

**PENERAPAN ALGORITMA *MOBILENETV3* DALAM
KLASIFIKASI PENYAKIT TANAMAN JAGUNG**

SKRIPSI

OLEH:

TULUS MARKUS SIAGIAN

208160014



PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MEDAN AREA

MEDAN

2024

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

Document Accepted 23/12/24

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah

3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area
Access From (repositorv.uma.ac.id)23/12/24

Penerapan Algoritma *MobileNetV3* Dalam Klasifikasi Penyakit Tanaman Jagung

SKRIPSI

Diajukan Sebagai Salah Satu Syarat untuk Memperoleh
Gelar Sarjana (S1) di Fakultas Teknik
Universitas Medan Area



OLEH
TULUS MARKUS SIAGIAN
208160014

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA
2024

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

Document Accepted 23/12/24

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

HALAMAN PENGESAHAN

Judul : PENERAPAN ALGORITMA *MOBILENETV3* DALAM
KLASIFIKASI PENYAKIT TANAMAN JAGUNG
Nama : Tulus Markus Siagian
NPM : 208160014
Fakultas : Teknik

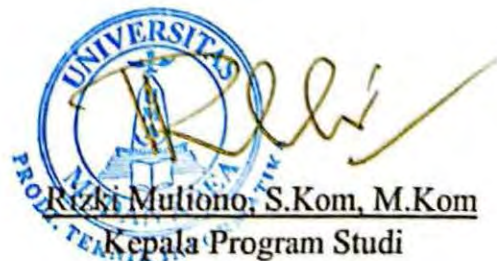
Disetujui Oleh
Komisi Pembimbing



Muhathir, S.T, M.Kom
Pembimbing



Dr. Eng. Supriatno, ST., MT
Dekan



Rizki Muliono, S.Kom, M.Kom
Kepala Program Studi

Tanggal Lulus: 30 Agustus 2024

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

Document Accepted 23/12/24

HALAMAN PERNYATAAN

Saya menyatakan bahwa skripsi yang saya susun, sebagai syarat memperoleh gelar sarjana merupakan hasil karya tulis saya sendiri. Adapun bagian-bagian tertentu dalam penulisan skripsi ini yang saya kutip dari hasil karya orang lain telah dituliskan sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan ilmiah.

Saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi-sanksi lainnya dengan perturan yang berlaku, apabila di kemudian hari ditemukan adanya plagiat dalam skripsi ini.

Medan, 30 Agustus 2024



Tulus Markus Siagian
NPM: 208160014

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR/SKRIPSI/TESIS UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Medan Area, saya yang bertanda tangan di bawah ini:

Nama : Tulus Markus Siagian

NPM : 208160014

Program Studi : Teknik Informatika

Fakultas : Teknik

Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Medan Area Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty-Free Right) atas karya ilmiah saya yang berjudul:

Penerapan Algoritma *MobileNetV3* Dalam Klasifikasi Penyakit Tanaman Jagung

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Medan Area berhak menyimpan, mengalihmedia/format-kan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir/skripsi/tesis saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di: Medan
Pada tanggal: 30 Agustus 2024
Yang Menyatakan



Tulus Markus Siagian
NPM: 208160014

RIWAYAT HIDUP

Penulis bernama lengkap Tulus Markus Siagian, lahir di Silombu, Desa Hutagur-gur 1, Kabupaten Toba pada 17 Juni 2001 anak pertama dari 5 bersudara dari ayah Tigor Siagian dan Ibu Ani Habeahan. Tahun 2019 penulis lulus dari SMK Swasta Soposurung Balige, Kabupaten Toba. Dan pada 2020 terdaftar sebagai mahasiswa Program Studi Teknik Informatika Universitas Medan Area.

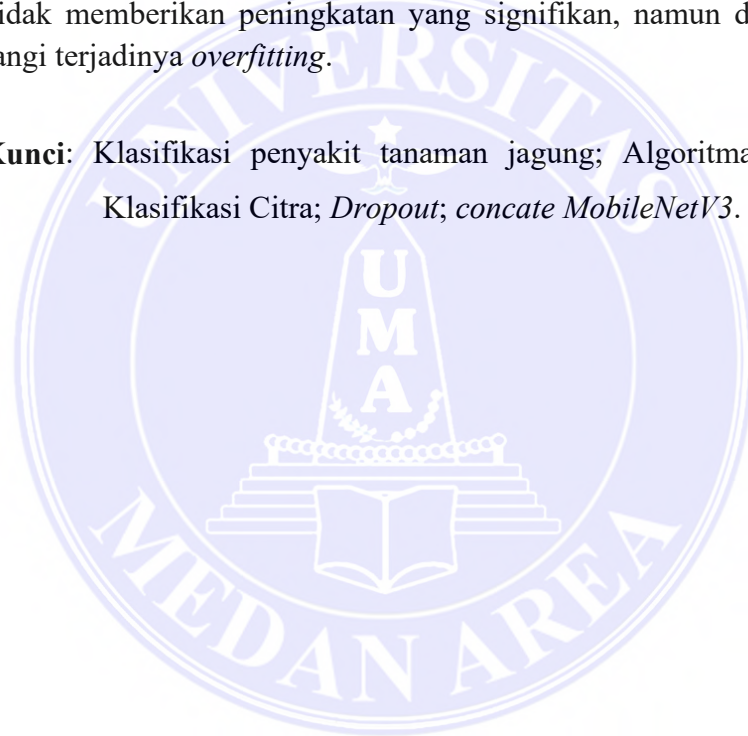
Berkat pertolongan Tuhan Yesus, dan usaha serta doa dari kedua orang tua penulis dapat menjalani aktivitas akademik di Universitas Medan Area, dan puji Tuhan penulis dapat menyelesaikan Tugas Akhir/skripsi berjudul “Penerapan Algoritma *MobileNetV3* Dalam Klasifikasi Penyakit Tanaman Jagung”.



ABSTRAK

Penelitian ini bertujuan untuk menerapkan algoritma *MobileNetV3* dalam mengklasifikasikan penyakit tanaman jagung. *Dataset* yang digunakan sebanyak 3000 gambar daun jagung yang terbagi menjadi tiga kategori, yaitu daun hawar, daun karat, dan daun sehat. Algoritma *MobileNetV3* dipilih karena memiliki kemampuan dalam mengelompokkan data citra gambar secara efisien serta mampu melakukan pengenalan pola citra. Dalam penelitian ini, dilakukan pengujian model *MobileNetV3-Large* dengan dan tanpa *dropout*, *MobileNetV3-Small* dengan dan tanpa *dropout*, serta kombinasi *concat* antara *MobileNetV3-Large* dan *Small*. Hasil penelitian menunjukkan bahwa *MobileNetV3-Large* tanpa *dropout* mendapatkan nilai akurasi tertinggi sebesar 95,00%. Penggunaan *dropout* pada model tidak memberikan peningkatan yang signifikan, namun dapat membantu mengurangi terjadinya *overfitting*.

Kata Kunci: Klasifikasi penyakit tanaman jagung; Algoritma *MobileNetV3*; Klasifikasi Citra; *Dropout*; *concat* *MobileNetV3*.



ABSTRACT

Indonesia's geographical location in the tropics and its status as an agrarian country makes farming the primary occupation of its people. One of the challenges in agriculture is leaf diseases, often caused by viruses, bacteria, fungi, and nematodes. These diseases affect the growth and production of crops, especially corn. However, visual observations made by farmers are often insufficiently sensitive and time-consuming. This study aimed to apply the MobileNetV3 algorithm in classifying corn leaf diseases. A dataset of 3000 corn leaf images was used, divided into three categories: blight, rust, and healthy leaves. The MobileNetV3 algorithm was chosen due to its ability to efficiently classify image data and recognize image patterns. In this research, MobileNetV3- Large models with and without dropout, MobileNetV3-Small models with and without dropout, and concatenated combinations of both MobileNetV3-Large and Small with and without dropout were tested. The results showed that the MobileNetV3-Small and Large models with dropout achieved the highest accuracy of 0.9567 or 95.67%. The use of dropout did not significantly improve performance but helped reduce overfitting.

Keywords: *Corn Plant Disease Classification; MobileNetV3; Image Classification; Dropout; MobileNetV3-Large; MobileNetV3-Small; MobileNetV3 Concatenation.*

KATA PENGANTAR

Puji Syukur penulis panjatkan kepada Tuhan Yang Maha Kuasa atas segala karuniaNya serta penyertaanNya sehingga skripsi ini berhasil diselesaikan. Tema yang dipilih dalam penelitian ini adalah “*Penerapan Algoritma MobileNetV3 Dalam Klasifikasi Penyakit Tanaman Jagung*”.

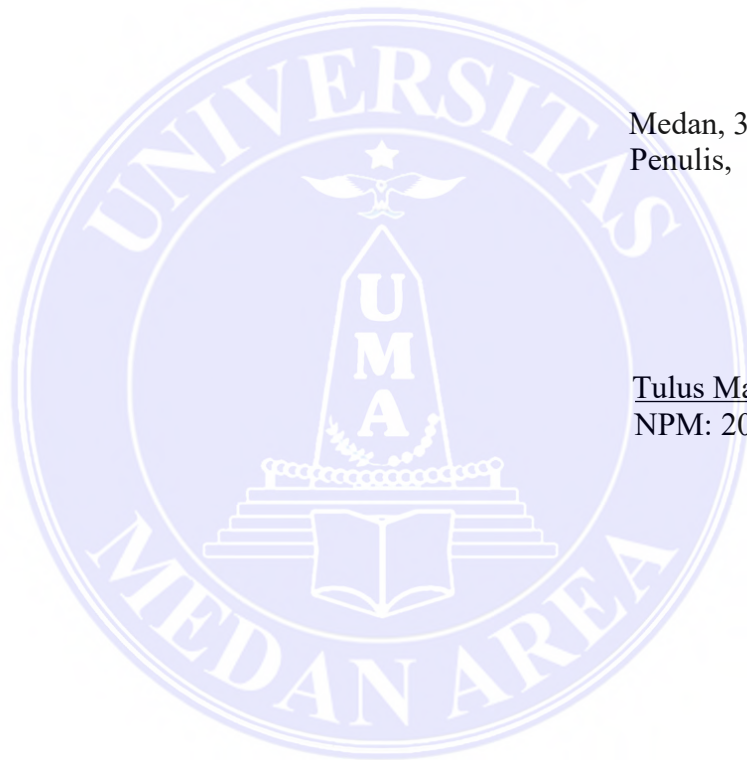
Skripsi ini merupakan salah satu syarat untuk menyelesaikan Pendidikan untuk mencapai gelar sarjana di Program Studi Teknik Informatika Fakultas Teknik Universitas Medan Area. Pada kesempatan kali ini penulis mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Dadan Ramdan, M.Eng, M.Sc. selaku Rektor Universitas Medan Area.
2. Bapak Dr. Eng. Supriatno, ST, MT selaku Dekan Fakultas Teknik Universitas Medan Area.
3. Bapak Rizki Muliono, S,Kom., M.Kom selaku Kepala Program Studi Teknik Informatika Universitas Medan Area.
4. Bapak Muhathir, S.T., M.Kom selaku Dosen Pembimbing yang telah membantu penulis dari segi materi dan moril sehingga penulis dapat menyelesaikan skripsi ini.
5. Bapak Dr. Sayuti Rahman, ST, M.Kom sebagai ketua, dan ibu Nurul Khairina, S.Kom, M.Kom sebagai pembanding II yang telah memberikan saran serta masukan dalam penulisan skripsi ini dan meluangkan waktunya menjadi panitia seminar proposal hingga siding skripsi penulis.

6. Seluruh Dosen Teknik Informatika Universitas Medan Area yang senantiasa memberikan pembelajaran dan ilmu yang bermanfaat kepada penulis.
7. IT Support Teknik Informatika bang Robby Kurniawan Sari Damanik, ST yang telah membantu penulis dalam menyiapkan pemberkasan yang diperlukan dari awal hingga selesai.
8. Orang tua penulis yaitu Bapak Tigor Siagian dan Ibu Ani Habeahan yang telah mendoakan dan memberi semangat serta membantu penulis dalam segi materi dan moril sehingga penulis dapat menyelesaikan skripsi ini dengan sebaik-baiknya.
9. Fittauli Lumban Gaol yang telah menemani mensupport saya dari awal penulisan hingga selesainya skripsi saya ini dengan baik.
10. Seluruh teman-teman yang sudah memberikan dukungannya selama penulisan skripsi ini, khususnya teman-teman Teknik Informatika Angkatan 2020.
11. Teman-teman satu kos saya yang tidak dapat saya sebutkan Namanya satu persatu yang telah memberikan support dan semangat dalam pengerjaan skripsi ini.
12. Seluruh pihak yang tidak dapat disebutkan satu persatu yang membantu dalam penyelesaian penulisan skripsi ini.
13. Dan saya juga mengucapkan terima kasih kepada diri saya sendiri yang telah berusaha kuat dalam penulisan skripsi ini dari awal hingga selesai.
Semoga kedepannya menjadi orang yang sukses dan menjadi

kebanggaan bagi orang tua dan keluarga, dan semoga skripsi ini dapat bermanfaat bagi orang lain yang telah membaca, Aamiin.

Penulis menyadari bahwa skripsi ini masih memiliki kekurangan, oleh karena itu kritik dan saran yang bersifat membangun yang penulis harapkan demi kesempurnaan skripsi ini. Penulis berharap skripsi ini dapat bermanfaat baik untuk kalangan Pendidikan maupun Masyarakat. Akhir kata penuli ucapkan terima kasih.



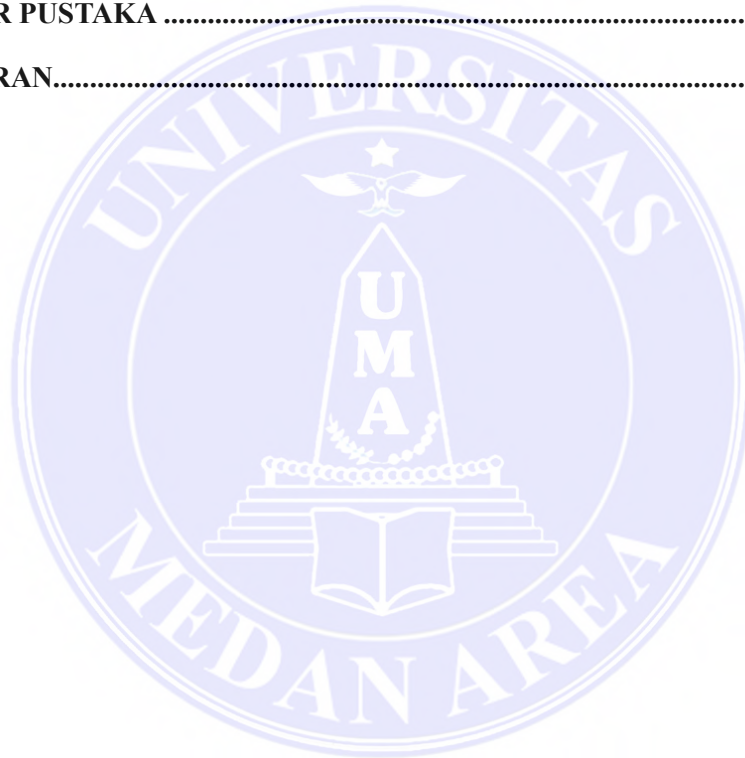
Medan, 30 Agustus 2024
Penulis,

Tulus Markus Siagian
NPM: 208160014

DAFTAR ISI

KATA PENGANTAR.....	ii
ABSTRAK	ii
ABSTRACT.....	iii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian	4
1.4 Manfaat Penelitian	4
1.5 Batasan Masalah	4
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 <i>Deep Learning</i>	6
2.2 Klasifikasi	7
2.3 <i>MobileNetV3</i>	7
2.4 Penyakit Tanaman Jagung.....	10
2.5 Penelitian Terdahulu.....	11
BAB III METODOLOGI PENELITIAN	16
3.1 Skema Penelitian.....	16
3.2 Pengumpulan Data	16
3.5 <i>Hyperparameter</i>	19
3.8 Waktu dan Tempat Penelitian.....	21
3.9 Spesifikasi Perangkat.....	22
BAB IV HASIL dan PEMBAHASAN	23
4.1 Hasil.....	23
4.1.1 Visualisasi Data.....	23
4.1.2 Augmentasi Data.....	24
4.1.3 <i>Training MobileNetV3-Large menggunakan Dropout</i>	25
4.1.4 <i>Training MobileNetV3-Small menggunakan Dropout</i>	28
4.1.5 <i>Training MobileNetV3 concate large dan small menggunakan Dropout</i>	31
4.1.6 <i>Training MobileNetV3-Large tanpa Dropout</i>	34

4.1.7.	<i>Training MobileNetV3-Small tanpa Dropout</i>	37
4.1.8.	<i>Training MobileNetV3 Concate Small dan Large menggunakan Dropout</i>	40
4.1.9.	<i>Training MobileNetV3 Concate Large dan Small tanpa Dropout</i>	43
4.1.10.	<i>Training MobileNetV3 Concate Small dan Large tanpa Dropout</i>	46
4.2.	Pembahasan.....	49
4.2.1	Temuan.....	49
4.2.2	Perbandingan dengan Penelitian Terkait	50
BAB V KESIMPULAN dan SARAN		52
5.1.	Kesimpulan	52
5.2.	Saran	53
DAFTAR PUSTAKA		54
LAMPIRAN		58

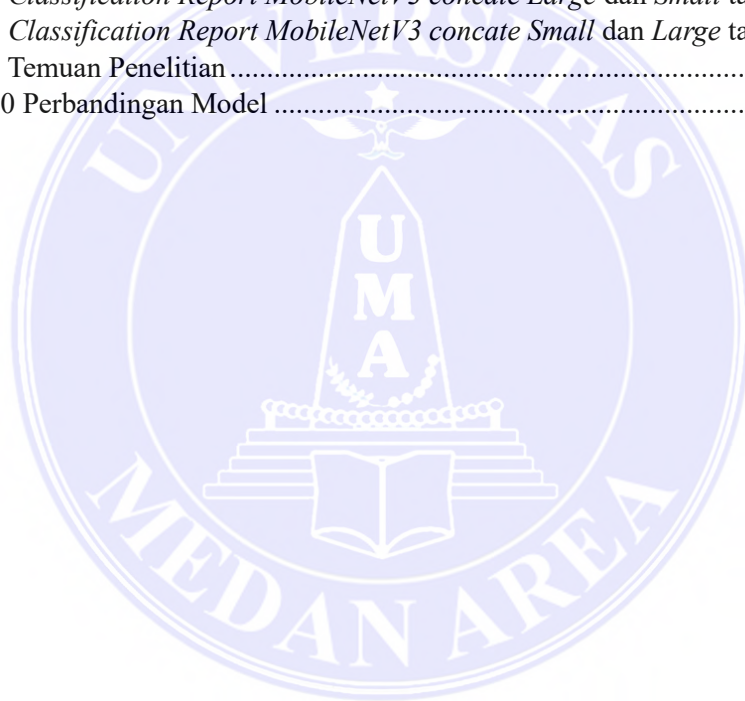


DAFTAR GAMBAR

Gambar 2.1 Alur Kerja Modul SE	8
Gambar 2.2 Arsitektur (a). <i>MobileNetV3-Large</i> , (b). <i>MobileNetV3-Small</i>	9
Gambar 3.1 Skema Penelitian	16
Gambar 4.1 Visualisasi Data (a) Daun Hawar, (b) Daun Karat, (c) Daun Sehat	23
Gambar 4.2 Hasil <i>Augmentasi</i> (a) <i>horizontal flip</i> , (b) <i>rotation range</i> 10, (c) <i>zoom range</i> 0.1, (d) <i>brightness</i> 0.8 dan 1.2	24
Gambar 4.3 <i>Grafik Training MobileNetV3-Large</i> menggunakan <i>Dropout</i>	25
Gambar 4.4 <i>Confussion Matrix MobileNetV3-Large</i> menggunakan <i>Dropout</i>	26
Gambar 4.5 <i>Grafik Training MobileNetV3-Small</i> menggunakan <i>Dropout</i>	28
Gambar 4.6 <i>Confussion Matrix MobileNetV3-Small</i> menggunakan <i>Dropout</i>	29
Gambar 4.7 <i>Grafik Training MobileNetV3 concate large dan small</i>	31
Gambar 4.8 <i>Confussion Matrix MobileNetV3 concate large dan small</i>	32
Gambar 4.9 <i>Grafik Training MobileNetV3-Large</i> tanpa <i>Dropout</i>	34
Gambar 4.10 <i>Confussion Matrix MobileNetV3-Large</i> tanpa <i>Dropout</i>	35
Gambar 4.11 <i>Grafik Training MobileNetV3-Small</i> tanpa <i>Dropout</i>	37
Gambar 4.12 <i>Confussion Matrix MobileNetV3-Small</i> tanpa <i>Dropout</i>	38
Gambar 4.13 <i>Grafik Training Concate MobileNetV3 Small dan Large</i>	40
Gambar 4.14 <i>Confussion Matrix Concate MobileNetV3 Small dan Large</i>	41
Gambar 4.15 <i>Grafik training MobileNetV3 concate Large dan Small</i> tanpa <i>Dropout</i>	43
Gambar 4.16 <i>Confussion Matrix MobileNetV3 concate Large dan Small</i> tanpa <i>Dropout</i>	44
Gambar 4.17 <i>Grafik training MobileNetV3 concate Small dan Large</i> tanpa <i>Dropout</i>	46
Gambar 4.18 <i>Confussion Matrix MobileNetV3 concate Large dan Small</i> tanpa <i>Dropout</i>	47
Gambar 4.19 <i>Grafik</i> akurasi tiap model	50

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu.....	11
Tabel 2.2 Penelitian Terkait.....	13
Tabel 3.1 Sampel Data	17
Tabel 3.2 <i>Split Dataset</i>	18
Tabel 3.3 <i>Hyperparameter</i>	19
Tabel 3.4 <i>Confussion Matrix</i>	20
Tabel 4.1 <i>Classification Report MobileNetV3-Large</i> menggunakan <i>Dropout</i>	27
Tabel 4.2 <i>Classification Report MobileNetV3-Small</i> menggunakan <i>Dropout</i>	30
Tabel 4.3 <i>Classification Report MobileNetV3 concate large dan small</i>	33
Tabel 4.4 <i>Classification Report MobileNetV3-Large</i> tanpa <i>Dropout</i>	36
Tabel 4.5 <i>Classification Report MobileNetV3-Small</i> tanpa <i>Dropout</i>	39
Tabel 4.6 <i>Classification Report MobileNetV3 Concate Small dan Large</i>	42
Tabel 4.7 <i>Classification Report MobileNetV3 concate Large dan Small</i> tanpa <i>Dropout</i> ..	45
Tabel 4.8 <i>Classification Report MobileNetV3 concate Small dan Large</i> tanpa <i>Dropout</i> ..	48
Tabel 4.9 Temuan Penelitian	49
Tabel 4.10 Perbandingan Model	50



BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia yang letak geografisnya berada di wilayah tropis serta negara agraris, sehingga mata pencaharian utama masyarakatnya merupakan bertani. Dengan kondisi iklim, tanah, dan sumber daya lainnya, Indonesia memiliki potensi besar untuk mengembangkan teknologi di sektor pertanian. Untuk meningkatkan produktivitas pertanian, sumber daya pertanian harus digunakan seefektif mungkin (Kusuma et al., 2023). Dalam jurnal yang ditulis oleh (Amin et al., 2022) yang dikutip dari *Smallholders, Food Security and the Environment, Int. Fund Agricult. Develop., Rome, Italy, 2013, p. 29*, mengatakan bahwa petani kecil yang ada di negara berkembang menghasilkan lebih dari 80% produksi pertanian serta laporan yang mengindikasikan bahwa lebih dari 50% kehilangan hasil panen diakibatkan oleh hama serta penyakit pada tanaman. Pemasalahan yang sering terjadi dalam pertanian adalah penyakit daun. Penyakit daun sering disebabkan oleh virus, bakteri, jamur, dan nematoda yang menyerang tanaman serta mempengaruhi pertumbuhan dan produksi yang dimiliki tanaman tersebut (Jeffry et al., 2022).

Jagung yang merupakan tumbuhan kebutuhan primer di Indonesia selain padi dan gandum, dengan didukung irigasi yang baik, tanaman jagung dapat tumbuh dengan baik meskipun di cuaca panas ataupun di cuaca yang dingin. Meskipun begitu tanaman tersebut sangat rentan terkena berbagai penyakit, terutama pada area daun, hal tersebut mengakibatkan penurunan kuantitas serta kualitas hasil jagung (Lesmana Hermawan, 2023). Penyakit tanaman jagung dapat menyerang

kapan saja yang diakibatkan oleh beberapa komponen seperti patogen atau disebut dengan parasit serta dampak dari lingkungan sekitar (Rahmanita et al., 2023).

Penting nya deteksi dini penyakit pada tanaman jagung, guna untuk mencegah penyebaran yang lebih luas dan kerusakan yang lebih besar. Pengamatan yang dilakukan secara visual oleh para petani seringkali tidak cukup sensitif dan memakan banyak waktu.

Untuk mengatasi hal tersebut dibutuhkan teknologi komputasi untuk mengklasifikasi penyakit tanaman jagung, sehingga membantu para petani meminimalisir waktu yang digunakan, serta meningkatkan kualitas hasil tanaman, salah satu teknologi yang digunakan ialah penggunaan teknologi deep learning dengan menggunakan algoritma *MobileNetV3* yang dapat mempermudah klasifikasi terhadap penyakit tersebut.

Algoritma *MobileNetv3* dipilih karena mampu untuk mengelompokkan data citra gambar yang efisien serta mampu untuk melakukan pengenalan pola citra tersebut. Dalam penempatan kelas penyakit tanaman jagung tersebut, algoritma ini diharapkan dapat memberikan hasil yang akurat, dan memungkinkan bagi para petani untuk mendapatkan hasil panen yang lebih baik.

Penelitian mengenai algoritma *MobileNetV3* pernah dilakukan oleh (Budiarto & Utamaningrum, 2023) penelitian dilakukan mengenai sistem klasifikasi permukaan jalan dan penghindaran jalan berlubang pada kursi roda pintar dengan metode *MobileNetV3-Small* mendapat hasil yang akurat dan baik. Kemudian (Prima et al., 2022) penelitian ini digunakan untuk mendeteksi otomatis nominal uang kertas rupiah untuk tuna netra menggunakan algoritma arsitektur *ssd MobileNetV3*. Selanjutnya penelitian (Abd Elaziz et al., 2021) dalam meningkatkan

klasifikasi gambar *COVID-19* menggunakan algoritma *MobileNetV3* dan *Aqila Optimizer*, dalam penelitian tersebut menyatakan bahwa algoritma *MobileNetV3* dapat memberikan hasil kinerja yang baik dalam mengklasifikasi gambar *Covid-19*.

Selain itu penelitian mengenai tentang penyakit tanaman jagung yang dilakukan oleh (Rahmanita et al., 2023) dalam mengimplentasikan metode *Naive Bayes* dan *Information Gain* untuk klasifikasi penyakit dan hama tanaman jagung dengan hasil akurasi terbaik. Kemudian penelitian yang dilakukan oleh (Azizah, 2023) dalam klasifikasi penyakit daun jagung menggunakan metode *Concolutional Neural Network AlexNet* dengan *dataset* yang digunakan sebanyak 4198 mendapatkan hasil akurasi yang baik. Kemudian penelitian yang dilakukan oleh (Suhendra & Juliwardi, 2022) dalam penelitiannya mengidentifikasi dan klasifikasi penyakit daun jagung menggunakan *Support Vector Machine* mendapatkan hasil yang baik dengan penggunaan fitur gabungan warna & tekstur.

Berdasarkan efektifnya penelitian-penelitian sebelumnya yang menggunakan algoritma *MobileNetV3* serta penelitian yang terkait dengan penyakit tanaman jagung, maka dilakukan penelitian dalam menerapkan algoritma *MobileNetV3* dalam klasifikasi penyakit tanaman jagung.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini ialah bagaimana menerapkan algoritma *MobileNetV3* dalam mengklasifikasi penyakit tanaman jagung?

1.3 Tujuan Penelitian

Salah satu tujuan dari penelitian ini adalah sebagai berikut:

1. Menguji kinerja algoritma *MobileNetV3* dalam mengklasifikasi penyakit pada tanaman jagung.
2. Menerapkan algoritma *MobileNetV3* dalam klasifikasi penyakit tanaman jagung.

1.4 Manfaat Penelitian

Adapun manfaat dalam penelitian ini adalah membantu dalam mendeteksi penyakit pada tanaman jagung sedini mungkin, sehingga mengurangi kerugian hasil panen.

1.5 Batasan Masalah

Adapun batasan masalah dalam penelitian ini, yaitu:

1. Objek penelitian yang diklasifikasi adalah tanaman jagung
2. Jumlah data eksperimen yang digunakan sebanyak 3000 dataset gambar daun tanaman jagung dengan format .jpg/jpeg yang diambil dengan menggunakan kamera *smartphone*.

1.6 Sistematika Penulisan

Dalam penulisan skripsi ini, penulis memberikan gambaran sistematika penulisan dalam mengklasifikasikan penyakit pada tanaman jagung terbagi menjadi 5 (lima) bagian yaitu:

BAB I PENDAHULUAN

Bagian ini berisi tentang latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas tentang teori yang diambil dari beberapa sumber yang sesuai dengan permasalahan dalam penelitian ini.

BAB III METODOLOGI PENELITIAN

Dalam bab ini membahas mengenai metodologi penelitian yang akan dilakukan serta langkah-langkah yang digunakan pada penelitian yang akan dilakukan.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi pengujian dan analisis dari hasil pengujian dari sistem yang telah dibangun berdasarkan hasil perancangan pada bab 3 sebelumnya.

BAB V PENUTUP

Bab ini berisi kesimpulan dan saran seluruh penelitian yang dilakukan.

BAB II

LANDASAN TEORI

2.1 Deep Learning

Deep Neural Network atau *Deep learning* adalah *Neural Network* yang memiliki banyak layer. *Deep learning* adalah bagian dari pembelajaran mesin yang berkaitan dengan algoritma, dimana algoritma ini meniru cara kerja struktur dan fungsi otak yang biasa disebut jaringan saraf tiruan (Rochmawati et al., 2021).

Deep Learning adalah seperangkat algoritma machine learning yang berusaha belajar dalam berbagai level, yang pada dasarnya *deep learning* merupakan jaringan saraf tiruan yang mampu belajar dan beradaptasi dalam jumlah data yang besar untuk menyelesaikan masalah yang sulit (Alfarizi et al., 2023). *Deep Learning* juga merupakan sebuah pembelajaran mesin, pembelajaran mesin ini mirip dengan bayi yang baru lahir, dimana ada miliaran neuron yang terkait di otak (Taye, 2023).

Dalam Kecerdasan Buatan (AI), pembelajaran mesin merupakan kemampuan untuk beradaptasi secara otomatis tanpa campur tangan manusia, salah satu bagian dari pembelajaran mesin adalah pembelajaran mendalam yang menggunakan jaringan saraf untuk meniru proses pembelajaran otak manusia. *Deep Learning* membutuhkan lebih banyak data, dan dapat menyesuaikan diri dengan situasi yang baru (Taye, 2023). Pemrosesan bahasa alami (NLP) merupakan suatu bidang penelitian yang telah melihat kemajuan besar. Para peneliti telah dapat mengembangkan model bahasa yang akurat dan efektif dengan bantuan model pembelajaran mendalam seperti jaringan saraf berulang (RNN) dan transformator. Model ini dapat membantu otomatisasi layanan, memungkinkan mesin pencari

dan asisten virtual yang lebih canggih, serta meningkatkan komunikasi antara manusia dengan mesin(Sharifani & Amini, 2023).

Dalam jurnal yang ditulis oleh (Yogasudha S et al., 2018) tentang *deep learning* mengatakan bahwa selama bertahun-tahun *deep learning* telah membuat kemajuan besar dalam memecahkan masalah, hal ini terbukti sangat baik dalam menemukan struktur yang rumit dalam data berdimensi tinggi, oleh karena itu dapat diterapkan dalam banyak ilmu pengetahuan, bisnis, serta dalam pemerintahan.

2.2 Klasifikasi

Klasifikasi ialah sebuah proses untuk menempatkan objek ke dalam kelas-kelas yang berbeda untuk membuatnya lebih mudah mengidentifikasi serta berusaha menggeneralisasikan struktur yang sudah diketahui untuk diterapkan pada data baru (Khultsum & Taufik, 2023).

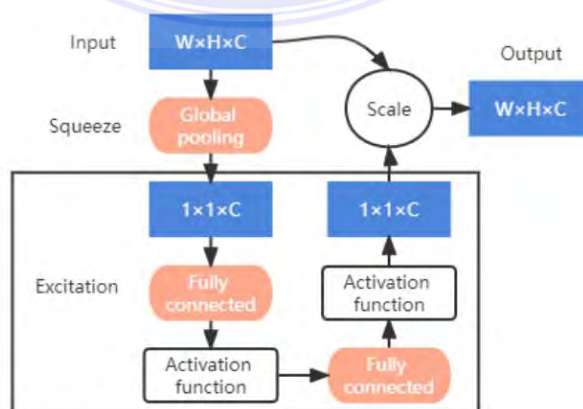
Klasifikasi juga merupakan sebuah metode pengelompokan yang didasarkan pada karakteristik yang dapat dilakukan oleh manusia maupun teknologi. Dalam bidang penambangan data, klasifikasi juga memiliki peran penting dalam pembelajaran pengetahuan yang sebelumnya (Fajar & Mutaqin, 2024).

2.3 *MobileNetV3*

MobileNetV3 merupakan bagian dari generasi jaringan ringan yang dibangun dari *MobileNetV1* dan *MobileNetV2* dengan menggunakan struktur residu terbalik dan konvolusi yang dapat dipisahkan secara mendalam, untuk meningkatkan efisiensi operasional dan kinerja jaringan dibutuhkan pencarian serta adaptasi arsitektur saraf platform untuk mengoptimalkan struktur dan parameter jaringan(Zhu et al., 2023).

Dalam jurnal (Nugroho & Baihaqi, 2023) *MobileNetV3* juga memanfaatkan teknologi *NAS (Neural Architecture Search)* untuk mengoptimalkan struktur jaringan. Pada proses ini, beberapa lapisan yang membutuhkan waktu komputasi yang lama dalam mengoptimalkan struktur jaringan tersebut. Sehingga konvolusi pada lapisan yang pertama dikurangi dari 32 menjadi 16, meskipun begitu akurasi dari jaringan konstan.

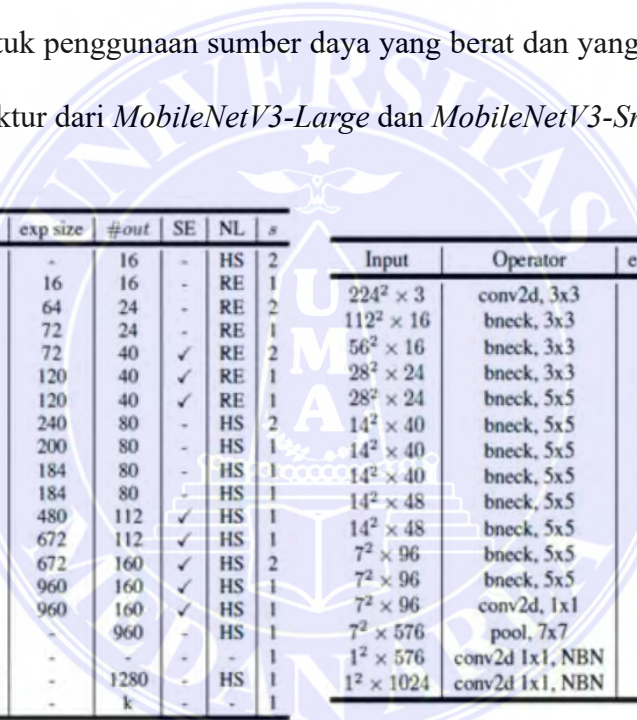
MobileNetV3 juga dikenal karena arsitekturnya yang efisien dan kinerjanya yang sangat baik, dengan demikian dapat memanfaatkan pembelajaran dalam klasifikasi penyakit tanaman jagung. Dengan menggabungkan metode augmentasi data sangat penting untuk meningkatkan ketahanan model serta kemampuan dalam menggeneralisasikan secara sengaja untuk memasukkan variasi kedalam dataset pelatihan (Patel & Modi, 2024), selain itu *MobileNetV3* juga memperkenalkan modul *SE (Squeeze and Exit)*, dimana selama proses pelatihan modul *SE* memberikan bobot pada setiap saluran, sehingga dapat menekankan fitur dengan bobot yang lebih tinggi dan menekan fitur yang tidak signifikan secara keseluruhan (Si et al., 2023).



Gambar 2.1 Alur Kerja Modul SE

Selain itu, dalam jurnal yang ditulis oleh(Jiang & Tong, 2022) mengatakan bahwa modul SE akan meningkatkan jumlah parameter dan jumlah total perhitungan jaringan. Meskipun jumlah perhitungan lapisan terhubung penuh yang digunakan tidak lebih besar dari lapisan konvolusional, jumlah parameternya akan meningkat secara signifikan.

Dalam penelitian ini menggunakan dua model *MobileNetV3* yaitu *MobileNetV3-Large* dan *MobileNetV3-Small*, kedua model tersebut masing-masing dirancang untuk penggunaan sumber daya yang berat dan yang ringan. Berikut ini adalah Arsitektur dari *MobileNetV3-Large* dan *MobileNetV3-Small*.



Input	Operator	exp size	#out	SE	NL	s	Input	Operator	exp size	#out	SE	NL	s
$224^2 \times 3$	conv2d	-	16	-	HS	2	$224^2 \times 3$	conv2d, 3x3	-	16	-	HS	2
$112^2 \times 16$	bneck, 3x3	16	16	-	RE	1	$112^2 \times 16$	bneck, 3x3	16	16	✓	RE	2
$112^2 \times 16$	bneck, 3x3	64	24	-	RE	2	$56^2 \times 16$	bneck, 3x3	72	24	-	RE	2
$56^2 \times 24$	bneck, 3x3	72	24	-	RE	1	$28^2 \times 24$	bneck, 3x3	88	24	-	RE	1
$56^2 \times 24$	bneck, 5x5	72	40	✓	RE	2	$28^2 \times 24$	bneck, 5x5	96	40	✓	HS	2
$28^2 \times 40$	bneck, 5x5	120	40	✓	RE	1	$14^2 \times 40$	bneck, 5x5	240	40	✓	HS	1
$28^2 \times 40$	bneck, 5x5	120	40	✓	RE	1	$14^2 \times 40$	bneck, 5x5	240	40	✓	HS	1
$28^2 \times 40$	bneck, 3x3	240	80	-	HS	2	$14^2 \times 40$	bneck, 5x5	120	48	✓	HS	1
$14^2 \times 80$	bneck, 3x3	200	80	-	HS	1	$14^2 \times 48$	bneck, 5x5	144	48	✓	HS	1
$14^2 \times 80$	bneck, 3x3	184	80	-	HS	1	$14^2 \times 48$	bneck, 5x5	288	96	✓	HS	2
$14^2 \times 80$	bneck, 3x3	184	80	-	HS	1	$7^2 \times 96$	bneck, 5x5	576	96	✓	HS	1
$14^2 \times 80$	bneck, 3x3	480	112	✓	HS	1	$7^2 \times 96$	bneck, 5x5	576	96	✓	HS	1
$14^2 \times 112$	bneck, 3x3	672	112	✓	HS	1	$7^2 \times 96$	conv2d, 1x1	-	576	✓	HS	1
$14^2 \times 112$	bneck, 5x5	672	160	✓	HS	2	$7^2 \times 96$	pool, 7x7	-	-	-	-	1
$7^2 \times 160$	bneck, 5x5	960	160	✓	HS	1	$1^2 \times 576$	conv2d 1x1, NBN	-	1024	-	HS	1
$7^2 \times 160$	bneck, 5x5	960	160	✓	HS	1	$1^2 \times 1024$	conv2d 1x1, NBN	-	k	-	-	1
$7^2 \times 160$	conv2d, 1x1	-	960	-	HS	1							
$7^2 \times 960$	pool, 7x7	-	-	-	-	1							
$1^2 \times 960$	conv2d 1x1, NBN	-	1280	-	HS	1							
$1^2 \times 1280$	conv2d 1x1, NBN	-	k	-	-	1							

(a).

(b).

Gambar 2.2 Arsitektur (a). *MobileNetV3-Large*, (b). *MobileNetV3-Small*

Perbedaan dua model *MobileNetV3* ini adalah model *large* memiliki layer yang lebih banyak dan kompleks, dengan artian model tersebut memerlukan daya komputasi yang lebih banyak dibandingkan dengan model *small*(Matthew Patrick, Chairisni Lubis, n.d.).

2.4 Penyakit Tanaman Jagung

Tanaman jagung merupakan salah satu tanaman sereal dan bahan pangan yang penting karena mengandung sumber karbohidrat. Namun Tumbuhan jagung berpotensi terserang serbuan hama ataupun penyakit yang bisa melanda kapan saja(Suherman, 2021).

Pada penyakit tanaman jagung yang diakibatkan oleh jamur *Puccinia sorghi* *Schwein* dan penyakit hawar daun yang diakibatkan oleh jamur *Helmithosporium turcicum* (pass). Penyakit daun karat disebabkan oleh jamur *Puccinia sorghi*, Terindikasi dini dalam bentuk bitnik-bintik merah serta keluar serbuk seperti tepung bercorak coklat kekuningan. Penyakit ini bisa mengakibatkan tumbuhan tidak bisa melakukan fotosintesis dengan sempurna hingga menghambat pertumbuhannya, dan membuat tanaman jagung bisa mati(Iswantoro & Handayani UN, 2022). Penyakit *Puccinia sorghi* banyak ditemukan di pegunungan tropic dan diwilayah beriklim sedang. Penyakit hawar daun terindikasi berbentuk bercak kecil, berupa oval dan bercak tersebut terus memanjang berupa elips serta tumbuh menjadi nekrotik(Yoggyanto et al., 2024).

Upaya yang dilakukan untuk menanggulangi penyakit yang mengganggu pada tanaman jagung terus dilakukan hingga saat ini, namun adanya factor dari lingkungan seperti, cuaca, suhu serta kelembaban yang dapat menyebabkan penyakit pada tanaman jagung makin berkembang. Sehingga dibutuhkan pemanfaatan teknologi dalam menanggulangi penyakit tanaman tersebut, dengan menerapkan metode kecerdasan buatan(Widianto et al., 2023).

2.5 Penelitian Terdahulu

Pada penelitian ini, penulis mengambil inspirasi serta merujuk pada penelitian sebelumnya untuk menjadikan referensi untuk menghindari plagiat dalam menyusun penelitian ini. Berikut adalah penelitian terdahulu dengan penelitian ini, dapat dilihat pada Tabel 2.1.

Tabel 2.1 Penelitian Terdahulu

No	Penulis dan Tahun	Judul Penelitian	Hasil Penelitian
1	Yi Zhao, Hancheng Huang, Zhixiang Li, Huang Yiwang dan Manjie Lu, 2022	Sistem Klasifikasi Sampah Cerdas Berdasarkan Peningkatan <i>MobileNetV3-Large</i> .	Dalam penelitian ini, teknologi pengenalan gambar mengadopsi model <i>MobileNetV3-Large</i> untuk menganalisis informasi gambar dan menghasilkan empat kategori sampah primer, seperti sampah yang dapat didaur ulang, sisa makanan, sampah berbahaya, serta 158 kategori sampah sekunder.
2	Dhritiman Saha, Meetkumar Pareshbhai Mangukai, dan Annamalai Manickavasagan, 2023	Penyebaran Model <i>MobileNetV3</i> Secara Real-Time di Perangkat Komputasi Tepi Menggunakan Citra Warna RGB untuk Klasifikasi Varietas Buncis.	Penelitian ini menggunakan model berbasis <i>CNN</i> dengan pendekatan <i>Transfer Learning</i> untuk membedakan empat varietas buncis yang berbeda. Model <i>CNN</i> yang digunakan untuk penelitian ini adalah <i>NasNet-A</i> , <i>MobileNetV3</i> , dan <i>EfficientNet-B0</i> , terlihat bahwa tiga model tersebut menggeneralisasi dengan baik pada dataset uji, dengan masing-masing

			96,82%, 99,09%, dan 98,18%.
3	Jin Zhu, Chuanhui Zhang, dan Changjiang Zhang, 2023	Klasifikasi <i>Papaver somniferum</i> dan <i>Papaver rhoeas</i> Berdasarkan Citra Kapsul yang Terlihat Menggunakan Jaringan <i>MobileNetV3-Small</i> yang Dimodifikasi dengan Transfer Learning.	Dalam penelitian ini, para peneliti mengusulkan model P-MobileNet untuk klasifikasi PSPR, berdasarkan MobileNetV3-Small yang telah dimodifikasi dengan Transfer Learning. Analisis kinerja klasifikasi model yang berbeda menunjukkan bahwa model P-MobileNet yang diusulkan memiliki keunggulan akurasi klasifikasi yang tinggi, parameter yang sedikit, dan kecepatan deteksi yang cepat. Dibandingkan dengan MobileNetV3-Small, P-MobileNet mempertahankan akurasi klasifikasi yang tinggi yaitu 98,9%.
4	Galih Ashari Rakhmat, Muhammad Fikri Haekal, 2022	Peningkatan Performa MobileNetV3 dengan <i>Squeeze-and-Excitation</i> (Studi Kasus Klasifikasi Kesegaran Ikan Berdasarkan Mata Ikan)	Penelitian yang dilakukan dengan mengukur kinerja dari sistem yang dibuat menggunakan model evaluation untuk menghitung tingkat accuracy, precision dan recall. Menunjukkan grafik perbandingan kinerja antara 18 variasi model yang telah dilatih dengan dilakukan pengujian dengan data uji sebanyak 876 citra kesegaran ikan.

			Didapatkan akurasi tertinggi pada model 8 yaitu model arsitektur MobileNetV3-Small (epoch = 100, dan learning rate = 0.0001, optimizer = ADAM, batch size = train 10, val 10) dengan 68% accuracy, 69% precision, 67% recall dan 68% F1-score.
--	--	--	--

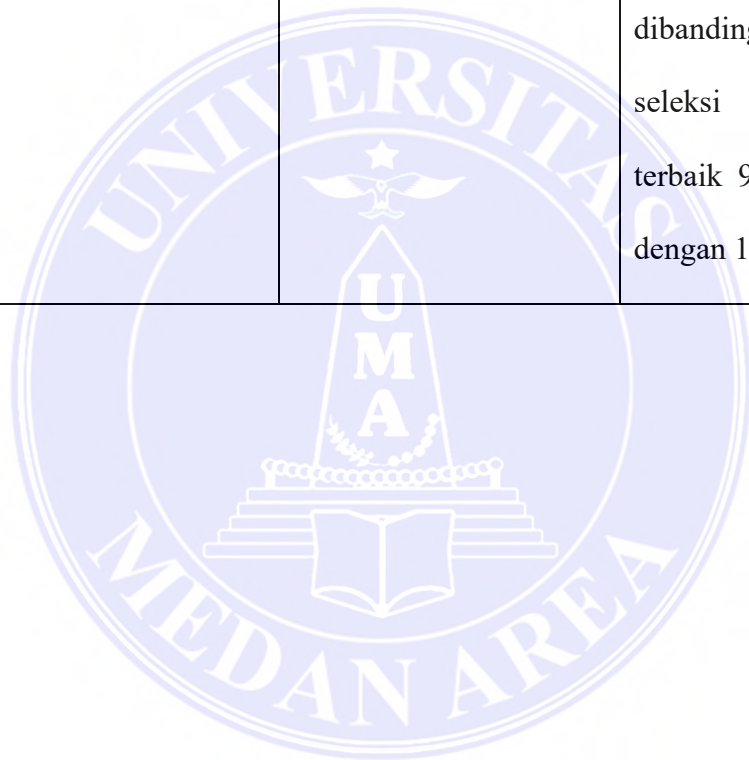
Selain dari penelitian terdahulu yang mengenai dengan algoritma *MobileNetV3*, penulis juga mengambil inspirasi yang terkait dengan topik penelitian yang akan dilakukan, sehingga memudahkan dalam penulisan dan membandingkan dengan penelitian yang dilakukan dengan algoritma yang berbeda. Berikut ini adalah penelitian terkait dapat dilihat pada Tabel 2.2.

Tabel 2.2 Penelitian Terkait

No	Penulis dan Tahun	Judul	Hasil penelitian
1	Afivah Dwi Nurcahyati, Ronny Mahfudding Akbar, Soffa Zahara, 2022	Klasifikasi Citra Penyakit Pada Daun Jagung Menggunakan <i>Deep Learning</i> dengan Metode <i>Convolution Neural Network</i> (CNN)	Hasil terbaik yang dicapai dengan kernel ukuran 3x3 menggunakan algoritma Adam, mendapatkan hasil akurasi sebesar 84% pada data <i>testing</i> dan 89% pada data <i>training</i> .

2	<p>Qudsiah Nur Azizah, 2023</p>	<p>Klasifikasi Penyakit Daun Jagung Menggunakan Metode <i>Convolutional Neural Network AlexNet</i></p>	<p>Uji coba menunjukkan bahwa system yang menggunakan arsitektur <i>AlexNet</i> dalam mengklasifikasikan penyakit daun jagung mendapatkan hasil akurasi 90%.</p>
3	<p>Rivansyah Suhendra, Ilham Juliwardi, Sanusi, 2022</p>	<p>Identifikasi dan Klasifikasi Penyakit Daun Jagung Menggunakan <i>Support Vector Machine</i></p>	<p>Klasifikasi menggunakan SVM dengan 22 fitur (18 warna dan 4 tekstur) dari 3600 data <i>plantVillage-Dataset</i>. Eksperimen dilakukan dengan variasi fitur warna, tekstur dan gabungan dan pembagian data (80:20, 70:30, 60:40). Hasil terbaik menunjukkan akurasi 99,5% dengan fitur gabungan warna dan tekstur.</p>

4	Eza Rahmanita, Yudha Dwi Putra Negara, Yeni Kustiyaningsih, Verdi Sasmeka, Bain Khusnul Khotimah	Implementasi Metode <i>Naïve Bayes</i> dan <i>Information Gain</i> untuk Klasifikasi Penyakit dan Hama Tanaman Jagung	Uji coba yang dilakukan menunjukkan bahwa penggunaan <i>Naïve Bayes</i> dengan <i>information gain</i> meningkatkan akurasi rata-rata sebesar 3,17% dibanding dengan tanpa seleksi fitur. Akurasi terbaik 98,47% dicapai dengan 15 fitur dari 47.
---	--	---	--

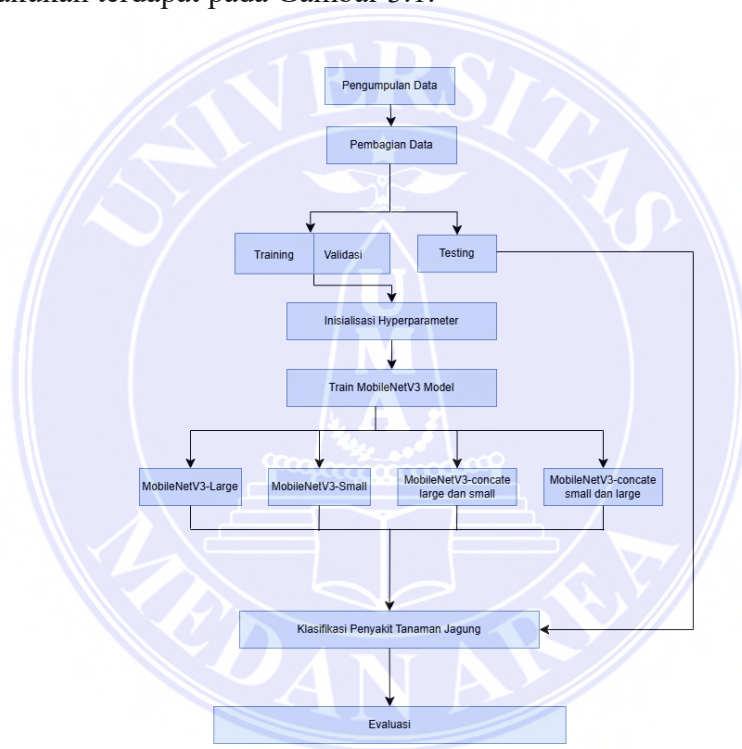


BAB III

METODOLOGI PENELITIAN

3.1 Skema Penelitian

Dalam mendukung jalannya penelitian ini supaya terarah serta sistematis, dibutuhkan suatu skema desain penelitian yang disusun secara terstruktur, sesuai dengan tujuan yang ingin dicapai. Berikut secara garis besar skema penelitian yang dilakukan terdapat pada Gambar 3.1.



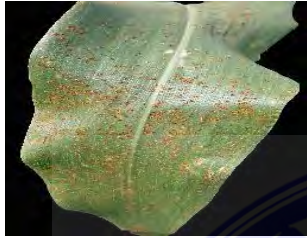


Gambar 3.1 Skema Penelitian

3.2. Pengumpulan Data

Dataset merupakan suatu Kumpulan data-data yang digunakan untuk keberlangsungan dalam penelitian *Dataset* yang digunakan dalam penelitian ini berupa citra gambar daun jagung yang terkena penyakit. Yang diambil dari salah satu petani jagung yang berada di daerah Toba, desa Hutagur-gur 1. *Dataset* tersebut dikumpulkan sebanyak 3000 gambar daun jagung dengan menggunakan kamera

smartphone dan kamera *canon ixus175*. Dataset tersebut dibagi menjadi 3 (tiga) kategori yang masing-masing kategori terdapat 1000 gambar per kategori. Adapun pembagian sampel data dapat dilihat pada Tabel 3.1.

Tabel 3.1 Sampel Data

Citra Daun	Nama Label	Jumlah Data
	Karat Daun	1.000
	Hawar Daun	1.000
	Daun Sehat	1.000

3.3. Pembagian Data

Dataset yang telah diaugmentasi akan terbagi menjadi 3 (tiga) data yaitu data *training*, data validasi dan data *testing*. Data *training* yang bertujuan untuk memahami serta mengenali objek, yang kemudian melakukan inisialisasi parameter dengan hasil yang baik dalam klasifikasi pada data *training* dan data validasi. Setelah melakukan *training* data dan validasi data di tahap evaluasi mulai

menghitung dan mengukur Tingkat keberhasilan yang sesuai dengan ukuran pola yang mendekati pola *training* dan validasi, maka hasil klasifikasi yang berupa jenis penyakit pada daun jagung. *Testing* data langsung dilakukan pada tahap klasifikasi untuk mengetahui hasil dari klasifikasi yang telah dilakukan pada data *training* dan data *testing*.

Data yang akan digunakan sebagai data *training* sebanyak 80%, data yang digunakan untuk data validasi sebanyak 10% dan data *testing* yang akan digunakan sebanyak 10%. Berikut adalah pembagian sampel data pada penyakit tanaman jagung.

Tabel 3.2 *Split Dataset*

No	Kelas	Dataset		
		<i>Training</i> (80%)	<i>Validasi</i> (10%)	<i>Testing</i> (10%)
1	Penyakit Karat Daun	800	100	100
2	Penyakit Hawar Daun	800	100	100
3	Daun Sehat	800	100	100

3.4. Augmentasi Data

Augmentasi data dilakukan untuk meningkatkan keragaman data sebab variasi objek yang agak terbatas serta kapasitas untuk mengumpulkan data di lokasi penelitian. Augmentasi data merupakan suatu Teknik untuk memanipulasi data tanpa kehilangan inti atau esensinya. Augmentasi yang dilakukan ialah *horizontal flip*, *rotation_range* 10, *zoom_range* 0.1, dan *brightness* 0.8, 1.2. *Horizontal flip* yang berfungsi untuk meningkatkan keberagaman dataset dengan membalik

gambar secara horizontal. Fungsi ini dilakukan pada semua gambar sebelum melakukan augmentasi dengan fungsi lainnya. Zoom yang berfungsi untuk memperbesar citra. *Brightness* digunakan untuk mengubah kecerahan pada gambar dataset, sehingga dapat meningkatkan kualitas pelatihan model dengan memberikan lebih banyak konteks dan variasi.

3.5. Hyperparameter

Selama proses *training*, ada beberapa parameter yang digunakan dalam penelitian ini terdapat pada tabel dibawah ini.

Tabel 3.3 Hyperparameter

No	Hyperparameter	Nilai
1	Optimizer	Adam
2	Batch size	32
3	Epoch	50
4	Dropout	0.5

3.6. Train MobileNetV3 Model

Pada tahap ini melakukan *train model* pada *MobileNetV3* yang dilakukan sebanyak 8 (delapan) scenario, yaitu *train model MobileNet-Large* dengan menggunakan *dropout*, *train model MobileNetV3-Large* tanpa menggunakan *dropout*, selanjutnya *train model MobileNetV3-Small* dengan menggunakan *dropout*, *train model MobileNetV3-Small* tanpa menggunakan *dropout*, *train model MobileNet concate Large dan Small* dengan menggunakan *dropout*, *train model*

MobileNetV3 concate Small dan Large dengan menggunakan dropout, train model
MobileNetV3 concate Large dan Small tanpa menggunakan dropout, train model
MobileNetV3 concate Small dan Large tanpa menggunakan dropout.

3.7. Evaluasi

Untuk mengetahui kualitas dari system bisa dilihat dari nilai *confussion matrix* yang bertujuan untuk menganalisis tingkat *accuracy, precision, recall*, dan *F1-Score*. Untuk memahami metriks yang digunakan sebelumnya akan didefinisikan terlebih dahulu bahwa *true positif* (TP) didefinisikan sebagai data positif yang diprediksi positif, *true negative* (TN) yang didefinisikan sebagai data negative yang diprediksi negative, *false positif* (FP) yang didefinisikan sebagai data negative yang diprediksi positif, dan *false negative* (FN) yang didefinisikan sebagai data positif yang diprediksi negative. Adapun kelas *confussion matrix* dapat dilihat pada tabel dibawah ini:

Tabel 3.4 *Confussion Matrix*

Confussion Matrix		Kelas Sebenarnya	
		Positif	Negatif
Prediksi	Positif	True Positif	False Positif
	Negatif	False Negatif	True Negatif

Dibawah ini menunjukkan rumus perhitungan dari *accuracy*, *precision*, *recall*, dan *f1-score*:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1-Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

3.8. Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada semester genap tahun akademik 2023/2024. Serta lokasi penelitian berada di salah satu petani jagung di daerah Toba, Desa Hutagur-gur 1, kecamatan Silaen, Kabupaten Toba, Provinsi Sumatera Utara.

3.9. Spesifikasi Perangkat

Berikut ini adalah perangkat keras dan perangkat lunak yang digunakan dalam penelitian ini, yaitu:

1. Perangkat Keras

Adapun perangkat keras serta spesifikasi yang dipakai adalah sebagai berikut:

- Laptop *Acer* dengan *processor AMD Athlon Silver 3050U with Radeon Grapics*.
- *CPU 2.30GHz*.
- *Random Access Memory (RAM) 4 GB*.
- *Storage 254 GB*.

2. Perangkat Lunak

Adapun perangkat lunak yang dipakai dalam penelitian ini adalah:

- *Microsoft Windows 11 tipe 64-bit* sebagai system operasi.
- *Google collab* sebagai *tools* menjalankan program.
- *Python* sebagai bahasa pemrograman.

BAB V

KESIMPULAN dan SARAN

5.1. Kesimpulan

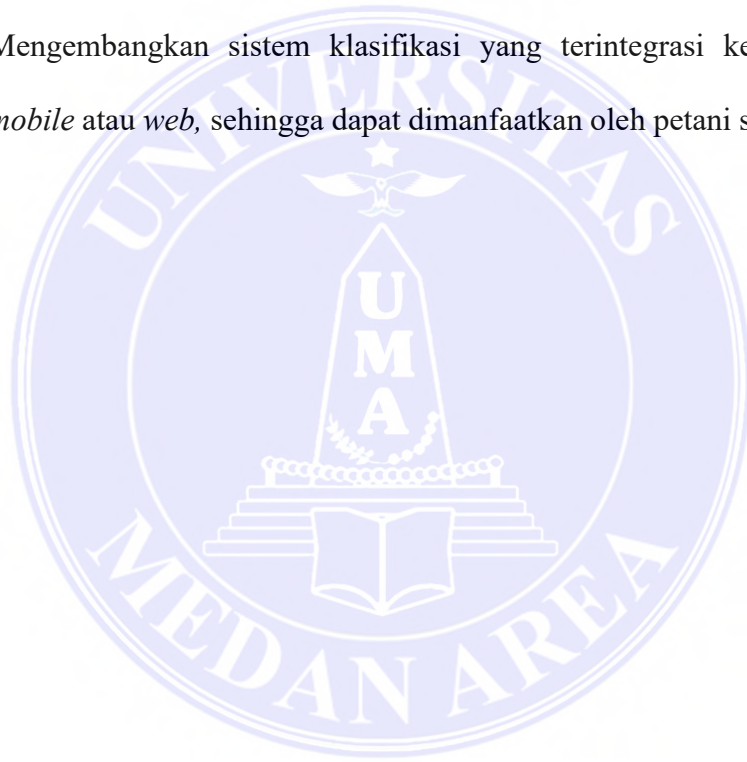
Adapun kesimpulan yang dapat diambil dari hasil penerapan algoritma *MobileNetV3* dalam klasifikasi penyakit tanaman jagung sebagai berikut:

1. Penerapan algoritma *MobileNetV3* dalam klasifikasi penyakit tanaman jagung memberikan hasil yang baik terhadap *dataset*, sehingga dapat menghasilkan klasifikasi jenis penyakit pada daun jagung dengan akurat.
2. Berdasarkan hasil *training* yang telah dilakukan pada model yang diuji, masing-masing model yaitu, *MobileNetV3-Large* dengan menggunakan *dropout* mendapatkan hasil akurasi 0,9500, *MobileNetV3-Small* dengan menggunakan *dropout* sebesar 0,9433, *MobileNetV3-Large* tanpa *dropout* mendapatkan hasil akurasi sebesar 0,9433, *MobileNetV3-Small* tanpa *dropout* sebesar 0,9233, *concat MobileNetV3-Large* dan *Small* dengan menggunakan *dropout* mendapatkan hasil akurasi 0,9467, *concat MobileNetV3-Small* dan *Large* dengan menggunakan *dropout* sebesar 0,9567, *concat MobileNetV3-Large* dan *Small* tanpa *dropout* dengan akurasi 0.9400, dan *concat MobileNetV3-Small* dan *Large* tanpa *dropout* mendapatkan hasil akurasi sebesar 0.9467.

5.2. Saran

Adapun saran yang dapat diberikan oleh penulis untuk pengembangan berikutnya adalah:

1. Untuk penelitian selanjutnya dapat menggunakan *dataset* yang lebih banyak dan beragam, tidak hanya terbatas pada tiga jenis penyakit daun jagung.
2. Melakukan klasifikasi dengan menyisipkan model *zero-shot* kedalam model *MobileNetV3* atau kedalam model *deep learning* lainnya, dan Mengembangkan sistem klasifikasi yang terintegrasi kedalam aplikasi *mobile* atau *web*, sehingga dapat dimanfaatkan oleh petani secara langsung.



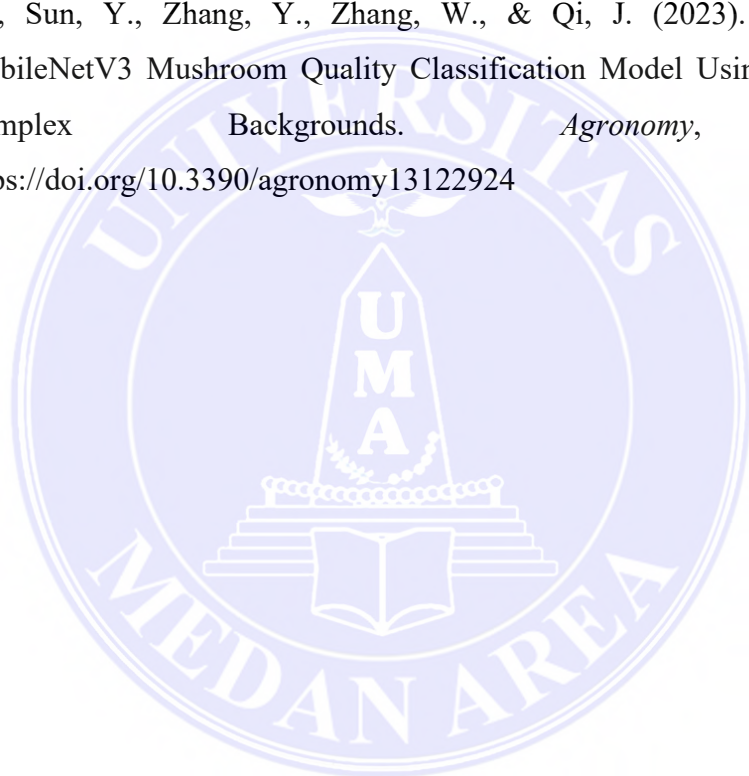
DAFTAR PUSTAKA

- Abd Elaziz, M., Dahou, A., Alsaleh, N. A., Elsheikh, A. H., Saba, A. I., & Ahmadein, M. (2021). Boosting covid-19 image classification using mobilenetv3 and aquila optimizer algorithm. *Entropy*, 23(11), 1–17. <https://doi.org/10.3390/e23111383>
- Alfarizi, M. R. S., Al-farish, M. Z., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2023). Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning. *Karya Ilmiah Mahasiswa Bertauhid (KARIMAH TAUHID)*, 2(1), 1–6.
- Amin, H., Darwish, A., Hassanien, A. E., & Soliman, M. (2022). End-to-End Deep Learning Model for Corn Leaf Disease Classification. *IEEE Access*, 10, 31103–31115. <https://doi.org/10.1109/ACCESS.2022.3159678>
- Azizah, Q. N. (2023). Klasifikasi Penyakit Daun Jagung Menggunakan Metode Convolutional Neural Network AlexNet. *Sudo Jurnal Teknik Informatika*, 2(1), 28–33. <https://doi.org/10.56211/sudo.v2i1.227>
- Budiarto, M. R. W., & Utamaningrum, F. (2023). Sistem Klasifikasi Permukaan Jalan dan Penghindaran Jalan Berlubang pada Kursi Roda Pintar dengan Metode MobileNetV3-Small. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 7(5), 2249–2254.
- Fajar, A. K., & Mutaqin, M. Z. (2024). *KLASIFIKASI KANKER PAYUDARA MENGGUNAKAN ALGORITMA NEURAL NETWORK DAN RANDOM FOREST*. 7, 74–80.
- Iswantoro, D., & Handayani UN, D. (2022). Klasifikasi Penyakit Tanaman Jagung Menggunakan Metode Convolutional Neural Network (CNN). *Jurnal Ilmiah Universitas Batanghari Jambi*, 22(2), 900. <https://doi.org/10.33087/jiubj.v22i2.2065>
- Jiang, Y., & Tong, W. (2022). Improved lightweight identification of agricultural diseases based on MobileNetV3. *CAIBDA 2022 - 2nd International Conference on Artificial Intelligence, Big Data and Algorithms*, 732–736.

- Khultsum, U., & Taufik, G. (2023). Komparasi Kinerja DenseNet 121 dan MobileNet untuk Klasifikasi Citra Penyakit Daun Kentang. *JURIKOM (Jurnal Riset Komputer)*, 10(2), 558. <https://doi.org/10.30865/jurikom.v10i2.6047>
- Kusuma, J., Rubianto, Rosnelly, R., Hartono, & Hayadi, B. H. (2023). Klasifikasi Penyakit Daun Pada Tanaman Jagung Menggunakan Algoritma Support Vector Machine, K-Nearest Neighbors dan Multilayer Perceptron. *Journal of Applied Computer Science and Technology*, 4(1), 1–6. <https://doi.org/10.52158/jacost.v4i1.484>
- Lesmana Hermawan, A. (2023). *JIIFKOM (Jurnal Ilmiah Informatika & Komputer) STTR Cepu Vol. 2 No. 02 (2023) Hal Klasifikasi Penyakit Daun Jagung Menggunakan Lightweight Convolutional Neural Network*. 2(02), 1-7.
- Matthew Patrick, Chairisni Lubis, A. B. D. (n.d.). *AJAH DI SMARTPHONE MENGGUNAKAN MOBILENETV3-SMALL*.
- Nugroho, A. D., & Baihaqi, W. M. (2023). Improved YOLOv5 with Backbone Replacement to MobileNet V3s for School Attribute Detection. *Sinkron*, 8(3), 1944–1954. <https://doi.org/10.33395/sinkron.v8i3.12702>
- Patel, J., & Modi, K. (2024). *Indian Food Image Classification and Recognition with Transfer Learning Technique Using MobileNetV3 and Data Augmentation*. 197. <https://doi.org/10.3390/asec2023-15341>
- Prima, A., Santoso, D. B., & Nurpulaela, L. (2022). Deteksi Otomatis Nominal Uang Kertas Rupiah Untuk Tunanetra Menggunakan Algoritma Arsitektur Ssd Mobiiilenetv3. *Teknokom*, 6(2), 151–159. <https://doi.org/10.31943/teknokom.v6i2.166>
- Rahmanita, E., Negara, Y. D. P., Kustiyahningsih, Y., Sasmeka, V., & Khotimah, B. K. (2023). Implementasi Metode Naïve Bayes dan Information Gain Untuk Klasifikasi Penyakit dan Hama Tanaman Jagung. *Teknika*, 12(3), 198–204. <https://doi.org/10.34148/teknika.v12i3.684>

- Rochmawati, N., Hidayati, H. B., Yamasari, Y., Tjahyaningtjas, H. P. A., Yustanti, W., & Prihanto, A. (2021). Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam. *Journal of Information Engineering and Educational Technology*, 5(2), 44–48. <https://doi.org/10.26740/jieet.v5n2.p44-48>
- Sharifani, K., & Amini, M. (2023). Machine Learning and Deep Learning: A Review of Methods and Applications. *World Information Technology and Engineering Journal*, 10(07), 3897–3904. <https://ssrn.com/abstract=4458723>
- Si, H., Wang, Y., Zhao, W., Wang, M., Song, J., Wan, L., Song, Z., Li, Y., Fernando, B., & Sun, C. (2023). Apple Surface Defect Detection Method Based on Weight Comparison Transfer Learning with MobileNetV3. *Agriculture (Switzerland)*, 13(4). <https://doi.org/10.3390/agriculture13040824>
- Suhendra, R., & Juliwardi, I. (2022). Identifikasi dan Klasifikasi Penyakit Daun Jagung Menggunakan Support Vector Machine. *Jurnal Teknologi Informasi*, 1(1), 29–35. <http://jurnal.utu.ac.id/JTI>
- Suherman, B. B. (2021). Sistem Pakar Diagnosa Penyakit Dan Hama Pada Tanaman Jagung Menggunakan Metode Naive Bayes. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 2(3), 390–398. <https://doi.org/10.33365/jatika.v2i3.1251>
- Taye, M. M. (2023). Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions. *Computers*, 12(5). <https://doi.org/10.3390/computers12050091>
- Widianto, B., Utami, E., & Ariatmanto, D. (2023). Identifikasi Penyakit Tanaman Jagung Berdasarkan Citra Daun Menggunakan Convolutional Neural Network. *Techno.Com*, 22(3), 599–608. <https://doi.org/10.33633/tc.v22i3.8425>
- Yogasudha S, Mounika K, Namitha P R, & Priya K M R. (2018). Deep Learning. *International Journal of Research in Engineering, Science and Management*, 10.

- Yoggyanto, A., Maulana, A., & Tri Cahyo, D. A. (2024). Penerapan Metode Convolutional Neural Network (CNN) Dalam Klasifikasi Penyakit Tanaman Jagung. *Prosiding Seminar Nasional Teknologi Dan Sains*, 3(2022), 251–256.
- Zayd, M. H., Oktavian, M. W., Meranggi, D. G. T., Figo, J. A., & Yudistira, N. (2022). Improvement of garbage classification using pretrained Convolutional Neural Network. *Teknologi*, 12(1), 1–8. <https://doi.org/10.26594/teknologi.v0i0.2403>
- Zhu, F., Sun, Y., Zhang, Y., Zhang, W., & Qi, J. (2023). An Improved MobileNetV3 Mushroom Quality Classification Model Using Images with Complex Backgrounds. *Agronomy*, 13(12). <https://doi.org/10.3390/agronomy13122924>



LAMPIRAN

1. Source Code

```
import numpy as np
import pandas as pd
import tensorflow as tf
from tensorflow import keras
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import seaborn as sns
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras import layers, models, Model
from tensorflow.keras.applications import MobileNetV3Large
from tensorflow.keras.applications import MobileNetV3Small
from tensorflow.keras.layers import Dense, Dropout, Concatenate
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping
# Fixed typo in ModelCheckpoint
from tensorflow.keras.optimizers import Adam
from tensorflow.keras import Model
from pathlib import Path
from sklearn.metrics import classification_report, confusion_matrix
import os

!wget https://raw.githubusercontent.com/mrdbourke/tensorflow-deep-learning/main/extras/helper_functions.py

# Import series of helper functions for our notebook
from helper_functions import create_tensorboard_callback,
plot_loss_curves, unzip_data, compare_historys, walk_through_dir,
pred_and_plot

from google.colab import drive
drive.mount('/content/drive')
```

```
BATCH_SIZE = 32
EPOCH = 50

# Define path
dataset_path = "/content/drive/MyDrive/datasets"
image_dir = Path(dataset_path)

# Get filepaths and labels
filepaths = list(image_dir.glob('**/*.JPG')) +
list(image_dir.glob('**/*.jpg')) + list(image_dir.glob('**/*.png')) +
list(image_dir.glob('**/*.jpeg'))
labels = [os.path.split(os.path.split(filepath)[0])[1] for filepath in
filepaths]

# Create dataframe
filepaths = pd.Series(filepaths, name='Filepath').astype(str)
labels = pd.Series(labels, name='Label')
image_df = pd.concat([filepaths, labels], axis=1)

# Split data into train, validation, and test
train_df, temp_df = train_test_split(image_df, test_size=0.2,
shuffle=True, random_state=101)
val_df, test_df = train_test_split(temp_df, test_size=0.5, shuffle=True,
random_state=101)

# Image data generators
train_datagen = ImageDataGenerator(

preprocessing_function=tf.keras.applications.mobilenet_v3.preprocess_in
put,

horizontal_flip=True,
```

```
        rotation_range=10,
        zoom_range=0.1,
        brightness_range=[0.8, 1.2]
    )
    test_val_datagen = ImageDataGenerator(

preprocessing_function=tf.keras.applications.mobilenet_v3.preprocess_in
put
    )

def create_dataset(dataframe, datagen, batch_size = BATCH_SIZE,
shuffle=True):
    dataset = datagen.flow_from_dataframe(
        dataframe,
        x_col='Filepath',
        y_col='Label',
        target_size=(224, 224),
        color_mode='rgb',
        class_mode='categorical',
        batch_size=BATCH_SIZE,
        shuffle=shuffle,
        seed=42
    )
    return dataset

train_dataset = create_dataset(train_df, train_datagen)
val_dataset = create_dataset(val_df, test_val_datagen, shuffle=False)
test_dataset = create_dataset(test_df, test_val_datagen, shuffle=False)

train_ds = tf.data.Dataset.from_generator(
    lambda: train_dataset,
    output_types=(tf.float32, tf.float32),
```

```
        output_shapes=((None, 224, 224, 3), (None,
len(train_dataset.class_indices)))
    ).repeat() # Apply repeat to the dataset, not the shape tuple

val_ds = tf.data.Dataset.from_generator(
    lambda: val_dataset,
    output_types=(tf.float32, tf.float32),
    output_shapes=((None, 224, 224, 3), (None,
len(val_dataset.class_indices)))
).repeat() # Apply repeat to the dataset

test_ds = tf.data.Dataset.from_generator(
    lambda: test_dataset,
    output_types=(tf.float32, tf.float32),
    output_shapes=((None, 224, 224, 3), (None,
len(test_dataset.class_indices)))
).repeat() # Apply repeat to the dataset

#Model Large
pretrained_model = tf.keras.applications.MobileNetV3Large(
    input_shape=(224, 224, 3),
    include_top=False,
    weights='imagenet',
    pooling='avg'
)

pretrained_model.trainable = False

# Build the model
inputs = pretrained_model.input
x = tf.keras.layers.Dense(256, activation='relu')(pretrained_model.output)
x = tf.keras.layers.Dropout(0.5)(x)
```

```
x = tf.keras.layers.Dense(256, activation='relu')(x)
x = tf.keras.layers.Dropout(0.5)(x)

outputs = tf.keras.layers.Dense(len(train_dataset.class_indices),
activation='softmax')(x)

model_large = Model(inputs=inputs, outputs=outputs)

model_large.compile(
    optimizer=Adam(0.0001),
    loss='categorical_crossentropy',
    metrics=['accuracy']
)

# Callbacks
checkpoint_path = "Klasifikasi_Penyakit_Tanaman_Jagung.weights.h5"
checkpoint_callback = ModelCheckpoint(checkpoint_path,
    save_weights_only=True,
    monitor="val_accuracy",
    save_best_only=True)

early_stopping = EarlyStopping(monitor="val_loss", patience=10,
restore_best_weights=True)

steps_per_epoch = len(train_df) // BATCH_SIZE
validation_steps = len(val_df) // BATCH_SIZE

# Train the model
history = model_large.fit(
    train_ds,
```



```
steps_per_epoch=steps_per_epoch,
validation_data=val_ds,
validation_steps=validation_steps,
epochs=EPOCH,
callbacks=[
    early_stopping,
    checkpoint_callback]
)

model_large.summary()

import matplotlib.pyplot as plt

# def plot_loss_curves(history):
loss = history.history['loss']
val_loss = history.history['val_loss']

accuracy = history.history['accuracy']
val_accuracy = history.history['val_accuracy']

epochs = range(len(history.history['loss']))
plt.figure(figsize=(12, 7))

# Plot loss
plt.subplot(1, 2, 1)
plt.plot(epochs, loss, label='Training_loss')
plt.plot(epochs, val_loss, label='Validation_loss')
plt.title('Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
```

```
# Plot accuracy
plt.subplot(1, 2, 2) # Changed the subplot index to 2 for the accuracy plot
plt.plot(epochs, accuracy, label='Training_accuracy') # Use 'accuracy'
instead of 'loss'
plt.plot(epochs, val_accuracy, label='Validation_accuracy') # Use
'val_accuracy' instead of 'val_loss'
plt.title('Accuracy') # Changed the title to 'Accuracy'
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()

# plt.tight_layout()
plt.show()

# plot loss curves after training
# plot_loss_curves(history) # You can uncomment this line if you want to
use the function later

pred = model_large.predict(test_dataset)
pred = np.argmax(pred,axis = 1)
# Map the label
labels = (train_dataset.class_indices)
labels = dict((v,k) for k,v in labels.items())
pred = [labels[k] for k in pred]

y_test = list(test_df.Label)

print(classification_report(y_test, pred, digits=4))

import seaborn as sns

fig, ax = plt.subplots(1, 1, figsize=(15, 8))
```

```
sns.heatmap(confusion_matrix(y_test, pred), ax=ax,
xticklabels=labels.values(), yticklabels=labels.values(), annot=True,
            cmap='Blues', annot_kws={"size":16}, fmt='d')
plt.title('Confusion Matrix MobileNetV3_Large', fontsize=16)
plt.xlabel('Predicted Label', fontsize=16)
plt.ylabel('True Label', fontsize=16)

plt.show()

#Model Small
pretrained_model_small = tf.keras.applications.MobileNetV3Small(
    input_shape=(224, 224, 3),
    include_top=False,
    weights='imagenet',
    pooling='avg'
)
pretrained_model.trainable = False

# Build the model
inputs = pretrained_model_small.input
y = tf.keras.layers.Dense(256,
activation='relu')(pretrained_model_small.output)
y = tf.keras.layers.Dropout(0.5)(y)
y = tf.keras.layers.Dense(256, activation='relu')(y)
y = tf.keras.layers.Dropout(0.5)(y)

outputs = tf.keras.layers.Dense(len(train_dataset.class_indices),
activation='softmax')(y)

model_small = Model(inputs=inputs, outputs=outputs)
```

```
model_small.compile(  
    optimizer=Adam(0.0001),  
    loss='categorical_crossentropy',  
    metrics=['accuracy']  
)  
  
# Callbacks  
checkpoint_path = "Klasifikasi_Penyakit_Tanaman_Jagung.weights.h5"  
checkpoint_callback = ModelCheckpoint(checkpoint_path,  
    save_weights_only=True,  
    monitor="val_accuracy",  
    save_best_only=True)  
  
early_stopping = EarlyStopping(monitor="val_loss", patience=10,  
    restore_best_weights=True)  
  
steps_per_epoch = len(train_df) // BATCH_SIZE  
validation_steps = len(val_df) // BATCH_SIZE  
  
# Train the model  
history = model_small.fit(  
    train_ds,  
    steps_per_epoch=steps_per_epoch,  
    validation_data=val_ds,  
    validation_steps=validation_steps,  
    epochs=EPOCH,  
    callbacks=[  
        early_stopping,  
        checkpoint_callback]  
)
```

```
model_small.summary()

import matplotlib.pyplot as plt

# def plot_loss_curves(history):
loss = history.history['loss']
val_loss = history.history['val_loss']

accuracy = history.history['accuracy']
val_accuracy = history.history['val_accuracy']

epochs = range(len(history.history['loss']))
plt.figure(figsize=(12, 7))

# Plot loss
plt.subplot(1, 2, 1)
plt.plot(epochs, loss, label='Training_loss')
plt.plot(epochs, val_loss, label='Validation_loss')
plt.title('Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()

# Plot accuracy
plt.subplot(1, 2, 2) # Changed the subplot index to 2 for the accuracy plot
plt.plot(epochs, accuracy, label='Training_accuracy') # Use 'accuracy'
instead of 'loss'
plt.plot(epochs, val_accuracy, label='Validation_accuracy') # Use
'val_accuracy' instead of 'val_loss'
plt.title('Accuracy') # Changed the title to 'Accuracy'
plt.xlabel('Epochs')
```

```
plt.ylabel('Accuracy')
plt.legend()

# plt.tight_layout()
plt.show()

# plot loss curves after training
# plot_loss_curves(history) # You can uncomment this line if you want to
use the function later

pred = model_small.predict(test_dataset)
pred = np.argmax(pred,axis = 1)
# Map the label
labels = (train_dataset.class_indices)
labels = dict((v,k) for k,v in labels.items())
pred = [labels[k] for k in pred]

y_test = list(test_df.Label)

print(classification_report(y_test, pred, digits=4))

import seaborn as sns

fig, ax = plt.subplots(1, 1, figsize=(15, 8))
sns.heatmap(confusion_matrix(y_test, pred), ax=ax,
            xticklabels=labels.values(), yticklabels=labels.values(), annot=True,
            cmap='Blues', annot_kws={"size":16}, fmt='d')
plt.title('Confusion Matrix MobileNetV3_Small', fontsize=16)
plt.xlabel('Predicted Label', fontsize=16)
plt.ylabel('True Label', fontsize=16)

plt.show()
```

```
#Concate Model
import tensorflow as tf
from tensorflow.keras.layers import Input
# Define inputs
inputs = Input(shape=(224, 224, 3))

# MobileNetV3Large
base_model_large = tf.keras.applications.MobileNetV3Large(
    input_shape=(224, 224, 3),
    include_top=False,
    weights='imagenet',
    pooling='avg'
)
base_model_large.trainable = False
features_large = base_model_large(inputs)

# MobileNetV3Small
base_model_small = tf.keras.applications.MobileNetV3Small(
    input_shape=(224, 224, 3),
    include_top=False,
    weights='imagenet',
    pooling='avg'
)
base_model_small.trainable = False
features_small = base_model_small(inputs)

# Concatenate features
concatenated_features = Concatenate()([features_large, features_small])

# Add Dense layers
x = tf.keras.layers.Dense(256, activation='relu')(concatenated_features)
```

```
x = tf.keras.layers.Dropout(0.5)(x)
x = tf.keras.layers.Dense(256, activation='relu')(x)
x = tf.keras.layers.Dropout(0.5)(x)

# Output layer
outputs = Dense(3, activation='softmax')(x)

# Create model
combined_model = Model(inputs=inputs, outputs=outputs)

# Compile model
combined_model.compile(
    optimizer=tf.keras.optimizers.Adam(0.00001),
    loss='categorical_crossentropy',
    metrics=['accuracy']
)
# Callbacks
checkpoint_path = "Klasifikasi_Penyakit_Tanaman_Jagung.weights.h5"
checkpoint_callback = ModelCheckpoint(checkpoint_path,
    save_weights_only=True,
    monitor="val_accuracy",
    save_best_only=True)

early_stopping = EarlyStopping(monitor="val_loss",
    patience=10,
    restore_best_weights=True)

# Train model
history = combined_model.fit(
    train_ds,
    steps_per_epoch=steps_per_epoch,
    validation_data=val_ds,
    validation_steps=validation_steps,
```



```
epochs=EPOCH,
callbacks=[
    early_stopping,
    checkpoint_callback]
)

combined_model.summary()

import matplotlib.pyplot as plt

# def plot_loss_curves(history):
loss = history.history['loss']
val_loss = history.history['val_loss']

accuracy = history.history['accuracy']
val_accuracy = history.history['val_accuracy']

epochs = range(len(history.history['loss']))
plt.figure(figsize=(12, 7))

# Plot loss
plt.subplot(1, 2, 1)
plt.plot(epochs, loss, label='Training_loss')
plt.plot(epochs, val_loss, label='Validation_loss')
plt.title('Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()

# Plot accuracy
plt.subplot(1, 2, 2) # Changed the subplot index to 2 for the accuracy plot
```

```
plt.plot(epochs, accuracy, label='Training_accuracy') # Use 'accuracy'
instead of 'loss'
plt.plot(epochs, val_accuracy, label='Validation_accuracy') # Use
'val_accuracy' instead of 'val_loss'
plt.title('Accuracy') # Changed the title to 'Accuracy'
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()

# plt.tight_layout()
plt.show()

# plot loss curves after training
# plot_loss_curves(history) # You can uncomment this line if you want to
use the function later

pred = combined_model.predict(test_dataset)
pred = np.argmax(pred,axis = 1)
# Map the label
labels = (train_dataset.class_indices)
labels = dict((v,k) for k,v in labels.items())
pred = [labels[k] for k in pred]

y_test = list(test_df.Label)

print(classification_report(y_test, pred, digits=4))

import seaborn as sns

fig, ax = plt.subplots(1, 1, figsize=(15, 8))
sns.heatmap(confusion_matrix(y_test, pred), ax=ax,
xticklabels=labels.values(), yticklabels=labels.values(), annot=True,
```

```
cmap='Blues', annot_kws={"size":16}, fmt='d')  
plt.title('Confusion Matrix Concate Model', fontsize=16)  
plt.xlabel('Predicted Label', fontsize=16)  
plt.ylabel('True Label', fontsize=16)  
  
plt.show()
```



2. Sk Pembimbing



UNIVERSITAS MEDAN AREA FAKULTAS TEKNIK

Kampus I : Jalan Kolam Nomor 1 Medan Estate / Jalan Gedung PBSI, Medan 20223
Kampus II : Jalan Sei Serayu Nomor 70 A / Jalan Setia Budi Nomor 79 B, Medan 20112 Telepon : (061) 8225602, 8201994
Fax : (061) 8226331 HP : 0811 607 259 website: www.uma.ac.id Email : unrv.medanarea@uma.ac.id

Nomor : 740/FT/01.10/III/2024

15 Maret 2024

Lampiran : -

Hal : **Pembimbing Tugas Akhir**

Yth. Pembimbing Tugas Akhir

Muhathir S.T, M.Kom (Sebagai Pembimbing)
di Tempat

Dengan hormat, sehubungan telah dipenuhinya persyaratan untuk memperoleh Tugas Akhir dari mahasiswa atas :

Nama : TULUS MARKUS SIAGIAN
NIM : 208160014
Jurusan : TEKNIK INFORMATIKA

Maka dengan hormat kami mengharapkan kesediaan saudara :

Muhathir S.T, M.Kom (Sebagai Pembimbing)

Adapun Tugas Akhir Skripsi berjudul :

"Penerapan Algoritma MobileNetV3 dalam Klasifikasi Penyakit Tanaman Jagung"

SK Pembimbing ini berlaku selama enam bulan terhitung sejak SK ini diterbitkan. Jika proses pembimbing melebihi batas waktu yang telah ditetapkan, SK ini dapat ditinjau ulang.


Demikian kami sampaikan, atas kesediaan saudara diucapkan terima kasih.

Dekan,



Dr Eng. Supriatno.ST, MT.

3. Surat Pengantar Riset

**UNIVERSITAS MEDAN AREA**
FAKULTAS TEKNIK

Kampus I : Jalan Kolam Nomor 1 Medan Estate/Jalan PBSI Nomor 1 (061) 7399878, 7399168, 7394348, 7395781 Fax (061) 7399998 Medan 20223
Kampus II : Jalan Setabuk Nomor 79 / Jalan Sei Serayu Nomor 70 A. (061) 8225992, Fax (061) 8226531 Medan 20122
Website: www.teknik.uma.ac.id E-mail: univ_medanarea@uma.ac.id

Nomor : 295 /FT. 6/01.10/VII/2024
Lamp : -
Hal : -
2 Juli 2024

: **Penelitian Dan Pengambilan Data Tugas Akhir**

Yth. Kepala Desa Hutagur- Gur I
Jln. Sibide
Di
Samosir

Dengan hormat,
Kami mohon kesediaan Bapak/Ibu berkenan untuk memberikan izin dan kesempatan kepada mahasiswa kami tersebut dibawah ini :


NO	NAMA	NPM	PRODI
1	Tulus Markus Siagian	208160014	Teknik Informatika

Untuk melaksanakan Penelitian dan Pengambilan Data Tugas Akhir pada perusahaan/Instansi yang Bapak/Ibu Pimpin.

Perlu kami jelaskan bahwa Pengambilan Data tersebut adalah semata-mata untuk tujuan ilmiah dan Skripsi yang merupakan salah satu syarat bagi mahasiswa tersebut untuk mengikuti ujian sarjana pada Fakultas Teknik Universitas Medan Area dan tidak untuk dipublikasikan, dengan judul penelitian :

Penerapan Algoritma *MobileNetV3* dalam Klasifikasi Penyakit Tanaman Jagung

Atas perhatian dan kerja sama yang baik diucapkan terima kasih.

Dekan,

Dr. Eng. Supriatno, ST, MT

Tembusan :
1. Ka. BPMPP
2. Mahasiswa
3. File

4. Surat Selesai Riset



PEMERINTAH KABUPATEN TOBA
KECAMATAN SILAEN
DESA HUTAGURGUR I

Hutagurgur I, 12 Juli 2024

Nomor : 88 / HTG I-18 / VII / 2024
Lampiran : -
Perihal : **Telah Melakukan Penelitian dan Pengambilan Data**

Kepada Yth,
Dekan Fakultas Teknik
Universitas Medan Area
Di -
Medan, Sumatera Utara

Sehubungan dengan surat saudara Nomor : 295 / FT.6 / 01.10 / VII / 2024 Perihal Permohonan Penelitian dan Pengambilan Data Tugas Akhir di Desa Hutagurgur I, maka dengan ini menerangkan nama Mahasiswa/i dibawah ini sebagai berikut :

Nama : Tulus Markus Siagian
NPM : 208160014
Fakultas : Teknik
Prodi : Teknik Informatika

Bahwa Mahasiswa/i tersebut benar telah selesai melaksanakan Penelitian dan Pengambilan Data di Desa Hutagurgur I untuk mendukung Tugas Akhir (Skripsi) dengan judul "*Penerapan Algoritma Mobile.NetV3 dalam Klafifikasi Penyakit Tanaman Jagung*".

Demikian disampaikan untuk dapat dipergunakan sebagaimana mestinya.

KEPALA DESA HUTAGURGUR I

HENGKY SUBANGUN SIAGIAN