

**PENERAPAN *VISION TRANSFORMER (ViT)* UNTUK
KLASIFIKASI PENYAKIT JAGUNG
MELALUI CITRA DAUN**

SKRIPSI

OLEH:

FITTAULI LUMBAN GAOL

NPM : 208160010



PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MEDAN AREA MEDAN

2024

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

Document Accepted 9/1/25

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

**PENERAPAN *VISION TRANSFORMER (ViT)* UNTUK
KLASIFIKASI PENYAKIT JAGUNG
MELALUI CITRA DAUN**

SKRIPSI

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

OLEH

FITTAULI LUMBAN GAOL

208160010

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS MEDAN AREA

MEDAN

2024

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

Document Accepted 9/1/25

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 (repository.uma.ac.id)9/1/25

HALAMAN PENGESAHAN

Judul Skripsi : PENERAPAN VISION TRANSFORMER (ViT) UNTUK
KLASIFIKASI PENYAKIT JAGUNG MELALUI CITRA DAUN

Nama : Fittauli Lumban Gaol

NPM : 208160010

Fakultas : Teknik

Disetujui Oleh

Komisi Pembimbing



Muhathir, S.T., M.Kom

Pembimbing I



Wahid Pratno, S.T., MT
Dekan



Rizki Muliono, S.Kom, M.Kom
Kaprodi

Tanggal Lulus: 30 Agustus 2024

HALAMAN PERNYATAAN

Saya menyatakan bahwa skripsi yang saya susun, sebagai syarat memperoleh gelar sarjana merupakan hasil karya 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 peraturan yang berlaku, apabila di kemudian hari ditemukan adanya plagiat dalam skripsi ini.

Medan, 30 Agustus 2024



Fittauli Lumban Gaol
Npm: 208160010

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 : Fittauli Lumban Gaol

NPM : 208160010

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 Vision Transformer (ViT) Untuk Klasifikasi Penyakit Jagung Melalui Citra Daun.

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



Fittauli Lumban Gaol

Npm: 208160010

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

Document Accepted 9/1/25

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 (repository.uma.ac.id) 9/1/25

ABSTRAK

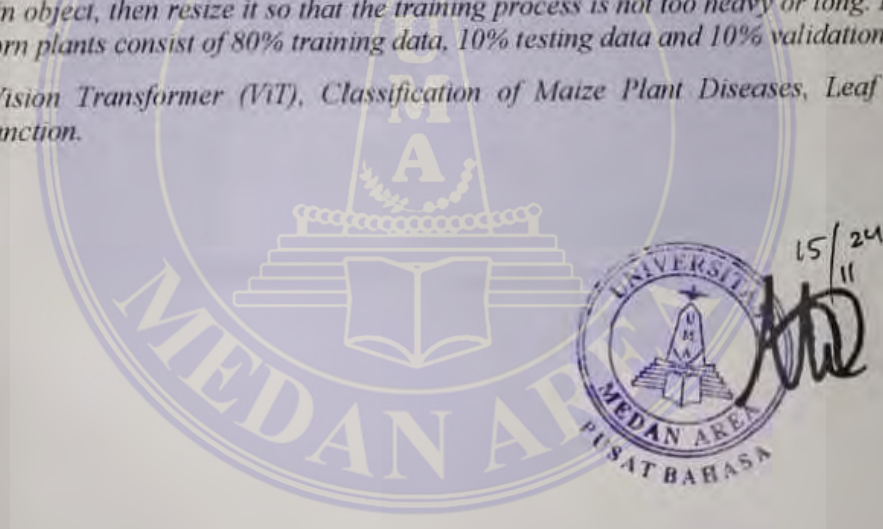
Penelitian ini bertujuan untuk menerapkan algoritma *Vision Transformer (ViT)* dalam mengklasifikasi penyakit pada tanaman jagung melalui citra daun. Penyakit pada tanaman jagung dapat menyebabkan kerugian yang signifikan bagi petani, sehingga diperlukan sistem yang dapat mendeteksi penyakit secara akurat. Dalam penelitian ini, dataset yang digunakan terdiri dari 3000 data citra daun jagung yang dibagi menjadi tiga kelas, yaitu daun hawar, daun karat, dan daun sehat. Pengujian model ViT dilakukan dengan menggunakan lima fungsi aktivasi yang berbeda yaitu *GELU*, *ELU*, *RELU*, *PRELU*, dan *Leaky RELU*. Hasil penelitian menunjukkan bahwa model *ViT* dengan fungsi aktivasi *PRELU* memperoleh akurasi tertinggi sebesar 94,33% diikuti dengan fungsi aktivasi *RELU* dan *Leaky RELU* dengan akurasi sebesar 93,67%. Dalam penelitian ini, data yang digunakan adalah gambar penyakit pada daun jagung sebanyak 3000 data, data tersebut dibagi menjadi 3 bagian yaitu data Training, data Testing dan *validasi*, lalu data tersebut dilakukan augmentasi seperti *propping* dan *resize*, *crop* sampel dilakukan agar fokus ke daun sebagai objek utama, lalu dilakukan *resize* supaya pada proses training tidak begitu berat atau lama, sampel penyakit pada tanaman jagung terdiri dari 80% data training dan 10% data testing dan 10% data validasi.

Kata Kunci: *Vision Transformer (ViT)*, Klasifikasi Penyakit Tanaman Jagung, Citra Daun, Fungsi Aktivasi.

ABSTRACT

This research aimed to apply the Vision Transformer (ViT) algorithm in classifying diseases in corn plants through leaf images. classifying diseases in corn plants through leaf images. Diseases in corn plants can cause significant losses for farmers, so a system that can accurately detect diseases is needed. that can detect diseases accurately. In this research, the dataset used dataset consists of 3000 corn leaf image data divided into three classes, namely leaf blight, leaf rust, and healthy leaves. rust, and healthy leaves. ViT model testing was conducted using five different activation functions activation functions, namely GELU, ELU, RELU, PRELU, and Leaky RELU. Research results. The results showed that the ViT model with the PRELU activation function obtained the highest accuracy of 94.33% followed by the PRELU activation function. 94.33% followed by RELU and Leaky RELU activation functions with an accuracy of 93.67%. In this research, the data used is 3000 images of diseases on corn leaves, the data is divided into 3 parts, namely training data, testing data and validation, then the data is augmented such as propping and resizing, sample cropping is done so that the focus is on the leaves as main object, then resize it so that the training process is not too heavy or long. Disease samples in corn plants consist of 80% training data, 10% testing data and 10% validation data.

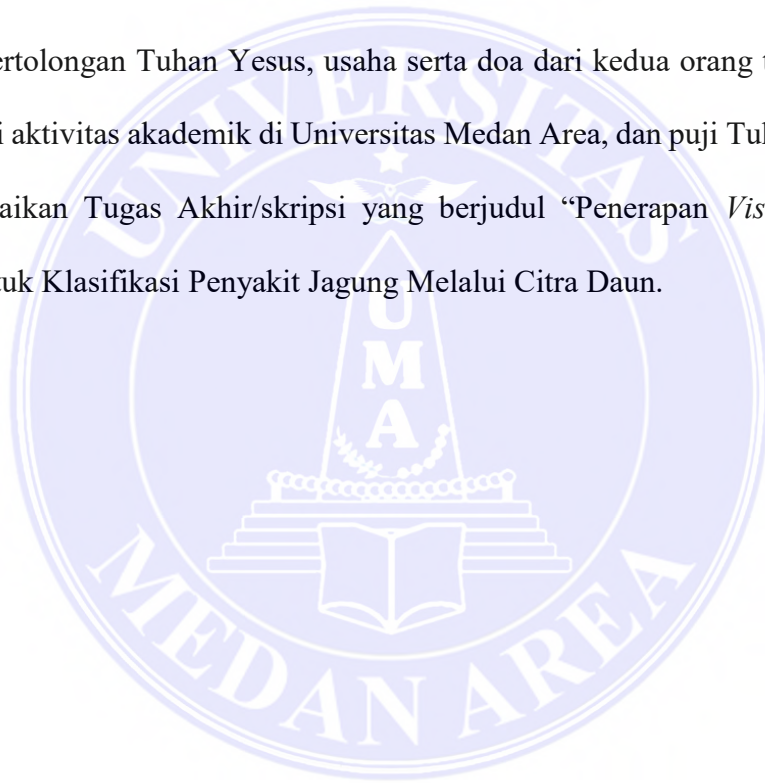
Keywords: *Vision Transformer (ViT), Classification of Maize Plant Diseases, Leaf Image, Activation Function.*



RIWAYAT HIDUP

Penulis bernama lengkap Fittauli Lumban Gaol, lahir di Desa Bukit Mengkirai, Kabupaten Langkat, pada 18 Agustus 2001 anak kelima dari 7 bersaudara dari Ayah Sudung Lumban Gaol dan Ibu Fatima Br Nababan. Tahun 2020 penulis lulus dari SMK NEGERI 1 TANJUNG PURA, Kabupaten Langkat. Dan pada 2020 terdaftar sebagai mahasiswa Program Studi Teknik Informatika Universitas Medan Area.

Berkat pertolongan Tuhan Yesus, 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 yang berjudul “Penerapan *Vision Transformer (ViT)* Untuk Klasifikasi Penyakit Jagung Melalui Citra Daun.



Motto

“In The Name Of Jesus Christ”

Tuhan menyediakan lebih dari yang pernah kamu sebutkan dalam doamu, ia bahkan sudah tau apa yang kita perlukan sebelum kita mengucapkannya. Dan direndahkan dimata manusia, ditinggikan dimata Tuhan, *“Prove Them Wrong”*.

“Selalu ada harga dalam sebuah proses. Nikmati saja lelah-lelah itu, lebarkan lagi rasa sabar itu, semua yang kau investasikan untuk menjadikan dirimu serupa yang kau impikan itu, mungkin tidak akan selalu berjalan lancar. Tapi gelombang-gelombang itu yang nanti akan bisa kau ceritakan suatu saat nanti....Karena masa depan sungguh ada, dan harapanmu tidak akan hilang (Amsal 23:18).

Janganlah hendaknya kamu kuatir tentang apapun juga, tetapi nyatakanlah dalam segala hal keinginan mu kepada Allah dalam doa dan permohonan dengan ucapan Syukur (filipi 4:6).

Medan, 30 Agustus 2024



Fittauli Lumban Gaol

Npm: 208160010

KATA PENGANTAR

Puji syukur kepada Tuhan yang Maha Esa atas berkat dan rahmat-nya sehingga penulis dapat menyelesaikan skripsi dengan judul *“Penerapan Vision Transformer (ViT) Untuk Klasifikasi Penyakit Tanaman Jagung Melalui Citra Daun”* ini dengan baik.

Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan Program Strata-1 pada Fakultas Teknik Program Studi Teknik Informatika di Universitas Medan Area. Dalam proses menyelesaikan skripsi ini, penulis menyadari bahwa tulisan ini masih jauh dari kata sempurna dan juga terdapat banyak kekurangan. Penulis juga mengharapkan kritik dan saran yang sifatnya membangun dari pada pembaca. Kemudian penulis ingin mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Dadan Ramdan, M.Eng., M.Sc., selaku Rektor Universitas Medan Area.
2. Bapak Dr.Eng. Supriatno,S.T., M.T. selaku Dekan Fakultas Teknik Universitas Medan Area.
3. Bapak Rizki Muliono, S.Kom., M.Kom., selaku Kepala Program Studi Teknik Informatika dan sekaligus sekretaris.
4. Bapak Muhathir, S.T. M.Kom. selaku dosen pembimbing yang telah meluangkan waktu untuk membimbing dan memberikan arahan kepada saya dengan penuh kesabaran kepada penulis dalam proses menyelesaikan skripsi ini. Bapak Sayuti Rahman, ST, M.Kom. Dr. sebagai ketua, dan Ibu Nurul Khairina, S.Kom, M. Kom. Sebagai pibanding II yang telah memberikan

saran dan masukan dalam skripsi penulis dan meluangkan waktunya menjadi panitia dalam seminar proposal sampai penulis sidang.

5. Seluruh Dosen Teknik Informatika Universitas Medan Area yang selama ini telah membekali penulis dengan ilmu yang sangat bermanfaat.
6. IT Support Teknik Informatika bang Robby Kurniawan Sari Damanik, ST yang telah banyak membantu penulis dalam menyiapkan pemberkasan yang diperlukan dari awal sampai selesai.
7. Teristimewa kepada orang tua penulis, Alm Bapak Sudung Lumban Gaol dan Ibu Fatima Br Nababan yang selalu memberikan kasih sayang, semangat, pelukan dan doa serta motivasi yang sangat luar biasa kepada penulis. Dan sosok orang tua yang berhasil membuat saya bangkit dari kata menyerah. Penulis sadar, bahwa setiap kata dalam skripsi ini adalah buah dari kerja keras penulis dan serta doa orang tua penulis. Skripsi ini adalah persembahan untukmu dari putrimu yang sudah tumbuh dewasa ini, terimakasih selalu ada buat saya memperjuangkan saya dengan sepuh hati dan tanpa rasa lelah sampai saya bisa mendapatkan gelar sarjana ini. Dan semoga Bapak tenang di alam sana, dan semoga mama hidup lebih lama lagi bersama penulis, Amin.
8. Kepada kakak penulis tersayang Rita Natalia Br Lumban Gaol S.Pd, Hariaty Lumban Gaol, Fitri Lumban Gaol, Irma Yanti Lumban Gaol,S.AK dan Adik-adik penulis tersayang Elsa Br Lumban Gaol, Sadar Parulian Lumban Gaol yang selalu mendukung dan membantu serta mendoakan penulis sehingga berhasil menyelesaikan pendidikan saat ini.

9. Tulus Markus Siagian, Winda Pramita Hutagalung dan Teman-teman Teknik Informatika 2020, terima kasih atas persahabatan dan persaudaraannya selama ini. Semoga Tuhan memudahkan untuk menyelesaikan study S-1 ini dan semoga kita bisa sukses bersama di akan datang.
10. Dan yang paling terakhir dan paling penting saya sendiri mengucapkan terima kasih kepada diri saya sendiri yaitu Fittauli Lumban Gaol S.Kom yang telah berusaha kuat dan bertahan atas segala perjuangan, air mata, dan ketidakpastian di perjalanan panjang ini, meskipun sering kali ingin menyerah dan merasa putus asa. Terima kasih sudah melibatkan Tuhan Yesus Kristus dalam setiap perjalananmu dan mengizinkan Yesus untuk menjadi batu sandaranmu. Berbanggalah kepada dirimu sendiri karena telah menjadi pahlawan dalam cerita hidupmu sendiri. Apapun kurang dan lebihmu semoga kedepannya menjadi seorang yang sukses dan menggapai cita-cita, Amin.

Medan, 30 Agustus 2024

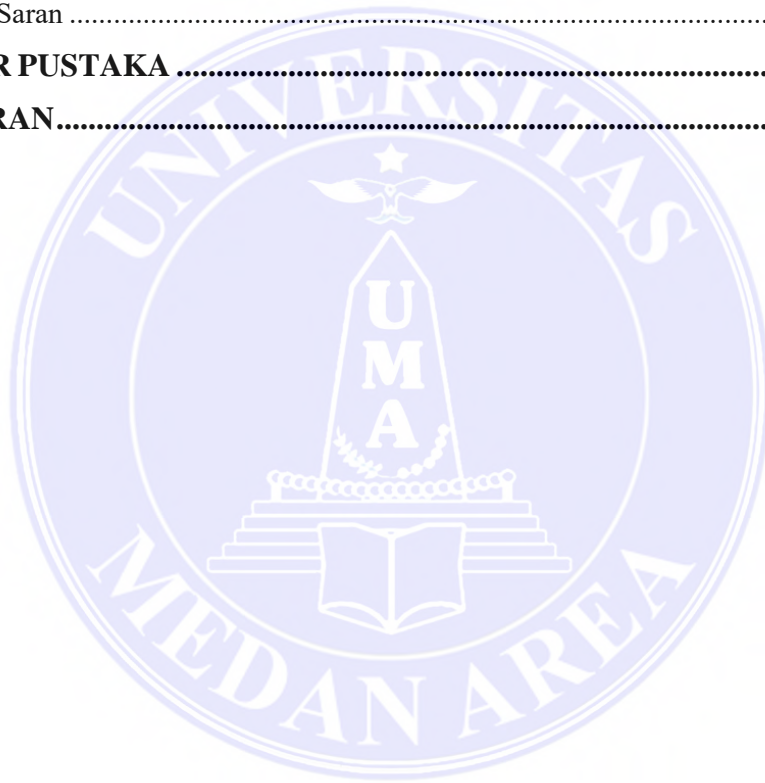


Fittauli Lumban Gaol
Npm: 208160010

DAFTAR ISI

HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN	i
ABSTRAK	ii
RIWAYAT HIDUP	iv
Motto	v
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian.....	6
1.4 Manfaat Penelitian	6
1.5 Batasan Masalah.....	6
1.6 Sistematika Penulisan	7
BAB II LANDASAN TEORI	8
2.1 <i>Deep Learning</i>	8
2.2 Klasifikasi	9
2.3 <i>Vision Transformer</i>	9
2.4 Penyakit Jagung.....	12
2.5 Penelitian Terdahulu.....	13
BAB III METODOLOGI PENELITIAN	20
3.3 Teknik Pengumpulan Data.....	22
3.4 Analisis Data	22
3.5 <i>Hyperparameter</i>	23
3.6 Evaluasi.....	30
BAB IV HASIL DAN PEMBAHASAN	33
4.1 Hasil	33
4.1.1 Visualisasi Data	33
4.1.3 Pengujian Model <i>Vision Transformer</i> Menggunakan Fungsi Aktivasi <i>GELU</i>	35
4.1.4 Pengujian Model <i>Vision Transformer</i> Menggunakan Fungsi Aktivasi	

<i>ELU</i>	38
4.1.5 Pengujian Model <i>Vision Transformer</i> Menggunakan Fungsi Aktivasi <i>RELU</i>	41
4.1.6. Pengujian Model <i>Vision Transformer</i> Menggunakan Fungsi Aktivasi <i>Parametrik RELU</i> (PRELU).....	44
4.1.7 Pengujian Model <i>Vision Transformer</i> Menggunakan Fungsi Aktivasi <i>Leaky RELU</i>	47
4.2. Pembahasan.....	49
BAB V KESIMPULAN dan SARAN	53
5.1. Kesimpulan	53
5.2. Saran	54
DAFTAR PUSTAKA	55
LAMPIRAN	58



DAFTAR GAMBAR

Gambar 2.1	Arsitektur <i>Vision Transformer</i>	11
Gambar 3.1	Kerangka Penelitian.....	20
Gambar 3.2	Arsitektur Model <i>Vision Transformer</i>	21
Gambar 4.1	Visualisasi Data Gambar Daun Jagung (a) Daun Hawar,	33
Gambar 4.2	<i>Augmentasi data</i> (a) <i>width shift range</i> 0.2, (b) <i>height sift range</i> 0.2,.....	34
Gambar 4.3	Grafik <i>Training Model</i> dan <i>Validation Model</i> fungsi aktivasi <i>GELU</i>	35
Gambar 4.4	<i>Confussion Matrix</i> fungsi aktivasi <i>GELU</i>	36
Gambar 4.5	Grafik <i>Training Model</i> dan <i>Validation Model</i> fungsi aktivasi <i>ELU</i>	38
Gambar 4.6	<i>Confussion Matrix</i> fungsi aktivasi <i>ELU</i>	39
Gambar 4.7	Grafik <i>Training Model</i> dan <i>Validation Model</i> fungsi aktivasi <i>RELU</i>	41
Gambar 4.8	<i>Confussion Matrix</i> fungsi aktivasi <i>RELU</i>	42
Gambar 4.9	Grafik <i>Training Model</i> dan <i>Validation Model</i> fungsi aktivasi <i>PRELU</i>	44
Gambar 4.10	<i>Confussion Matrix</i> fungsi aktivasi <i>PRELU</i>	45
Gambar 4.11	Grafik <i>Training Model</i> dan <i>Validation Model</i>	47
Gambar 4.12	<i>Confussion Matrix</i> fungsi aktivasi <i>Leaky RELU</i>	48

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	13
Tabel 2.2 Penelitian Terkait	15
Tabel 3.1 <i>Split Dataset</i>	23
Tabel 3.2 <i>Confussion Matrix</i>	31
Tabel 4.1 <i>Classification report</i> fungsi aktivasi <i>GELU</i>	37
Tabel 4.2 <i>Classification Report</i> fungsi aktivasi <i>ELU</i>	40
Tabel 4.3 <i>Classification Report</i> fungsi aktivasi <i>RELU</i>	43
Tabel 4.4 <i>Classification Report</i> fungsi aktivasi <i>PRELU</i>	46
Tabel 4.5 <i>Classification Report</i> fungsi aktivasi <i>Leaky RELU</i>	49
Tabel 4.6 Temuan Penelitian.....	50
Tabel 4.7 Perbandingan dengan penelitian terkait	51



BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam bidang pertanian, kehadiran penyakit daun bisa merugikan produksi dan perkembangan tanaman, khususnya tanaman jagung. Di Indonesia pekerjaan utama masyarakatnya ialah seorang petani, karena berada di wilayah tropis, kondisi iklim, tanah, dan sumber daya lainnya diseluruh negeri memiliki potensi yang besar untuk pengembangan pertanian (Prakosa dkk., 2023). Jagung merupakan salah satu sector pertanian yang penting dalam perekonomian global maupun nasional. karena sumber daya pangan sebagai bahan makanan, dan bahan baku lainnya bagi masyarakat (Firgia & Thomas, 2023). Menurut (Fraivan dkk., 2022) dalam penelitiannya yang berjudul klasifikasi penyakit jagung dari gambar daun menggunakan deep Transfer learning, mengatakan bahwa pada tahun 2020, produksi jagung diseluruh dunia mencapai 1,15 miliar ton. Banyak penyakit secara alami dapat menyerang berbagai bagian tanaman, seperti daun, batang, dan buah, selama tahap pertumbuhan. Hal ini berdampak langsung pada hasil panen jagung dan dapat menyebabkan kerugian besar. Penyakit yang paling berbahaya adalah yang menginfeksi daun jagung selama proses pertumbuhan.

Dengan meningkatnya permintaan jagung dipasar domestik, Indonesia memiliki peluang untuk menyeimbangkan permintaan dan penawaran jagung domestik. Salah

satu cara untuk mencapai keseimbangan ini adalah dengan memproduksi jagung sendiri dengan menggunakan sumber daya domestic atau dengan mengimpor jagung dari negara lain, menurut penelitian yang dilakukan oleh (Suherman, 2021) menyatakan bahwa, Beberapa penyakit yang menyerang tanaman jagung adalah Hawar daun, busuk pelapah, bulai, dan busuk tongkol. Penyakit bulai pada jagung telah lama dianggap menimbulkan kerugian yang signifikan, sehingga para petani sangat mengetahuinya. Penyakit bulai pada jagung dapat menyebabkan kerugian yang bervariasi. Kemungkinan mengalami kehilangan 90%. Penyakit tumbuhan jagung yang signifikan termasuk penyakit karat daun oleh jamur *puccinia sorghi* Schwein dan penyakit hawar daun oleh jamur *Helmithosporium turcicum* (pass.) Leonard et Suggs. Jamur *Puccinia sorghi* menyebabkan karat daun. Tanda pertama adalah bintik-bintik merah dan bubuk berwarna kuning kecokelatan. Penyakit ini menyebabkan tanaman tidak dapat melakukan fotosintesis normal, memperlambat pertumbuhan, atau bahkan mati. *Puccinia sorghi* paling sering hidup dipengunungan tropis dan wilayah dengan iklim sedang. Kerusakan akibat penyakit ini mencapai 70%. Hawar daun adalah penyakit penting pada jagung yang telah menyebar ke banyak negara di Amerika, Asia, Afrika, dan Eropa. Gejala awalnya berupa bercak kecil berbentuk oval, yang kemudian berubah menjadi bercak besar (Yoggyanto dkk., 2024).

Komputer sekarang dapat melakukan pengenalan objek otomatis dengan mengambil dan mengolah data dari gambar berkat kemajuan teknologi komputasi dan prosesnya. Salah satu cara untuk menemukan, mengidentifikasi jenis penyakit daun pada tanaman jagung adalah dengan menggunakan pengolahan gambar ini. teknologi pencarian informasi yang memiliki kapasitas untuk mengumpulkan data besar, dapat

mempercepat dan mempermudah pengenalan data dan mengkategorikan jenis berdasarkan label dengan menggunakan metode data mining, salah satunya adalah klasifikasi (Kusuma dkk., 2023). Sampai saat ini, upaya terus dilakukan untuk menghilangkan hama dan penyakit tanaman pengganggu seperti jagung(Widianto dkk., 2023). Diindustri pertanian, penyakit daun jagung dapat mengganggu pertumbuhan dan produktivitas tanaman,khususnya tanaman jagung. Teknologi deep Learning dapat membantu mencegah penyakit daun dan mengurangi kerugian karena gagal panen sambil meningkatkan kualitas panen. Dengan tingkat keakuratan yang tinggi, untuk mendeteksi dan mengklasifikasikan citra gambar, metode pembelajaran yang diawasi,yang mencakup data uji dan data latih,untuk mengklasifikasi data yang berlabel(Yos Sudarso KM.,dkk 2023).

Menurut (Kusuma dkk., 2023) seorang pakar atau ahli tanaman dapat menemukan,mengidentifikasi,dan menghentikan penyakit pada tanaman jagung. Namun,ini sangat mengganggu petani karena biaya konsultasi yang mahal dan memakan waktu, terutama bagi petani yang lokasinya jauh dari kota. Hal ini dapat mengakibatkan penurunan produksi jagung karena kurangnya informasi tentang penyakit daun pada tanaman jagung. Para petani dapat menggunakan system computer sebagai pengganti tugas seorang ahli tanaman atau pakar dalam menemukan penyakit tanaman pada daun jagung.

Dibutuhkan sistem yang dapat mendeteksi penyakit pada daun jagung. Salah satu system yang dapat mendeteksi penyakit pada daun jagung adalah sistem segmentasi citra yang menggunakan model jaringan syaraf tiruan, yang menggunakan pengolahan gambar digital untuk mempelajari hal-hal yang berkaitan dengan perbaikan kualitas

gambar (seperti meningkatkan kontras, pewarnaan, dan warna), Ada teknik pengolahan citra yang dapat mengidentifikasi penyakit pada daun tanaman (Rachmawanto & Hadi, 2021).

Penelitian yang mengenai penyakit tanaman jagung (Soleymani dkk., 2021), dalam penelitiannya tentang Klasifikasi Bahan Konstruksi pada Penggunaan Kumpulan Data Tidak Seimbang, pada penelitian tersebut menunjukkan hasil akurasi yang baik, selanjutnya penelitian yang dilakukan oleh (Pangestu dkk., 2024), Penelitian ini menjelaskan tentang klasifikasi tingkat kematangan pisang menggunakan vision transformer mendapatkan akurasi sebesar 91,61%, maka penelitian yang dilakukan berjalan dengan baik. Berikutnya dalam penelitian Boyuan Wang, 2022, mengenai Model klasifikasi Spesies jamur otomatis untuk foodborne pencegahan penyakit berdasarkan *Vision Transformer*, juga mendapat hasil akurasi yang baik dengan memvisualisasikan output dimensi dari model *ViT-L/32* untuk mencapai interpretasi. Setelah itu penelitian Yi Yang dan *Xin Li*, 2024 tentang Klasifikasi Bintang dengan *Vision Transformer* dan Gambar Fotometri SDSS mendapatkan hasil yang baik.

1.2 Rumusan Masalah

Adapun yang menjadi rumusan masalah dalam penelitian ini yaitu bagaimana klasifikasi penyakit jagung melalui citra daun dengan menggunakan *Vision Transformer (ViT)*, dengan menggunakan fungsi aktivasi *GELU*, *ELU*, *RELU*, *PARAMETRIK RELU (PRELU)*, *LEAKY RELU*

1. Fungsi aktivasi adalah komponen penting dalam jaringan saraf tiruan (neural network) yang mengatur output dari neuron berdasarkan input yang diterima.

Secara lebih formal, fungsi aktivasi mengambil input dari naural (biasanya hasil dari penjumlahan bobot input dan bias) dan menghasilkan nilai output yang menentukan apakah neuron tersebut aktif atau tidak dalam mengirim sinyal. Fungsi ini mengubah jumlah input yang diterima oleh neuron menjadi output yang diproses lebih lanjut dalam jaringan. Dengan demikian, fungsi aktivasi memperkenalkan non-linearitas kedalam model jaringan saraf, dan memungkinkan jaringan untuk mempelajari dan memodelkan hubungan yang kompleks dan tidak linear dalam data. Terdapat beberapa cara kerja fungsi aktivasi yaitu;

2. Menerapkan fungsi aktivasi pada Z

Setelah mendapatkan nilai Z, neuron menerapkannya ke fungsi aktivasi. Fungsi aktivasi ini akan mengambil nilai Z sebagai input dan menghasilkan output A, yang merupakan nilai yang akan diteruskan ke neuron berikutnya atau sebagai output dari jaringan. Misalnya, jika fungsi aktivasi yang digunakan adalah *ReLU* (*Rectified Linear Unit*), maka:

$$A = f(Z) = \max(0, Z)$$

Ini berarti jika Z bernilai negatif, outputnya adalah 0, sedangkan jika Z bernilai positif, outputnya adalah Z itu sendiri.

3. Menghasilkan Output untuk diteruskan ke lapisan berikutnya

Output A dari fungsi aktivasi ini akan diteruskan ke lapisan berikutnya dalam jaringan saraf. dilapisan berikutnya, neuron akan menerima nilai A sebagai input baru, dan proses diatas berulang sampai mencapai lapisan output.

4. Backpropagation dan pembaruan Bobot

Selama pelatihan, jaringan menggunakan proses *backpropagation* untuk menghitung kesalahan dari output yang dihasilkan dan dibandingkan dengan nilai sebenarnya. Fungsi aktivasi berperan penting dalam proses ini karena kemiringan (*gradien*) dari fungsi aktivasi menentukan bagaimana bobot-bobot diperbarui. *Gradien* menunjukkan seberapa sensitif output neuron terhadap perubahan bobot, dan digunakan untuk menyesuaikan bobot agar kesalahan jaringan diminimalkan.

1.3 Tujuan Penelitian

Salah satu tujuan penelitian ini adalah sebagai berikut:

1. Menerapkan Algoritma *Vision Transformer (Vit)* dalam klasifikasi penyakit jagung melalui citra daun.
2. Menemukan hasil identifikasi penyakit tanaman jagung.

1.4 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah membantu untuk meningkatkan hasil dari produktivitas hasil pertanian jagung.

1.5 Batasan Masalah

Adapun batasan masalah dalam penelitian ini, yaitu:

1. Data yang digunakan sebagai sampel pada penelitian ini adalah data citra gambar pada daun jagung.
2. Keterbatasan sumber daya komputasi, seperti daya baterai dan kekuatan dalam komputasi.

1.6 Sistematika Penulisan

Dalam penulisan skripsi ini, penulis memberikan gambaran sistematika penulisan dalam klasifikasi penyakit pada tanaman jagung dalam citra daun yang terbagi menjadi 5 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 PUSATAKA

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 lapisan. Bagian dari pembelajaran mesin yang dikenal sebagai deep learning adalah algoritma yang berusaha belajar dalam berbagai tingkat, dan algoritma ini meniru cara kerja struktur dan fungsi otak (Rochmawati dkk., 2021).

Pada dasarnya, *deep learning* adalah jaringan saraf tiruan yang memiliki kemampuan untuk belajar dan beradaptasi dalam jumlah data yang besar untuk menyelesaikan masalah yang kompleks (Alfarizi dkk., 2023). Teknologi ini menjadi semakin populer karena kemampuannya untuk menganalisis kumpulan data yang besar, membuat prediksi, dan memberikan wawasan yang sebelumnya tidak mungkin diperoleh (Sharifani & Amini, 2023).

Dengan peningkatan produksi data, kekuatan pemrosesan yang lebih baik, dan kemajuan kecerdasan buatan (AI) sebagai sebuah layanan, algoritma *deep learning* mendapatkan manfaat. *Deep learning* memungkinkan mesin memecahkan masalah yang rumit bahkan dengan pengumpulan data yang sangat bervariasi, terhubung, dan terstruktur. Semakin banyak yang dipelajari algoritma pembelajaran mendalam yang lebih baik. Tujuan utama dari penelitian ini adalah untuk menarik perhatian pada komponen DL yang paling penting sehingga para peneliti dan mahasiswa dapat memperoleh pemahaman menyeluruh tentang DL dari satu tinjauan. Selain itu, hal ini memungkinkan orang untuk lebih memahami perkembangan terbaru dibidang tersebut,

yang akan meningkatkan penelitian DL. Untuk memberikan kesempatan yang lebih tepat dilapangan para peneliti(Taye, 2023).

2.2 Klasifikasi

Klasifikasi adalah proses menempatkan objek kedalam kelas-kelas yang berbeda untuk membuatnya lebih mudah dikenali serta berusaha menggeneralisasikan struktur yang sudah diketahui untuk diterapkan pada data baru.(Khultsum & Taufik, 2023).

Klasifikasi adalah proses yang menemukan sekumpulan model dan fungsi dimana yang menjelaskan dan membedakan data ke dalam kelas yang tertentu. Tujuan dari klasifikasi adalah untuk menggunakan model ini untuk menentukan kelas objek yang belum diketahui kelasnya. Ada dua proses yang termasuk dalam kategori ini yaitu proses belajar atau pelatihan, yang dimana melakukan pembuatan model dengan data pelatihan,dan proses pengujian,yang melakukan pengujian terhadap data pengujian dengan model yang telah diperoleh dari proses pelatihan(Azizah, 2023).

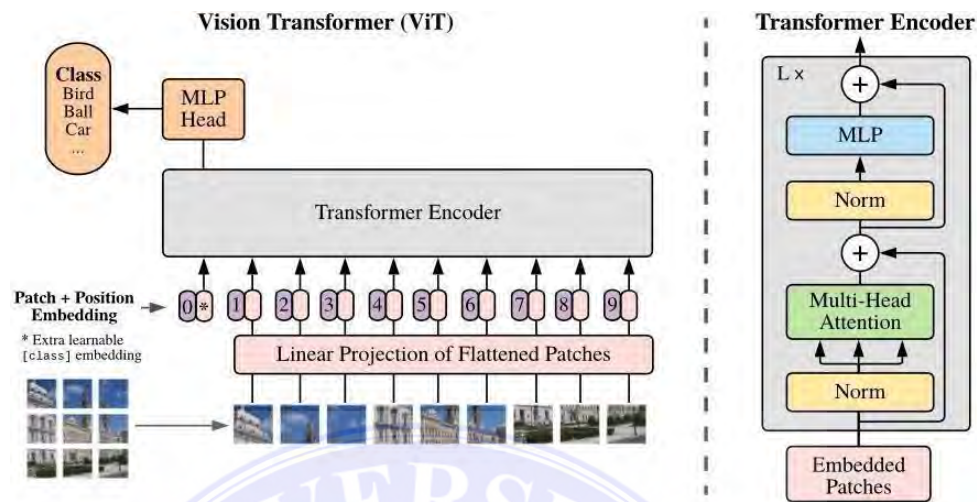
Selain itu, klasifikasi adalah metode pengelompokan yang didasarkan pada fitur yang diguakan oleh manusia dan teknologi. Klasifikasi juga sangat penting untuk pembelajaran penegtahuan sebelumnya dalam bidang penambangan data (Fajar & Mutaqin, 2024).

2.3 Vision Transformer

Arsitektur *Vision Transformer (ViT)* didasarkan pada arsitektur Transformer, yang merupakan *state-of-the-art (SOTA)* dalam pemrosesan Bahasa alami. Patch embeddings,multi-head attention,dan multi-layer perceptron adalah beberapa proses yang membentuk arsitektur *Vision transformer*(Figo dkk., 2023). Arsitektur

Transformer, yang awalnya dimaksudkan untuk pemrosesan Bahasa alami, telah berkembang untuk berbagai aplikasi. Misalnya, *Vision Transformer (ViT)* telah diciptakan untuk menyusuaikannya dengan visi computer. Sebuah gambar dibagi menjadi serangkaian dalam *ViT. Patch*, yang kemudian ditanam secara linier dan digunakan sebagai token input untuk tumpukan lapisan transformer. *Transformer encoder* disebut secara kolektif sebagai tumpukan lapisan transformer. *ViT* telah mencapai kinerja tercanggih pada berbagai tolak ukur klasifikasi gambar, menunjukkan efektivitas arsitektur Transformer di banyak domain.

Model klasifikasi gambar yang menggunakan arsitektur transformer disebut *Vision Transformer (ViT)*. Menurut penelitian yang dilakukan oleh Dosovitskiy dkk. (2020), langkah pertama yang dilakukan *ViT* adalah membagi citra input menjadi potongan-potongan kecil atau *patch* dengan ukuran yang sama. Ini terjadi untuk citra input yang di mana tinggi, lebar, dan jumlah channel, di mana ukuran patch adalah tinggi atau lebar. Setelah itu, ratakan patch menjadi. Mengubah matriks menjadi vector dikenal sebagai flatten (Pangestu dkk., 2024).



Gambar 2.1 Arsitektur *Vision Transformer*

Sumber: (Pangestu dkk., 2024)

Dengan menggunakan model *deep learning ViT*, gambar dibagi menjadi beberapa bagian yang masing-masing memiliki ukuran yang sama, dan nilai piksel dari setiap bagian akan dijadikan vector datar. Selanjutnya vector-vector tersebut dimasukkan ke dalam arsitektur transformator. Arsitektur transformator memungkinkan model untuk menampilkan kinerja klasifikasi yang lebih baik dengan mempelajari hubungan antara patch yang berbeda dalam gambar melalui mekanisme perhatiannya (Alakus & Baykara, 2024).

Arsitektur model Vision Transformer digunakan untuk mengekstrak pola gambar untuk tugas klasifikasi. Lapisan model ini digunakan untuk memperhitungkan interaksi antara berbagai bagian gambar yang diinput, sehingga model tersebut menghitung nilai antara semua pasangan penyematan token input.

2.4 Penyakit Jagung

Jagung adalah salah satu tanaman pangan terbesar ketiga di dunia, termasuk di Indonesia, setelah beras dan gandum. Jagung berasal dari *mexico* yang telah dibudidayakan selama ribuan tahun dimana tanaman jagung telah menjadi dasar kebudayaan *Aztec* dan maya. Saat ini, jagung digunakan sebagai pakan ternak dan diolah menjadi makanan ringan, tepung, dan minyak rendah kolestrol.

Ada beberapa penyakit pada tanaman jagung yang menjadi tanda perubahan pada daun ialah karat atau common rust yang disebabkan oleh jamur *Puccinia sorghi Schw* dan *Puccinia polypore Underw*, dan hawar (*blight*) yang disebabkan oleh *Helminthosporium turcicum*. Penyakit tersebut dapat menyebabkan hasil panen mengalami penurunan, jika tidak ditangani segera maka para petani mengalami kerugian (Sarah Siti, 2023). Gejala pada karat daun ini berupa Uredinia, bercak kecil berbentuk bulat sampai oval yang ditemukan dipermukaan daun jagung bagian atas dan bawah. Gejala awal terinfeksi hawar daun adalah bercak kecil berbentuk oval yang semakin memanjang dan berkembang menjadi nekrotik yang disebut hawar. Bercak ini muncul didaun terbawah dan kemudian berkembang menuju daun atas, serangan penyakit hawar daun dapat menyebabkan tanaman jagung mati atau mengering.

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).

Upaya yang dilakukan untuk mengatasi penyakit yang mengganggu pada tanaman jagung terus dilakukan, tetapi faktor lingkungan seperti cuaca, suhu, dan kelembapan dapat menyebabkan penyakit berkembang biak. Oleh karena itu, penggunaan teknologi seperti kecerdasan buatan diperlukan untuk mengatasi penyakit pada tanaman jagung(Widianto dkk., 2023).

2.5 Penelitian Terdahulu

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

Tabel 2.1 Penelitian Terdahulu

No	Penulis dan Tahun	Judul Penelitian	Hasil Penelitian
1	(Boyuan Wang dkk.,2022)	Model klasifikasi Spesies jamur otomatis untuk foodborne pencegahan penyakit berdasarkan Vision Transformer.	Penelitian tersebut membandingkan metode CNN dengan metode vision Transformer dengan memvisualisasikan output dimensi dari model ViT-L/32 untuk mencapai interpretasi dengan

			menunjukkan bahwa ViT-L/32 mencapai akurasi 95,97%.
2	(Zheng Jiang and Liang Chen dkk.,2022)	Penggabungan patch tingkat multisemantik untuk Diagnosis Pneumonia.	Penelitian tersebut mendapatkan hasil akurasi dan F1 score dari masing-masing baseline adalah 0,911859 dan 0,903365.
3	(Yuping Zheng,dan Wei Wei Jiang dkk.,2022)	Evaluasi Transformer Penglihatan untuk klasifikasi Rambu lalu lintas.	Dalam Penelitian ini, bertujuan untuk menyelidiki apakah keberhasilan vision Transformer dapat direplikasi dalam area pengenalan rambu lalu lintas.bersdasarkan sumber data yang ada,

4	(Yi Yang dan Xin Li dkk.,2024.)	Klasifikasi Bintang dengan Vision Transformer dan Gambar Fotometri SDSS.	Melalui analisis mendalam, penelitian ini menunjukkan performa klasifikasi tipe spektral bintang,dengan F1 skor mencapai 0,709.
---	---------------------------------	--	---

Selain mengambil inspirasi dari penelitian sebelumnya tentang algoritma transformasi visi, penulis juga mengambil topik penelitian yang akan dipelajari untuk memudahkan penulisan dan membandingkan penelitian dengan algoritma yang berbeda. Berikut ini adalah penelitian terkait dapat dilihat pada Tabel 2.2

Tabel 2.2 Penelitian Terkait

No	Penulis	Metode	Data	hasil
1	(Pamungkas dkk.,2023)	CNN dengan EfficientNet-B0 dan Resnet 50.	Jumlah Data: 4188 Kategori: Hawar, Karat Biasa, Bercak	Hasil akurasi yang didapatkan pada model uji scenario 1 yaitu EfficientNet-B0 adalah 94%, dan

			daun abu-abu, dan Sehat	untuk scenario 2 yaitu ResNet-50 didapatkan akurasi 93%.
2	(Mochammad Faisal Nur Sayyid dkk.,)	Klasifikasi Penyakit Daun Jagung Menggunakan Metode CNN Dengan Image Processing HE Dan CLAHE	Jumlah Data: 800 Kategori: Cercospora Leaf Spot, Common Rust, Northern Leaf Blight, Healthy	Dari pengujian klasifikasi penyakit daun jagung menggunakan metode CNN dengan image processing HE dan CLAHE memperoleh nilai akurasi tertinggi sebesar 95% dari 160 data yang diuji dengan menggunakan image processing CLAHE. Sedangkan nilai

				akurasi yang didapatkan dengan image processing HE sebesar 91%.
3	(Prasetyo dkk., 2022)	Klasifikasi Penyakit Tanaman Jagung Berdasarkan Daun Menggunakan Arsitektur Residual Network-9.	Jumlah Data: 9145 Kategori: Karat Biasa, Bercak daun abu-abu, Hawar Daun, dan Sehat	Dalam penelitian ini memiliki Tingkat akurasi sebesar 99%.
4	(Hatem dkk., 2023)	Mengidentifikasi Penyakit Daun Jagung Dengan Transfer Learning Secara Ekstensif: Sebuah Studi Perbandingan.	Kategori: Hawar, Karat Biasa, Bercak daun abu-abu, dan Daun Sehat.	Dalam penelitian ini menggunakan 4 model CNN yang telah dilatih yaitu: GoogleNet, AlexNet, ResNet50, dan VGG16, dengan kinerja model transfer learning, keempat model

				CNN tersebut memiliki hasil akurasi masing-masing sebesar 98.57%, 98.81%, 99.05%, dan 99.36%.
5	(Kusuma dkk., 2022)	Klasifikasi Penyakit Daun Pada Tanaman Jagung Menggunakan Algoritma Support Vector Machine, K-Nearest Neighbors dan Multilayer Perceptron	Jumlah Data: 8200 Kategori: Bercak daun abu-abu, Karat Biasa, Hawar Daun, dan Daun Sehat	Pada penelitian ini melakukan komparasi algoritma klasifikasi yaitu Support Vector Machine, K-Nearest Neighbors dan Multilayer Perceptron untuk menemukan algoritma terbaik dalam klasifikasi penyakit daun pada tanaman jagung

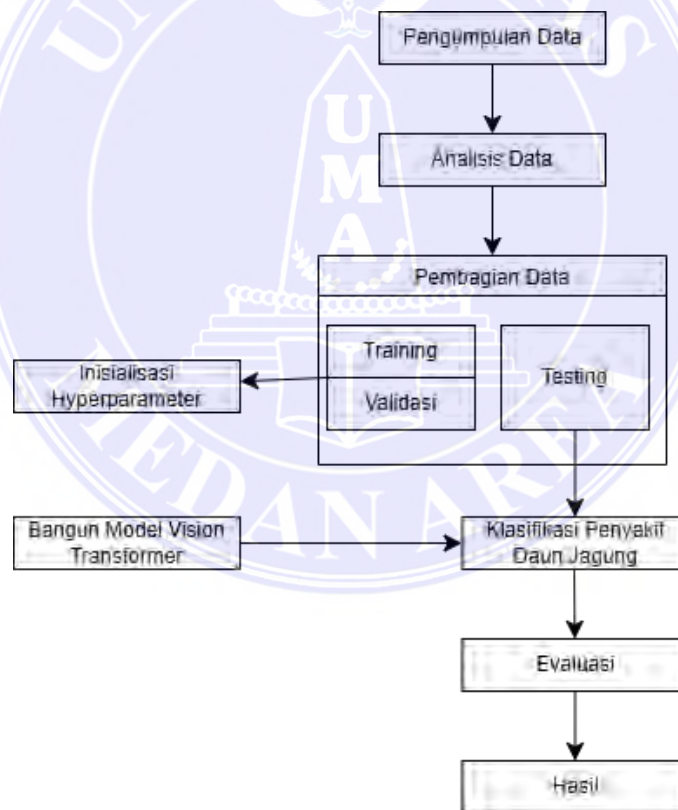
				<p>dengan menggunakan model deep learning VGG-16 yang digunakan sebagai ekstraksi fitur citra. Hasil yang didapatkan menunjukkan bahwa algoritma Multilayer Perceptron menghasilkan nilai terbaik dengan <i>accuracy, precision</i> dan <i>recall</i> masing-masing sebesar 97.4%</p>
--	--	--	--	---

BAB III

METODOLOGI PENELITIAN

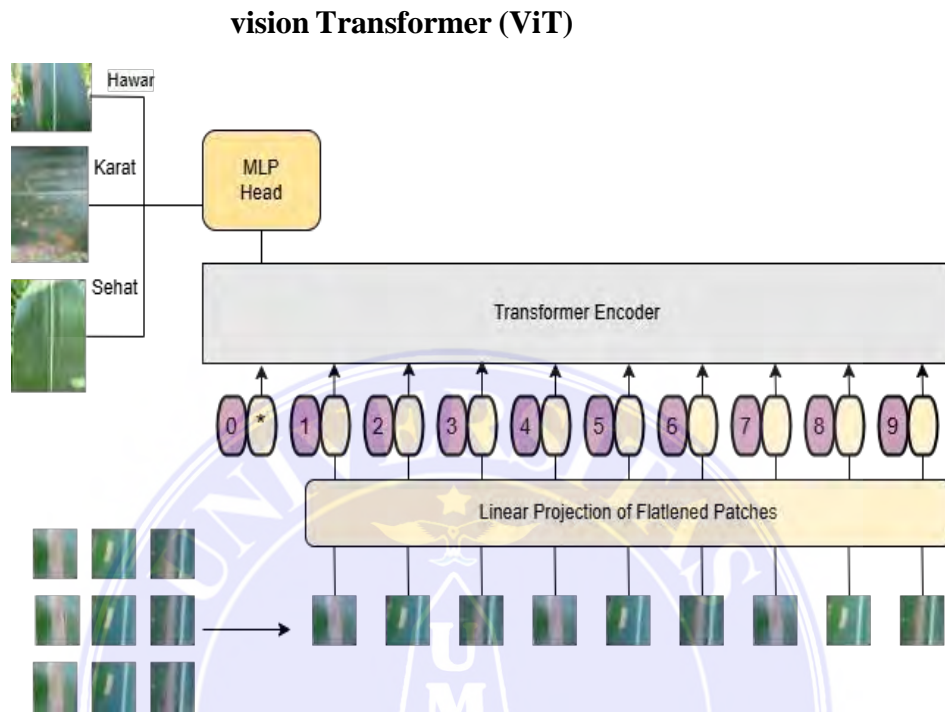
3.1 Metode Penelitian

Dalam analisis arsitektur *vision Transformer* untuk klasifikasi jenis penyakit pada tanaman jagung akan dibuat beberapa tahapan dalam bentuk diagram alur guna mempermudah dalam penyusunan penelitian ini dapat dilihat dari gambar 3.1 sebagai kerangka penelitian 3.2 sebagai Model *Vision Transformer*.



Gambar 3.1 Kerangka Penelitian

3.2 Model ViT



Gambar 3.2 Arsitektur Model *Vision Transformer*

Dengan menggunakan model *deep learning ViT*, gambar dibagi menjadi 3 bagian yaitu hawar, karat, dan sehat. Dimana masing-masing memiliki ukuran yang sama, dan nilai piksel dari setiap bagian akan dijadikan vektor datar. Selanjutnya vektor-vektor tersebut dimasukkan ke dalam Arsitektur Transformer. Arsitektur model Vision Transformer digunakan untuk mengekstrak pola gambar untuk tugas klasifikasi. Lapisan model ini digunakan untuk memperhitungkan interaksi antara berbagai bagian gambar yang diinput, sehingga model tersebut menghitung nilai antara semua pasangan penyematan token input.

3.3 Teknik Pengumpulan Data

Dalam penelitian ini data masukan diperoleh dari citra yang diambil dari ladang para petani pada tanaman jagung yang dilakukan di Desa Hutagur-gur 1. Pada pengambilan data ini dilakukan dengan menggunakan smartphone dan kamera *canon ixus 175* berformat jpeg dan JPG, kemudian dalam pengujian ini dilakukan kelas diantaranya yaitu kelas positif dan negatif, dan menggunakan studi literature untuk mendukung pada penelitian ini. Data tersebut dikumpulkan sebanyak 3000 gambar daun jagung dan data tersebut dibagi menjadi 3 (tiga) kategori yang masing-masing kategori terdapat 1000 gambar per kategori.

3.4 Analisis Data

Dalam penelitian ini, data yang digunakan adalah gambar penyakit pada daun jagung sebanyak 3000 data, data tersebut dibagi menjadi 3 bagian yaitu data Training, data Testing dan validasi, lalu data tersebut dilakukan augmentasi seperti propping dan resize, crop sampel dilakukan agar fokus ke daun sebagai objek utama, lalu dilakukan resize supaya pada proses training tidak begitu berat atau lama, sampel penyakit pada tanaman jagung terdiri dari 80% data training dan 10% data testing dan 10% data validasi.

Tabel 3.1 *Split Dataset*

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

3.5 *Hyperparameter*

Hyperparameter sangat penting untuk mengatur dan mengontrol metode pemodelan karena mereka banyak sekali melibatkan parameter untuk melakukan proses training data.

Selama proses training, ada beberapa parameter yang digunakan dalam sebuah penelitian ini adalah *Optimizer* dan fungsi aktivasi. *Optimizer* mampu meminimalkan loss function, karena *optimizer* bertugas untuk mengatur dan mengubah atribut model. Salah satu *optimizer* yang digunakan adalah *Adam Optimizer*. Selain *optimizer*, *hyperparameter* yang digunakan adalah fungsi aktivasi, berikut adalah fungsi aktivasi yang akan digunakan pada penelitian ini:

1. GELU

GELU (Gaussian Error Linear Unit) adalah fungsi aktivasi yang digunakan dalam jaringan saraf tiruan, terutama dalam bidang deep learning. *GELU*

menggabungkan karakteristik fungsi aktivasi linear dan non-linear dan telah terbukti berfungsi dengan baik pada sejumlah model jaringan saraf. Menurut *LeCun*, fungsi aktivasi adalah sebuah fungsi non linear yang digunakan untuk melakukan training terhadap neural network, dengan mendefinisikan output dari neuron dari satu set input yang diberikan. Cara kerja fungsi aktivasi di dalam neural network adalah; input diberikan kepada network dari input layer. Di dalam neuron, jumlah bobot dari input-input dikalkulasi, dan biasa ditambahkan kepada jumlah tersebut. Jumlah ini kemudian melalui fungsi aktivasi. Persamaan *GELU* dapat dilihat sebagai berikut:

$$GELU(x) \approx 0.5x(1 + \tanh(\pi 2(x + 0.044715x^3))) \dots\dots\dots 1$$

2. ReLU

ReLU (Rectified Linear Unit) Salah satu fungsi aktivasi yang paling banyak digunakan dalam jaringan saraf tiruan (neural networks), terutama dalam jaringan deep learning, adalah ReLU, atau Rectified Linear Unit. Meskipun sederhana, fungsi ini berhasil mengatasi masalah yang dihadapi oleh fungsi aktivasi lain seperti sigmoid atau tanh, seperti masalah vanishing gradient. Fungsi aktivasi ReLU memiliki kinerja yang lebih baik dibandingkan fungsi aktivasi Tanh untuk model penelitian ini, terlihat dari rata-rata akurasi dan presisinya yang lebih tinggi.

Definisi Fungsi ReLU

Fungsi ReLU didefinisikan secara matematis sebagai:

$$ReLU(x) = \max(0, x) \dots\dots\dots 2$$

Dimana:

X adalah input.

Artinya, jika input x bernilai lebih besar dari 0, outputnya sama dengan x ; tetapi jika input x kurang dari atau sama dengan 0, outputnya akan menjadi 0. Sifat-sifat Fungsi ReLU Linear untuk Input Positif: Saat $x > 0$, fungsi ReLU bersifat linear dan outputnya sama dengan inputnya.

Nol untuk Input Negatif: Saat $x \leq 0$, fungsi ReLU memberikan output 0, yang membuatnya sangat efisien dalam komputasi.

Cara Kerja ReLU

Input Neuron: Setiap neuron di jaringan saraf menerima input yang merupakan kombinasi linear dari input sebelumnya, ditambah dengan bias:

$$Z = W_1X_1 + W_2X_2 + \dots + W_nX_n + b \dots\dots\dots 3$$

Dimana:

W_1, W_2, \dots, W_n adalah bobot.

X_1, X_2, \dots, X_n adalah input.

b adalah bias

z adalah hasil kombinasi linear (nilai input neuron sebelum fungsi aktivasi

diterapkan). Aplikasi Fungsi ReLU: Setelah mendapatkan nilai input z , fungsi

ReLU diterapkan:

$$ReLU(z) = \max(0, z) \dots\dots\dots 4$$

Aktivasi atau Non-Aktivasi Neuron: Jika $z > 0$, neuron diaktifkan (outputnya sama dengan z).

Jika $z \leq 0$, neuron tidak diaktifkan (outputnya adalah 0).

Forward Propagation yang merupakan dari output fungsi ReLU ini kemudian diteruskan ke lapisan berikutnya dalam jaringan saraf. Proses ini diulang untuk setiap neuron dalam jaringan sampai mencapai lapisan output.

3. ELU

ELU (*Exponential Linear Unit*) adalah salah satu fungsi aktivasi yang digunakan dalam jaringan saraf tiruan (*neural networks*), untuk menentukan apakah suatu neuron harus diaktifkan atau tidak, fungsi ini mengubah input menjadi output yang akan diteruskan ke lapisan berikutnya kedalam jaringan saraf. fungsi ELU juga adalah pilihan yang baik ketika ingin mengatasi beberapa masalah dari fungsi aktivasi lain seperti saturasi atau "dead neurons" yang sering terjadi pada fungsi ReLU.

Definisi Matematis Fungsi Elu

Secara matematis, fungsi aktivasi Elu didefinisikan sebagai berikut:

$$ELU(x) = \begin{cases} x & , \text{jika } x > 0 \\ \alpha(e^x - 1), & \text{jika } x \leq 0 \end{cases} \dots\dots\dots 5$$

Dimana:

x adalah input ke fungsi aktivasi.

α adalah konstanta yang mengontrol nilai saturasi untuk input negatif (biasanya diatur ke 1).

Fungsi dan Tujuan Utama ELU:

Mengaktifkan Neuron dalam Jaringan Saraf: Fungsi ELU memutuskan seberapa banyak sinyal yang harus diteruskan ke neuron-neuron berikutnya dalam

jaringan. Ini penting untuk pembelajaran, karena menentukan bagaimana jaringan memodifikasi bobot-bobotnya. Mengurangi *Vanishing Gradient*: Dengan mempertahankan gradien yang lebih besar untuk nilai negatif, ELU membantu mengurangi masalah *vanishing gradient* yang sering terjadi dalam jaringan saraf yang dalam. Ini memungkinkan jaringan untuk terus belajar meskipun banyak lapisan yang terlibat. Normalisasi Aktivasi: ELU menghasilkan keluaran dengan nilai rata-rata mendekati nol, yang membantu menjaga distribusi data lebih teratur di seluruh lapisan jaringan saraf. Ini bisa meningkatkan efisiensi pelatihan dengan mempercepat konvergensi. Meningkatkan Stabilitas Pelatihan: Fungsi ELU mengurangi ketidakstabilan yang disebabkan oleh nilai input yang besar, karena outputnya menjadi jenuh secara eksponensial untuk nilai input yang sangat negatif. Hal ini mencegah pembaruan bobot yang ekstrem selama pelatihan.

4. PARAMETRIK ReLU (PReLU)

Parametric ReLU (PReLU) adalah fungsi aktivasi yang digunakan dalam jaringan saraf tiruan (*neural networks*) yang merupakan varian dari fungsi ReLU (*Rectified Linear Unit*). PReLU memperkenalkan parameter yang dapat dipelajari, yang memungkinkan fungsi untuk menyesuaikan sendiri nilai *slope* (kemiringan) pada bagian negatifnya selama proses pelatihan. Secara sederhana, PReLU mengatasi masalah "*dead neurons*" yang sering terjadi pada ReLU standar. Dalam ReLU, output untuk semua input negatif adalah nol, sehingga gradien menjadi nol, dan neuron tersebut berhenti belajar. PReLU memperkenalkan parameter α (*alpha*) yang menentukan seberapa besar output untuk nilai input negatif. Nilai ini tidak tetap tetapi dipelajari selama pelatihan, memberikan jaringan fleksibilitas yang

besar. Manfaat ReLU terletak pada kesederhanaan, efisiensi komputasi, kemampuannya untuk mengatasi masalah *vanishing gradient*, dan kemampuannya untuk menghasilkan aktivasi yang jarang. Ini menjadikannya pilihan yang sangat populer dalam banyak arsitektur jaringan saraf modern, terutama untuk tugas-tugas yang melibatkan data dalam jumlah besar dan jaringan yang dalam. Cara kerja PReLU melibatkan penggunaan parameter α /alpha yang dapat dipelajari untuk mengontrol nilai output untuk input negatif, yang membuat jaringan saraf lebih fleksibel dan adaptif. Dengan pembaruan parameter ini selama pelatihan, PReLU dapat meningkatkan stabilitas, mengurangi masalah "dead neurons", dan meningkatkan kinerja model pembelajaran mesin.

Definisi Matematis

Fungsi PReLU didefinisikan sebagai:

$$\text{PReLU}(x) = \begin{cases} x & , \text{jika } x > 0 \\ \alpha x & , \text{jika } x \leq 0 \end{cases} \quad 6$$

Dimana:

x adalah input ke fungsi aktivasi

α adalah parameter yang dapat dipelajari dan biasanya diinisialisasi dengan nilai kecil, seperti 0,01

5. LEAKY ReLU

Leaky ReLU (*Leaky Rectified Linear Unit*) adalah varian dari fungsi aktivasi ReLU (Rectified Linear Unit) yang dirancang untuk mengatasi masalah "dead neurons" pada jaringan saraf tiruan. Pada ReLU standar, input negatif

menghasilkan output nol, yang dapat menyebabkan beberapa neuron menjadi tidak aktif secara permanen dan tidak berkontribusi lagi selama pelatihan. Leaky ReLU mengatasi masalah ini dengan memperkenalkan *slope* (kemiringan) kecil untuk input negatif, sehingga neuron tetap menerima pembaruan gradien meskipun inputnya negatif. Fungsi ini mempertahankan gradien dan memungkinkan pembelajaran yang lebih efisien, mengurangi risiko "*dead neurons*," mempercepat konvergensi, dan meningkatkan kinerja model dalam berbagai tugas pembelajaran mesin. Leaky ReLU (*Leaky Rectified Linear Unit*) bekerja dengan cara memberikan kemiringan (*slope*) kecil untuk nilai input negatif, dan memastikan bahwa neuron tetap aktif meskipun menerima input negatif, yang membantu dalam proses pembelajaran jaringan saraf.

3.6 Evaluasi

Confusion Matrix adalah parameter yang digunakan untuk mengevaluasi kualitas dan kerja dari model yang ada. Untuk mengetahui kualitas dari sistem kita dapat melihat nilai *matrix confusion* yang dirancang untuk menganalisis tingkat ketepatan, ketepatan, recall, dan skor F1. *True positive* (TP) didefinisikan sebagai data positif yang diprediksi positif, *true negative* (TN) didefinisikan sebagai data negative yang diprediksi negative, dan *false positif* (FP) didefinisikan sebagai data negative yang diprediksi negatif. Adapun kelas *confusion matrix* dapat dilihat pada table dibawah ini.

Tabel 3.2 *Confussion Matrix*

<i>Confussion Matrix</i>		Kelas Sebenarnya	
		Positif	Negatif
Prediksi	Positif	<i>True Positif</i>	<i>False Negatif</i>
	Negatif	<i>False Negatif</i>	<i>True Negatif</i>

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

1. *Accuracy*

Accuracy merupakan metrik untuk model klasifikasi yang menentukan kemiripan prediksi yang benar dengan data gambar secara keseluruhan.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

2. *Precision*

Precision ialah rasio sampel yang diprediksi benar positif dibandingkan dengan seluruh hasil yang diprediksi positif. *Precision* digunakan untuk menunjukkan jumlah data kategori yang diklasifikasikan dengan benar, dibagi dengan jumlah total data yang diklasifikasikan positif.

$$Precision = \frac{TP}{TP+FP}$$

3. *Recall*

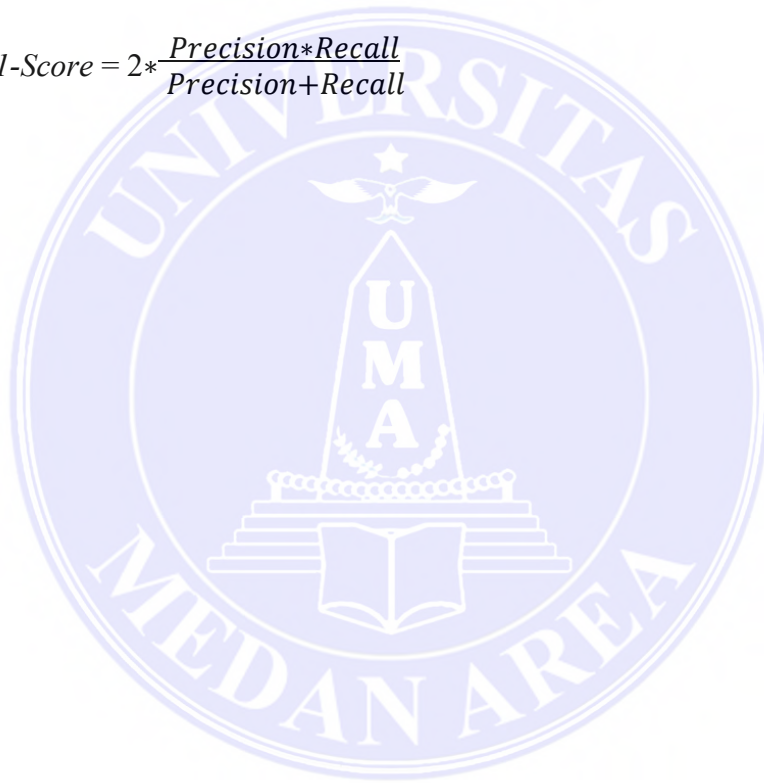
Recall merupakan rasio sampel prediksi benar positif terhadap keseluruhan data. *Recall* dapat dihitung dalam persamaan berikut:

$$Recall = \frac{TP}{TP+FN}$$

4. *F1-score*

F1-score adalah ukuran kinerja yang menghitung skor dengan mempertimbangkan *presisi* dan *recall*. *F1-score* sendiri memiliki nilai 1 sebagai nilai tertinggi dan nilai 0 sebagai nilai terendah. Nilai *F1-score* dapat dihitung dengan persamaan berikut.

$$F1-Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$



BAB V

KESIMPULAN dan SARAN

5.1. Kesimpulan

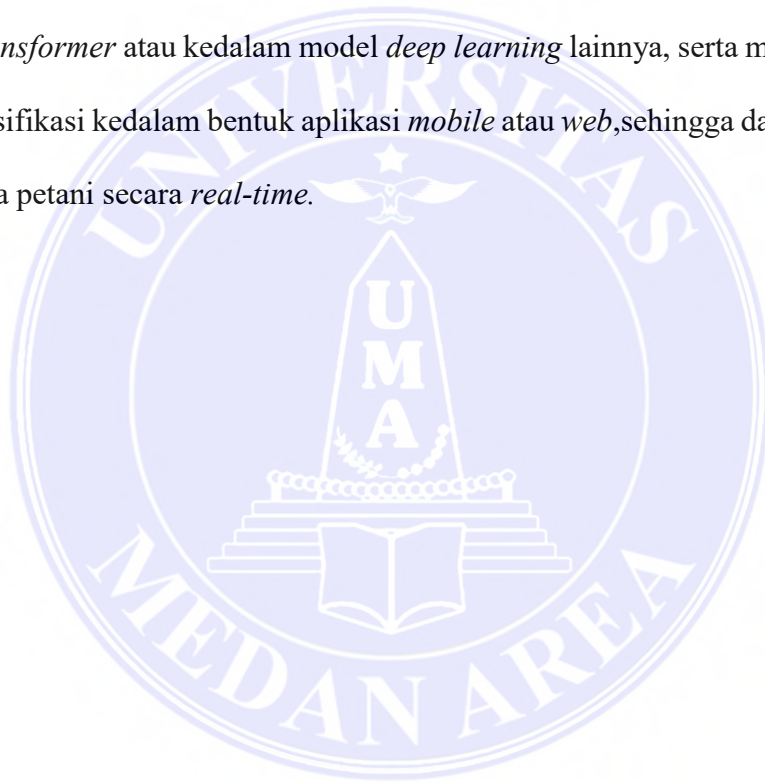
Adapun Kesimpulan yang ada dapat diambil dari hasil penerapan *Vision Transformer* untuk klasifikasi penyakit tanaman jagung melalui citra daun adalah sebagai berikut:

1. Penerapan algoritma *Vision Transformer* (ViT) untuk klasifikasi penyakit pada tanaman jagung melalui citra daun 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, pada Penerapan algoritma *Vision Transformer* (ViT) untuk klasifikasi penyakit pada tanaman jagung melalui citra daun memberikan hasil yang baik. Hal ini terlihat dari nilai akurasi yang diperoleh pada masing-masing pengujian menggunakan fungsi aktivasi yang berbeda, yaitu: GELU (93,00%), ELU (92,00%), RELU (93,67%), PRELU (94,33%), dan Leaky RELU (93,67%).

5.2. Saran

Saran yang dapat penulis berikan untuk pengembangan selanjutnya adalah sebagai berikut:

1. Perlu dilakukan pengumpulan dataset yang lebih besar dan beragam untuk meningkatkan akurasi model.
2. Melakukan klasifikasi dengan menyisipkan model *few-shot* kedalam model *Vision Transformer* atau kedalam model *deep learning* lainnya, serta mengembangkan sistem klasifikasi kedalam bentuk aplikasi *mobile* atau *web*, sehingga dapat dimanfaatkan oleh para petani secara *real-time*.



DAFTAR PUSTAKA

- Alakuş, T. B., & Baykara, M. (2024). Classification and Determination of Severity of Corneal Ulcer with Vision Transformer Based on the Analysis of Public Image Dataset of Fluorescein-Stained Corneas. *Diagnostics*, 14(8). <https://doi.org/10.3390/diagnostics14080786>
- 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.
- 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>
- Cara, M. C., Gleyzer, S., Dahale, G. R., Dong, Z. T., Forestano, R. T., Keadilan, D., Kong, K., Magorsch, T., Matchev, K. T., & Unlu, E. B. *aksioma Transformator Visi Kuantum untuk Klasifikasi Quark – Gluon*. 1–14.
- Fajar, A. K., & Mutaqin, M. Z. (2024). *KLASIFIKASI KANKER PAYUDARA MENGGUNAKAN ALGORITMA NEURAL NETWORK DAN RANDOM FOREST*. 7, 74–80.
- Figo, J. A., Yudistira, N., & Widodo, A. W. (2023). Deteksi Covid-19 dari Citra X-ray menggunakan Vision Transformer. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 7(3), 1116–1125. <https://j-ptiik.ub.ac.id>
- Firgia, L., & Thomas, S. (2023). Deteksi Jenis Penyakit Dan Hama Pada Tanaman Jagung Menggunakan Arsitektur Spatial Pyramid Pooling Pada YOLOv5s. *Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)*, 8(2), 452–459. <https://tunasbangsa.ac.id/ejurnal/index.php/jurasik>
- Fraiwan, M., Faouri, E., Khasawneh, N., Komputer, D. T., Sains, U., Departemen, Y., Lunak, R. P., & Sains, U. (2022). *Klasifikasi Penyakit Jagung dari Gambar Daun*

Menggunakan Deep Transfer Learning.

- 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>
- Pangestu, A., Purnama, B., & Risnandar, R. (2024). Vision Transformer untuk Klasifikasi Kematangan Pisang. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(1), 75–84. <https://doi.org/10.25126/jtiik.20241117389>
- Prakosa, A. B., Hendry, & Tanone, R. (2023). Implementasi Model Deep Learning Convolutional Neural Network (CNN) Pada Citra Penyakit Daun Jagung Untuk Klasifikasi Penyakit Tanaman. *Jurnal Pendidikan Teknologi Informasi (JUKANTI)*, 6(1), 107–116.
- Rachmawanto, E. H., & Hadi, H. P. (2021). Optimasi Ekstraksi Fitur Pada Knn Dalam Klasifikasi Penyakit Daun Jagung. *Dinamik*, 26(2), 58–67. <https://doi.org/10.35315/dinamik.v26i2.8673>
- Rochmawati, N., Hidayati, H. B., Yamasari, Y., Tjahyaningtijas, 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>
- Sarah Siti, G. (2023). Identifikasi Penyakit Tanaman Jagung Berdasarkan Citra Daun Tinjauan Literatur Sistematis (Slr). *Semaster*, 278–289(Prosiding-Seminar Nasional Teknologi Informasi & Ilmu Komputer (SEMASTER)), 1–12.

- 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>
- Soleymani, M., Bonyani, M., & Mahami, H. (2021). *Construction Management Construction Material Classification on Imbalanced Datasets Using Vision Transformer Architecture (Vit)*. 0–3.
- 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>
- 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), 251–25

LAMPIRAN

1. Source Code

```
# !pip install tensorflow_addons
!pip install --upgrade tensorflow-addons
!pip install --upgrade tensorflow==2.14.0
!pip install --upgrade keras==2.14.0

import numpy as np
import pandas as pd
import tensorflow as tf
from tensorflow.keras import layers as L
#import tensorflow.keras.layers as L
import tensorflow_addons as tfa
import glob, random, os, warnings
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix, classification_report
import seaborn as sns
from sklearn.model_selection import train_test_split
import shutil
from pathlib import Path

print('TensorFlow Version ' + tf.version_)

def seed_everything(seed = 0):
    random.seed(seed)
    np.random.seed(seed)
    tf.random.set_seed(seed)
    os.environ['PYTHONHASHSEED'] = str(seed)
    os.environ['TF_DETERMINISTIC_OPS'] = '1'
```

```

seed_everything()

import warnings
warnings.filterwarnings("ignore")
from google.colab import drive
drive.mount('/content/drive')
image_size = 224
batch_size = 16
n_classes = 3
EPOCHS = 30

# 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)

classes = {1 : "daun_hawar", 2 : "daun_karat", 3 : "daun_sehat"}

# 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)

def data_augment(image):

```

```

p_spatial = tf.random.uniform([], 0, 1.0, dtype = tf.float32)
p_rotate = tf.random.uniform([], 0, 1.0, dtype = tf.float32)
p_pixel_1 = tf.random.uniform([], 0, 1.0, dtype = tf.float32)
p_pixel_2 = tf.random.uniform([], 0, 1.0, dtype = tf.float32)
p_pixel_3 = tf.random.uniform([], 0, 1.0, dtype = tf.float32)

# Pixel-level transforms
with tf.device('/CPU:0'): # Force computation on CPU
    if p_pixel_2 >= .4:
        image = tf.image.random_contrast(image, lower = .8, upper = 1.2)

# Flips
image = tf.image.random_flip_left_right(image)
image = tf.image.random_flip_up_down(image)

if p_spatial > .75:
    image = tf.image.transpose(image)

# Rotates
if p_rotate > .75:
    image = tf.image.rot90(image, k = 3) # rotate 270°
elif p_rotate > .5:
    image = tf.image.rot90(image, k = 2) # rotate 180°
elif p_rotate > .25:
    image = tf.image.rot90(image, k = 1) # rotate 90°

# Pixel-level transforms
with tf.device('/CPU:0'):
    if p_pixel_1 >= .4:
        image = tf.image.random_saturation(image, lower = .7, upper = 1.3)
    if p_pixel_2 >= .4:
        image = tf.image.random_contrast(image, lower = .8, upper = 1.2)

```

UNIVERSITAS MEDAN AREA

```

        if p_pixel_3 >= .4:
            image = tf.image.random_brightness(image, max_delta = .1)

    return image

datagen = tf.keras.preprocessing.image.ImageDataGenerator(rescale=1./255,
                                                            samplewise_center=True,
                                                            samplewise_std_normalization=True,
                                                            preprocessing_function=data_augment)

train_images = datagen.flow_from_dataframe(
    dataframe=train_df,
    x_col='Filepath',
    y_col='Label',
    target_size=(224, 224),
    class_mode='categorical',
    batch_size=16,
    shuffle=True,
    seed=42
)

val_images = datagen.flow_from_dataframe(
    dataframe=val_df,
    x_col='Filepath',
    y_col='Label',
    target_size=(224, 224),
    class_mode='categorical',
    batch_size=16,
    shuffle=False
)

test_images = datagen.flow_from_dataframe(
    dataframe=test_df,

```



```
x_col='Filepath',
y_col='Label',
target_size=(224, 224),
class_mode='categorical',
batch_size=16,
shuffle=False
)
import tensorflow as tf
import matplotlib.pyplot as plt
import warnings

warnings.filterwarnings("ignore")

with tf.device('/CPU:0'):
    images = [train_images[0][0][i] for i in range(16)]

fig, axes = plt.subplots(3, 5, figsize=(10, 10))

for i, ax in enumerate(axes.flatten()):
    if i < len(images):
        ax.imshow(images[i])
        ax.axis('off')

plt.show()

!pip install vit_keras
from vit_keras import vit

vit_model = vit.vit_b16(
    image_size = image_size,
    activation = 'sigmoid',
    pretrained = True,
```

UNIVERSITAS MEDAN AREA

```

        include_top = False,
        pretrained_top = False,
        classes = 3)
# Define Patches class with color transformation
class Patches(tf.keras.layers.Layer):
    def __init__(self, patch_size):
        super(Patches, self).__init__()
        self.patch_size = patch_size

    def call(self, images):
        batch_size = tf.shape(images)[0]
        patches = tf.image.extract_patches(
            images=images,
            sizes=[1, self.patch_size, self.patch_size, 1],
            strides=[1, self.patch_size, self.patch_size, 1],
            rates=[1, 1, 1, 1],
            padding='VALID',
        )
        # Calculate correct number of patches per image
        num_patches = (image_size // patch_size) ** 2
        patches = tf.reshape(patches, [batch_size, num_patches, -1]) # Reshape to
        (batch_size, num_patches, elements_per_patch)

        # Apply color transformation to patches
        # Example: Inverting colors
        patches = 1.0 - patches

    return patches

image_size = 224
patch_size = 7 # Size of the patches to be extracted from the input images

```

UNIVERSITAS MEDAN AREA # Ambil gambar dari dataset

© 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

```

x = train_images.next()
image = x[0][0]

plt.figure(figsize=(4, 4))
plt.imshow(image)
plt.axis('off')
plt.show()

# Ubah ukuran gambar
resized_image = tf.image.resize(
    tf.convert_to_tensor([image]), size=(image_size, image_size)
)

patch_layer = Patches(patch_size)
patches = patch_layer(resized_image)
print(f'Image size: {image_size} X {image_size}')
print(f'Patch size: {patch_size} X {patch_size}')
print(f'Patches per image: {patches.shape[1]}')
print(f'Elements per patch: {patches.shape[1]}')

# for i in range(5):
#     print(f'Patch {i} values: {patches[0, 1].numpy()[:5]}')

# Menampilkan patch gambar dengan warna asli
n = int(np.sqrt(patches.shape[1]))
plt.figure(figsize=(10, 10))

for i, patch in enumerate(patches[0]):
    ax = plt.subplot(n, n, i + 1)
    patch_img = tf.reshape(patch, (patch_size, patch_size, 3))
    plt.imshow(patch_img.numpy().astype("uint8"))
    plt.axis('off')

```

```

plt.show()

model = tf.keras.Sequential([
    vit_model,
    tf.keras.layers.LayerNormalization(),
    tf.keras.layers.Flatten(),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dense(128, activation = tfa.activations.gelu),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dense(64, activation = tfa.activations.gelu),
    tf.keras.layers.Dense(32, activation = tfa.activations.gelu),
    tf.keras.layers.Dense(3, 'softmax')
],
name = 'vision_transformer')

model.summary()
warnings.filterwarnings("ignore")

learning_rate = 1e-4

optimizer = tfa.optimizers.RectifiedAdam(learning_rate = learning_rate)

# The number of classes in the loss function should match the model output
model.compile(optimizer = optimizer,
              loss = tf.keras.losses.CategoricalCrossentropy(label_smoothing = 0.2,
                    from_logits=False), # from_logits should be False when using softmax activation
              metrics = ['accuracy'])

STEP_SIZE_TRAIN = train_images.n // train_images.batch_size
STEP_SIZE_VALID = val_images.n // val_images.batch_size

```

```
early_stopping_callbacks = tf.keras.callbacks.EarlyStopping(patience = 15,  
restore_best_weights = True, verbose = 1)
```

```
# Train the model
```

```
history = model.fit(  
    x=train_images,  
    steps_per_epoch=STEP_SIZE_TRAIN,  
    validation_data=val_images,  
    validation_steps=STEP_SIZE_VALID,  
    epochs=EPOCHS,  
    callbacks=[early_stopping_callbacks]  
)
```

```
# Save The Model
```

```
model.save('model.h5')
```

```
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
# Predict and evaluate
predicted_classes = np.argmax(model.predict(test_images, steps=test_images.n //
test_images.batch_size + 1), axis=1)
true_classes = test_images.classes
class_labels = list(test_images.class_indices.keys())

confusionmatrix = confusion_matrix(true_classes, predicted_classes)
plt.figure(figsize=(8, 6))
sns.heatmap(confusionmatrix, annot=True, fmt="d", cmap="Blues",
xticklabels=class_labels, yticklabels=class_labels)
plt.ylabel('True Label')
plt.xlabel('Predicted Label')

```

```
plt.show()
```

```
print(classification_report(true_classes, predicted_classes,  
target_names=class_labels, digits = 4))
```





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 : univ_medanarea@uma.ac.id

Nomor : 804/FT/01.10/III/2024

18 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 : FITTAULI LUMBAN GAOL

NIM : 208160010

Jurusan : TEKNIK INFORMATIKA

Maka dengan hormat kami mengharapkan kesediaan saudara :

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

Adapun Tugas Akhir Skripsi berjudul :

"Penerapan Vision Transformer (Vit) untuk Klasifikasi Penyakit Jagung Melalui Citra Daun"

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.



UNIVERSITAS MEDAN AREA

FAKULTAS TEKNIK

Kampus I : Jalan Kolam Nomor 1 Medan Estate/Jalan PBSI Nomor 1 ☎ (061) 7366878, 7360168, 7364348, 7366781, Fax (061) 7366998 Medan 20223
 Kampus II : Jalan Setrabudi Nomor 79 / Jalan Sei Serayu Nomor 70 A. ☎ (061) 8225602; Fax (061) 8226331 Medan 20122
 Website www.teknik.uma.ac.id E-mail univ_medanarea@uma.ac.id

Nomor : 294 /FT.6/01.10/VII/2024 2 Juli 2024
 Lamp : -
 Hal : **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	N A M A	N P M	PRODI
1	Fittauli Lumban Gaol	208160010	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 *Vision Transformer (Vit)* untuk Klasifikasi Penyakit Jagung Melalui Citra Daun

Atas perhatian dan kerja sama yang baik diucapkan terima kasih.

Dekan,

Dr. Eng. Supriatno, ST, MT

Tembusan :
 1 Ka. BPMPP
 2 Mahasiswa
 3 File



PEMERINTAH KABUPATEN TOBA
KECAMATAN SILAEN
DESA HUTAGURGUR I

Hutagurgur I, 12 Juli 2024

Nomor : 87 / 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 : 294 / 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 : Fittauli Lumban Gaol
NPM : 208160010
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 Vision Transformer (ViT) untuk Klasifikasi Penyakit Jagung Melalui Citra Daun*".

Demikian disampaikan untuk dapat dipergunakan sebagaimana mestinya.

KEPALA DESA HUTAGURGUR I

HENGKY SUBANGUN SIAGIAN

