat untuk belajar dan membuat prediksi yang lebih baik dengan mengurangi *loss* dan *validation loss*.
2. Inisialisasi parameter dapat mempengaruhi tingkat kerugian dan keakuratan model. Nilai parameter disesuaikan untuk mencapai hasil yang optimal, memungkinkan model membuat prediksi yang baik dengan cepat dan baik, secara signifikan mengurangi kerugian dan kerugian validasi.

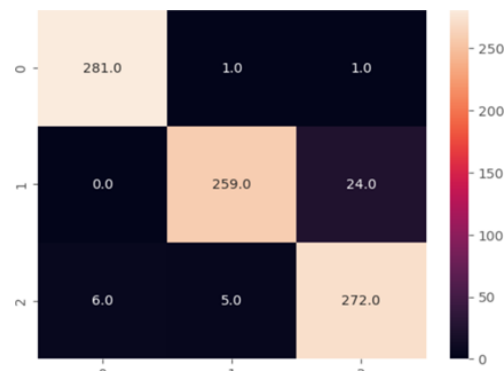
- Parameter *learning rate* diatur dalam *optimizer*. Ini memengaruhi perubahan yang dilakukan pada parameter model saat data diperbarui selama proses pelatihan. Jika Anda memilih parameter kecil untuk memulai, penyesuaian pembelajaran mesin tidak akan terlalu drastis jika Anda menggunakan parameter bernomor lebih besar.

Tabel 5 Parameter pada model VGG19

Nama Parameter	Nilai Parameter
<i>Weights</i>	<i>none</i>
<i>Include_top</i>	<i>true</i>
<i>Pooling</i>	<i>max</i>
<i>Classifier_activation</i>	<i>softmax</i>
<i>Input_tensor</i>	256 x 256 x 3
<i>Classes</i>	3.00



Gambar 3 Hasil pelatihan model inception VGG19



Gambar 4 Hasil matriks konfusi VGG19

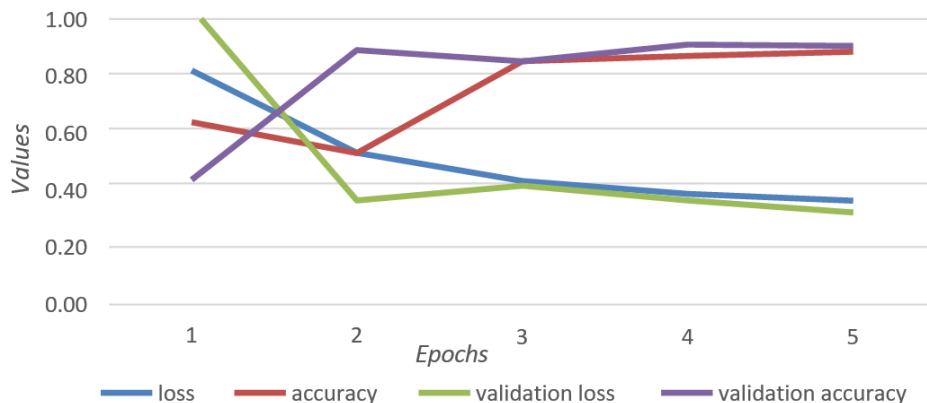
Pada Gambar 4, dapat dilihat bahwa pada baris 1 atau prediksi sebagai daun tomat sehat: (i) terdapat 281 data yang diprediksi dengan benar terhadap kelas daun tomat sehat (*True Positive/TN*); (ii) terdapat 1 data yang prediksinya salah diklasifikasikan sebagai kelas daun tomat berbakteri (*False Negative/FN*); (iii) terdapat 1 data yang prediksinya salah diklasifikasikan sebagai kelas daun tomat tua (*False Negative / FN*). Pada baris 2 atau prediksi sebagai daun tomat berbakteri: (i) terdapat 0 data yang prediksinya salah diklasifikasikan sebagai kelas daun tomat sehat (*False Positive/FP*); (ii) terdapat 259 data yang prediksinya benar diklasifikasikan sebagai kelas daun tomat berbakteri (*True Positive/TP*); (iii) terdapat 24 data yang prediksinya salah diklasifikasikan sebagai kelas daun tomat tua (*False Negative/FN*). Pada baris 3 atau prediksi sebagai daun tomat tua: (i) terdapat 6 data yang prediksinya salah diklasifikasikan sebagai kelas daun tomat sehat (*False Positive/FP*); (ii) terdapat 5 data yang prediksinya salah diklasifikasikan sebagai kelas daun tomat berbakteri (*False Positive/FP*); (iii) terdapat 272 data yang prediksinya salah diklasifikasikan sebagai kelas daun tomat tua (*True Positive/TP*).

## MobileNet

MobileNet adalah arsitektur jaringan saraf konvolusional yang dikembangkan oleh Google pada tahun 2017 untuk digunakan dalam aplikasi berbasis visi pada perangkat seluler atau perangkat yang dibatasi sumber daya. MobileNet dirancang untuk membuat model yang efisien dan ringan sekaligus mempertahankan kinerja yang baik pada tugas pengenalan gambar. Arsitektur MobileNet didasarkan pada konsep konvolusi yang dapat dipisahkan secara mendalam. Ini terdiri dari dua lapisan: lapisan konvolusional yang memproses setiap fitur masukan secara individual (konvolusi kedalaman) dan lapisan konvolusional reguler yang menggabungkannya (konvolusi titik). Hasil konvolusi kedalaman ke fitur keluaran. Konsep ini memungkinkan arsitektur MobileNet mengurangi jumlah parameter dan operasi yang diperlukan untuk pelatihan dan pengujian model, mempercepat eksekusi model, dan mengurangi penggunaan memori. Tabel 6 adalah parameter yang diberikan pada model MobileNet.

Tabel 6 Parameter pada model MobileNet

Nama Parameter	Nilai Parameter
<i>Weights</i>	<i>none</i>
<i>Pooling</i>	<i>max</i>
<i>Classifier_activation</i>	<i>softmax</i>
<i>Input_tensor</i>	256 x 256 x 3
<i>Include_top</i>	<i>true</i>
<i>Classes</i>	3.00
<i>Alpha</i>	0.50
<i>Dropout</i>	0.50

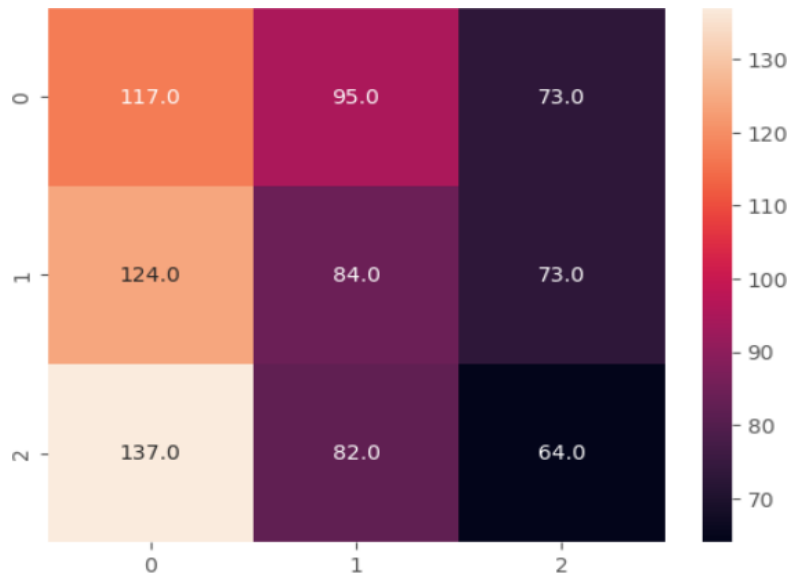


Gambar 5 Hasil pelatihan model MobileNet

Dari Gambar 5 dapat dilihat bahwa pada iterasi pertama dan iterasi kedua terjadi penurunan drastis dari variabel *loss* dan *validation loss* pada hasil pelatihan model MobileNet. Hal ini dapat terjadi karena beberapa faktor antara lain:

1. Hal ini dapat dipengaruhi oleh banyaknya data pelatihan yang digunakan. Karena semua data yang digunakan dipisahkan pada setiap iterasi, kemungkinan akan ada pola pada iterasi pertama dan kedua yang relatif mudah untuk dipelajari oleh model. Hal ini memungkinkan model beradaptasi dengan cepat untuk belajar dan membuat prediksi yang lebih baik dengan mengurangi *loss* dan *validation loss*.
2. Inisialisasi parameter dapat mempengaruhi tingkat kerugian dan keakuratan model. Nilai parameter disesuaikan untuk mencapai hasil yang optimal, memungkinkan model membuat prediksi yang baik dengan cepat dan baik, secara signifikan mengurangi kerugian dan kerugian validasi.
3. Parameter *learning rate* diatur dalam *optimizer*. Ini memengaruhi perubahan yang dilakukan pada parameter model saat data diperbarui selama proses pelatihan. Jika Anda memilih parameter kecil untuk memulai, penyesuaian pembelajaran mesin tidak akan terlalu drastis jika Anda menggunakan parameter bernomor lebih besar.





Gambar 6 Hasil matriks konfusi model MobileNet

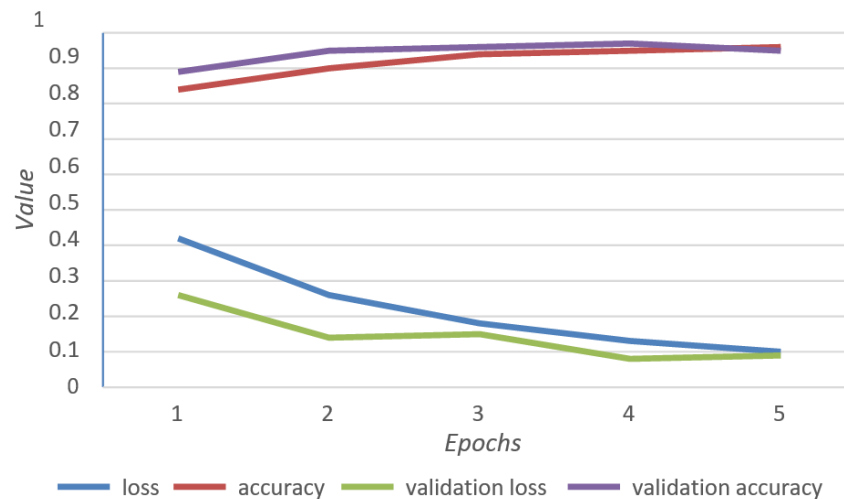
Pada Gambar 6 dapat dilihat bahwa pada baris 1 atau sebagai kelas daun tomat sehat: (i) terdapat 117 data citra yang diprediksi secara benar sebagai kelas daun tomat sehat (*True Positive/TN*); (ii) terdapat 95 data citra yang prediksinya salah diklasifikasikan sebagai kelas daun tomat berbakteri (*False Negative/FN*); (iii) terdapat 73 data citra yang prediksinya salah diklasifikasikan sebagai kelas daun tomat tua (*False Negative/FN*). Pada baris 2 atau prediksi sebagai kelas daun tomat berbakteri: (i) terdapat 124 data citra yang prediksinya salah diklasifikasikan sebagai daun tomat sehat (*False Positive/FP*); (ii) terdapat 84 data citra yang diprediksi secara benar sebagai kelas daun tomat berbakteri (*True Positive/TP*); (iii) terdapat 73 data citra yang prediksinya salah diklasifikasikan sebagai kelas daun tomat tua (*False Negative/FN*). Pada baris 3 atau prediksi sebagai kelas daun tomat tua: (i) terdapat 137 data citra yang prediksinya salah diklasifikasikan sebagai daun tomat sehat (*False Positive/FP*); (ii) terdapat 82 data citra yang prediksinya salah diklasifikasikan sebagai daun tomat berbakteri (*False Positive/FP*); (iii) terdapat 64 data citra yang diprediksi secara benar sebagai kelas daun tomat tua (*True Positive/TP*).

### Inception V3

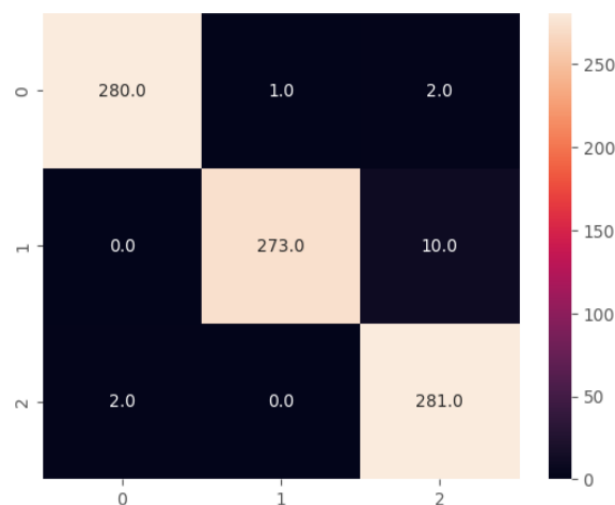
Inception V3 adalah salah satu arsitektur dari *Convolutional Neural Network* (CNN) yang dikembangkan oleh Google yang berfungsi untuk menyelesaikan masalah analisis gambar dan deteksi objek. *Canny Edge Detector* diterapkan pada setiap gambar. Metode *transfer learning* dilakukan dalam tahapan *training* pada lapisan terakhir model Inception V3 dari ImageNet. Metode ini mendapatkan persentase keakuratan yang tinggi dengan akurasi data latih sebesar 93% , dan pengujian antara 88% - 98%. Tabel 7 merupakan parameter yang digunakan untuk model Inception V3.

Tabel 7 Parameter pada model inception v3

Nama Parameter	Nilai Parameter
<i>Weights</i>	<i>none</i>
<i>Include_top</i>	<i>true</i>
<i>Pooling</i>	<i>max</i>
<i>Classifier_activation</i>	<i>softmax</i>
<i>Input_tensor</i>	256 x 256 x 3
<i>Classes</i>	3.00



Gambar 7 Hasil pelatihan model mobilenet



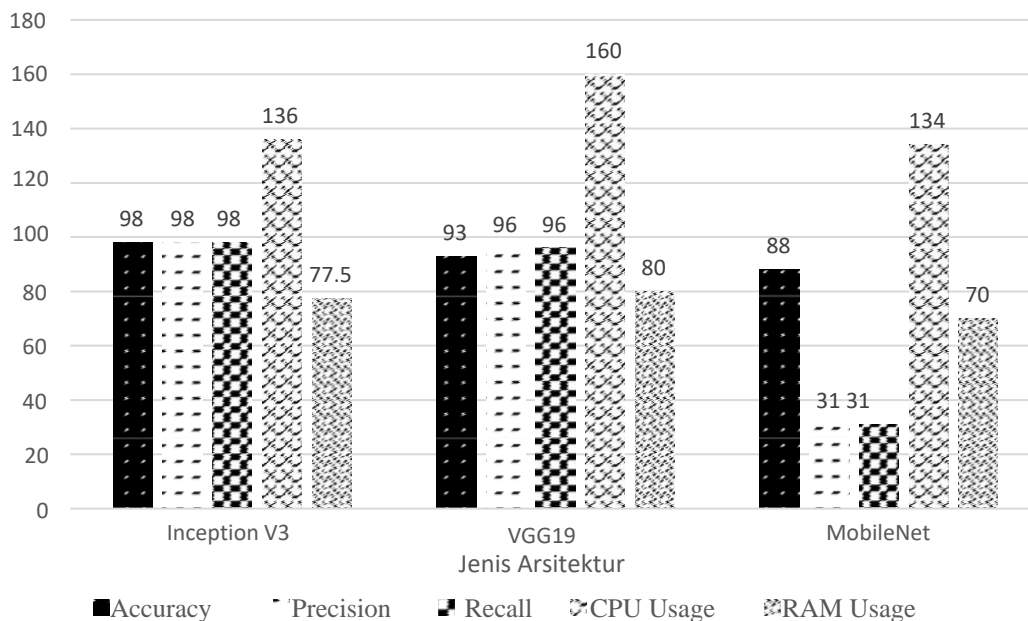
Gambar 8 Hasil matriks konvusi Inception V3

Gambar 7 menunjukkan bahwa model Inception v3 pada iterasi pertama tingkat *loss* dan *validation loss* berada di bawah 0.5 atau di bawah 50%. Jawaban yang paling memungkinkan mengapa hal ini dapat terjadi adalah jumlah data citra yang digunakan. Dengan menggunakan data citra yang sedikit, maka dengan itu dapat memungkinkan model mempelajari pola-pola dalam data tersebut dengan cepat. Hal kedua yang memungkinkan adalah *learning rate* yang digunakan itu rendah. Dengan menggunakan nilai rendah maka dapat memungkinkan mesin dapat belajar secara perlahan dan mempelajari pola-pola pada data dengan lebih baik.

Pada Gambar 8 dapat dilihat bahwa pada baris 1 atau prediksi sebagai kelas daun tomat sehat: (i) terdapat 280 data citra yang diprediksi dengan benar sebagai kelas daun tomat sehat (*True Positive/TP*); (ii) terdapat 1 data citra yang di prediksinya salah diklasifikasikan sebagai kelas daun tomat berbakteri (*False Negative/FN*); (iii) terdapat 2 data citra yang di prediksinya salah diklasifikasikan sebagai kelas daun tomat tua (*False Negative/FN*). Pada baris 2 atau prediksi sebagai kelas daun tomat berbakteri: (i) terdapat 0 data citra yang diprediksinya salah diklasifikasikan sebagai kelas daun tomat sehat (*False Positive/FP*); (ii) terdapat 273 data citra yang diprediksi secara benar sebagai kelas daun tomat sehat (*True Positive/TP*); (iii) terdapat 10 data citra yang prediksinya salah diklasifikasikan sebagai daun tomat tua (*False Negative/FN*). Pada baris 3 atau prediksi sebagai kelas daun tomat tua: (i) terdapat 2 data citra yang prediksinya salah diklasifikasikan sebagai kelas daun tomat sehat (*False Positive/FP*); (ii) terdapat 0 data citra yang prediksinya salah diklasifikasikan sebagai kelas daun tomat berbakteri (*False Positive/FP*); (iii) terdapat 281 data citra yang diprediksi secara benar sebagai kelas daun tomat tua (*True Positive/TP*).

## KESIMPULAN

Sistem klasifikasi daun tomat berdasarkan 3 kelas masih merupakan tugas yang menantang karena kesamaan antar kelas dan variasi intra kelas. Dalam penelitian ini, tiga model diusulkan untuk pengenalan penyakit pada tomat yaitu VGG 19, InceptionV3, dan Mobile Net. Kami menguji kinerja pendekatan kami dengan kumpulan data yang tersedia untuk umum yang disebut *Plant Disease Classification Merged Dataset*. Hasil kinerja yang diperoleh disajikan dan dibandingkan antara tiga model *transfer learning* yang digunakan. Gambar 9 merupakan hasil dari perbandingan model.



Gambar 9 Hasil perbandingan model

Model Inception V3 yang memiliki akurasi 98%, *precision* 98%, *recall* 98%, CPU 136%, dan RAM 77.5% merupakan model terbaik dibandingkan dengan model VGG19 dan MobileNet. Hal ini dikarenakan model InceptionV3 sangat cocok digunakan pada data multi kelas dan dengan penggunaan sumber daya komputer yang relatif sama dengan model lainnya. Selanjutnya VGG19 dengan akurasi 93%, *precision* 96%, *recall* 96%, CPU 160%, dan RAM 80% menjadi model yang terbaik kedua. Hal ini dikarenakan VGG19 sudah cukup baik dalam melatih dataset yang penulis gunakan. Hasil model VGG19 ini hampir menyerupai hasil yang didapatkan pada model InceptionV3. Sedangkan MobileNet memiliki akurasi 88%, *precision* 31%, *recall* 31%, CPU 134%, dan RAM 70% menjadi model paling rendah akurasinya disebabkan MobileNet kurang cocok untuk dataset yang memiliki bentuk yang hampir identik. Disamping itu, MobileNet memiliki prioritas untuk menciptakan model yang ringan sehingga pelatihan menjadi lebih singkat daripada VGG19 dan Inception V3 yang menyebabkan akurasi menjadi paling rendah dibandingkan dengan model lainnya. Artinya dalam pendekatan klasifikasi 3 kelas, dengan *dataset* yang digunakan, MobileNet tidak cocok diterapkan menjadi model, walaupun dalam segi sumber daya, MobileNet memiliki sumber daya paling rendah dibandingkan VGG19 dan Inception V3.

## DAFTAR PUSTAKA

- Agarwal, M., Singh, A., Arjaria, S., Sinha, A. and Gupta, S. 2020. *ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network*. *Procedia Computer Science*, 167, pp.293–301. doi:<https://doi.org/10.1016/j.procs.2020.03.225>.
- Ashok, S., Kishore, G., Rajesh, V., Suchitra, S., Sophia, S.G.G. and Pavithra, B. 2020. *Tomato Leaf Disease Detection Using Deep Learning Techniques*. [daring] IEEE Xplore. doi:<https://doi.org/10.1109/ICCES48766.2020.9137986>.
- Edel, G. and Kapustin, V. 2022. Exploring of the MobileNet V1 and MobileNet V2 models on NVIDIA Jetson Nano microcomputer. *Journal of Physics: Conference Series*, 2291(1), p.012008. doi:<https://doi.org/10.1088/1742-6596/2291/1/012008>.
- H.D., G. 2020. *Machine Learning Approach towards Tomato Leaf Disease Classification*. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(1), pp.490–495. doi:<https://doi.org/10.30534/ijatcse/2020/67912020>.
- Hendrikus, H. ed., 2020. Hama Tanaman Tomat. [daring] Cybex Pertanian. Tersedia Di:<Http://Cybex.Pertanian.Go.Id/Mobile/Artikel/94637/Hama-Dan-Penyakit-Penting-Tanaman-Tomat--Lycopersicum-esculentum/> [diakses 14 Mar. 2023].
- Hong, H., Lin, J. and Huang, F. 2020. *Tomato Disease Detection and Classification by Deep Learning*. [daring] IEEE Xplore. doi:<https://doi.org/10.1109/ICBAIE49996.2020.00012>.
- Kaur, M. and Bhatia, R. 2019. *Development Of An Improved Tomato Leaf Disease Detection and Classification Method*. 2019. *IEEE Conference on Information and Communication Technology*. doi:<https://doi.org/10.1109/cict48419.2019.9066230>.
- Keras. (n.d.). Keras documentation: Optimizers. [online] keras.io. Tersedia: <https://keras.io/api/optimizers/>.
- Mamun, Md.A.A., Karim, D.Z., Pinku, S.N. and Bushra, T.A. 2020. TLNet: A Deep CNN model for Prediction of tomato Leaf Diseases. 2020 23rd International Conference on Computer and Information Technology (ICCIT). doi:<https://doi.org/10.1109/iccit51783.2020.9392664>.
- Pradana, W. 2021. Gagal Panen, Petani di Lembang Babat Habis 25 Ribu Pohon Tomatnya. [daring] detiknews. Tersedia di: <https://news.detik.com/berita-jawa-barat/d-5869139/gagal-panen-petani-di-lembang-babat-habis-25-ribu-pohon-tomatnya> [diakses 11 Maret 2023].
- S, Irvan. 2019. Budidaya Tanaman Tomat. [daring] Pertanian. Tersedia di: <http://cybex.pertanian.go.id/mobile/artikel/90456/Budidayah-Tanaman-Tomat/> [diakses 19 Maret 2023].
- Saleem, M.H., Potgieter, J. and Arif, K.M. 2020. *Plant Disease Classification: A Comparative Evaluation of Convolutional Neural Networks and Deep Learning Optimizers*. *Plants*, [daring] 9(10), p.1319. doi:<https://doi.org/10.3390/plants9101319>.
- SP., Ramlan. 2019. Penyakit Tomat dan Cara Pengendaliannya. [daring] Tersedia: <http://cybex.pertanian.go.id/mobile/artikel/72724/Penyakit-Tomat-dan-Cara-Pengendaliannya> [Diakses 27 Mei 2023].
- Tammina, S. 2019. Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images. *International Journal of Scientific and Research Publications (IJSRP)*, 9(10), p.p9420. doi:<https://doi.org/10.29322/ijsrp.9.10.2019.p9420>.
- Wang, Q., Qi, F., Sun, M., Qu, J. and Xue, J. 2019. *Identification of Tomato Disease Types and Detection of Infected Areas Based on Deep Convolutional Neural Networks and Object Detection Techniques*. *Computational Intelligence and Neuroscience*, 2019, pp.1–15. doi:<https://doi.org/10.1155/2019/9142753>.
- Wang, C., Chen, D., Hao, L., Liu, X., Zeng, Y., Chen, J. and Zhang, G. 2019. Pulmonary Image Classification Based on Inception-v3 Transfer Learning Model. *IEEE Access*, 7, pp.146533–146541. doi:<https://doi.org/10.1109/access.2019.2946000>.

- Widiyanto, S., Fitrianto, R. and Wardani, D.T. 2019. *Implementation of Convolutional Neural Network Method for Classification of Diseases in Tomato Leaves*. 2019 Fourth International Conference on Informatics and Computing (ICIC). doi:<https://doi.org/10.1109/icic47613.2019.8985909>.
- Wu, X., Zhan, C., Lai, Y.-K., Cheng, M.-M. and Yang, J. 2019. *IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition*. [daring] IEEE Xplore. doi:<https://doi.org/10.1109/CVPR.2019.00899>.
- [www.bps.go.id](http://www.bps.go.id). 2022. Badan Pusat Statistik. [daring] Tersedia di: <https://www.bps.go.id/indicator/55/61/1/produksi-tanaman-sayuran.html>.