

**KLASIFIKASI CITRA PENYAKIT PADI MENGGUNAKAN CNN
BERDASARKAN SEGMENTASI K-MEANS**



TESIS

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister
Ilmu Komputer (M.Kom)**

FAJAR SARASATI

14002387

Program Studi Ilmu Komputer (S2)

Fakultas Teknologi Informasi

Universitas Nusa Mandiri

2021

**KLASIFIKASI CITRA PENYAKIT PADI MENGGUNAKAN CNN
BERDASARKAN SEGMENTASI K-MEANS**



TESIS

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister
Ilmu Komputer (M.Kom)**

FAJAR SARASATI

14002387

Program Studi Ilmu Komputer (S2)

Fakultas Teknologi Informasi

Universitas Nusa Mandiri

2021

SURAT PERNYATAAN ORISINALITAS DAN BEBAS PLAGIARISME

Yang bertanda tangan di bawah ini:

Nama : Fajar Sarasati
NIM : 14002387
Program Studi : Ilmu Komputer
Fakultas : Teknologi Informasi
Jenjang : Strata Dua (S2)
Konsentrasi : Image Processing

Dengan ini menyatakan bahwa tesis yang telah saya buat dengan judul: “Klasifikasi Citra Penyakit Padi Menggunakan *CNN* Berdasarkan Segmentasi *K-Means*” adalah hasil karya sendiri, dan semua sumber baik yang kutip maupun yang dirujuk telah saya nyatakan dengan benar dan tesis belum pernah diterbitkan atau dipublikasikan dimanapun dan dalam bentuk apapun.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya. Apabila dikemudian hari ternyata saya memberikan keterangan palsu dan atau ada pihak lain yang mengklaim bahwa tesis yang telah saya buat adalah hasil karya milik seseorang atau badan tertentu, saya bersedia diproses baik secara pidana maupun perdata dan kelulusan saya dari Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri dicabut/dibatalkan.

Jakarta, 23 Agustus 2021

Yang menvatakan,



Fajar Sarasati

HALAMAN PERSETUJUAN DAN PENGESAHAN TESIS

Tesis ini diajukan oleh:

Nama : Fajar Sarasati
NIM : 14002387
Program Studi : Ilmu Komputer
Jenjang : Strata Dua (S2)
Konsentrasi : *Image Processing*
Judul Tesis : Klasifikasi Citra Penyakit Padi Menggunakan CNN
Berdasarkan Segmentasi *K-Means*

Telah dipertahankan pada periode 2021-1 dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri.

Jakarta, 23 Agustus 2021

PEMBIMBING TESIS

Pembimbing I : Dr. Dwiza Riana, S.Si, MM,
M.Kom



DEWAN PENGUJI

Penguji I : Dr. Agus Subekti, M.T



Penguji II : Dr. Lindung Parningotan Manik,
M.T.I



Penguji III /
Pembimbing I : Dr. Dwiza Riana, S.Si, MM,
M.Kom





LEMBAR BIMBINGAN TESIS

UNIVERSITAS NUSA MANDIRI

NIM : 14002387
 Nama Lengkap : Fajar Sarasati
 Dosen Pembimbing : Dr. Dwiza Riana, S.Si, M.M, M.Kom
 Judul Tesis : Klasifikasi Citra Penyakit Padi Menggunakan CNN Berdasarkan Segmentasi *K-Means*

No.	Tanggal Bimbingan	Pokok Bahasan	Paraf Dosen Pembimbing
1	10 April 2021	Pengajuan Tema dan Literature Review	<i>Dwiza Riana</i>
2	12 April 2021	Klasifikasi Manual	<i>Dwiza Riana</i>
3	09 Mei 2021	Laporan Pengambilan Dataset dan Eksperimen Awal	<i>Dwiza Riana</i>
4	31 Mei 2021	Laporan Progres Eksperimen	<i>Dwiza Riana</i>
5	09 Juni 2021	Bab I	<i>Dwiza Riana</i>
6	13 Juni 2021	Acc Bab I	<i>Dwiza Riana</i>
7	23 Juni 2021	Bab III	<i>Dwiza Riana</i>
8	27 Juni 2021	Acc Bab III	<i>Dwiza Riana</i>
9	08 Juli 2021	Pengajuan Bab II	<i>Dwiza Riana</i>
10	10 Juli 2021	Acc Bab II	<i>Dwiza Riana</i>
11	16 Juli 2021	Pengajuan Bab IV dan V	<i>Dwiza Riana</i>
12	24 Juli 2021	Revisi Bab IV dan Acc Bab V	<i>Dwiza Riana</i>
13	26 Juli 2021	Acc Bab IV	<i>Dwiza Riana</i>
14	31 Juli 2021	Abstrak, Lampiran dan PPT	<i>Dwiza Riana</i>
15	03 Agustus 2021	Acc Keseluruhan	<i>Dwiza Riana</i>

Catatan untuk Dosen Pembimbing

Bimbingan Tesis

- Dimulai pada tanggal : 10 April 2021
- Diakhiri pada tanggal : 03 Agustus 2021
- Jumlah pertemuan bimbingan : 15

Disetujui oleh,
Dosen Pembimbing



(Dr. Dwiza Riana, S.Si, M.M,
M.Kom)

KATA PENGANTAR

Puji syukur Alhamdulillah, penulis panjatkan kehadiran Allah SWT, yang telah melimpahkan rahmat dan karunia-Nya, sehingga pada akhirnya penulis dapat menyelesaikan laporan tesis ini tepat pada waktunya. Dimana laporan tesis ini penulis sajikan dalam bentuk buku yang sederhana.

Adapun judul tesis, yang penulis ambil sebagai berikut “Klasifikasi Citra Penyakit Padi Menggunakan *CNN* Berdasarkan Segmentasi *K-Means*”.

Tujuan penulisan laporan tesis ini dibuat sebagai salah satu untuk mendapatkan gelar Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri.

Laporan Tesis ini diambil berdasarkan hasil penelitian atau riset sehingga diperoleh citra penyakit padi sebanyak 1275 citra, selanjutnya dilakukan proses klasifikasi manual jenis penyakit padi dengan bantuan ahli sehingga diperoleh 5 jenis penyakit padi dengan berbagai kelas yakni Blas, *Brown Spot*, Hawar, Kresek, dan *Narrow Brown Spot*.

Penulis menyadari bahwa tanpa bimbingan dan dukungan dari semua pihak dalam pembuatan laporan tesis ini, maka penulis tidak dapat menyelesaikan laporan tesis ini tepat waktu. Untuk itu izinkanlah penulis kesempatan ini, untuk mengucapkan

1. Rektor Universitas Nusa Mandiri
2. Wakil Rektor Universitas Nusa Mandiri
3. Dekan Fakultas Teknologi Informasi Universitas Nusa Mandiri
4. Ketua Program Studi Ilmu Komputer
5. Ibu Dr. Dwiza Riana, S.Si, M.M, M.Kom selaku Dosen pembimbing tesis yang telah meluangkan waktu, pikiran serta tenaga selama proses membimbing penulis dalam menyelesaikan tesis ini.
6. Orang tua tercinta yang telah memberikan dukungan material dan moral kepada penulis.
7. Seluruh Dosen Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri yang telah memberikan pelajaran yang berarti bagi penulis selama menempuh studi.

8. Seluruh staf di lingkungan Universitas Nusa Mandiri yang telah memberikan pelayanan kepada penulis selama kuliah.
9. Dan lain-lain

Serta semua pihak yang terlalu banyak untuk disebut satu persatu sehingga terwujudnya laporan tesis ini. Penulis menyadari bahwa penulisan tesis ini masih jauh sekali dari sempurna. Untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan karya ilmiah yang penulis hasilkan untuk yang akan datang.

Akhir kata semoga laporan tesis ini dapat berguna bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya.

Jakarta, 23 Agustus 2021



Fajar Sarasati
Penulis

**SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH
UNTUK KEPENTINGAN AKADEMIS**

Nama : Fajar Sarasati
NIM : 14002387
Program Studi : Ilmu Komputer
Fakultas : Teknologi Informasi
Jenjang : Strata Dua (S2)
Konsentrasi : *Image Processing*
Jenis Karya : Tesis

Demi pengembangan ilmu pengetahuan, dengan ini menyetujui untuk memberikan ijin kepada pihak Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri **Hak Bebas Royalti Non-Eksklusif (*Non-exclusive Royalti-Free Right*)** atas karya ilmiah kami yang berjudul : “Klasifikasi Citra Penyakit Padi Menggunakan *CNN* Berdasarkan Segmentasi *K-Means*” beserta perangkat yang diperlukan (apabila ada).

Dengan **Hak Bebas Royalti Non-Eksklusif** ini pihak Universitas Nusa Mandiri berhak menyimpan, mengalih-media atau bentuk-kan, mengelolanya dalam pangkalan data (*database*), mendistribusikannya dan menampilkan atau mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari kami selama tetap mencantumkan nama kami sebagai penulis/pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Universitas Nusa Mandiri, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 23 Agustus 2021
Yang menyatakan,


Fajar Sarasati

ABSTRAK

Nama : Fajar Sarasati
NIM : 14002387
Program Studi : Ilmu Komputer
Fakultas : Teknologi Informasi
Jenjang : Strata Dua (S2)
Konsentrasi : Image Processing
Judul : Klasifikasi Citra Penyakit Padi Menggunakan *CNN*
Berdasarkan Segmentasi *K-Means*

Usaha sektor pertanian khususnya produksi tanaman padi sering mengalami ketidakpastian akibat penyakit yang menyerang. Tanaman padi yang terinfeksi penyakit jika tidak segera dibasmi beresiko menjangar ke tanaman padi lainnya, sehingga menyebabkan gagal panen. Resiko gagal panen tersebut terjadi akibat kesulitan yang dialami petani dalam mengidentifikasi penyakit padi secara kasat mata dengan cepat dan akurat. Tren teknologi dengan *image processing* yang diterapkan di bidang pertanian untuk mendeteksi penyakit padi sejak dini sehingga diharapkan dapat membantu petani untuk melakukan pengendalian penyebaran penyakit. Penelitian ini bertujuan untuk mengklasifikasikan penyakit padi berdasarkan kategori penyakit yang telah disepakati oleh 9 orang ahli melalui media kuesioner. Dimana dalam mengklasifikasikan penyakit padi ini dilakukan dengan menggunakan arsitektur *EfficientNetB4* menjadi 5 kategori penyakit padi diantaranya Blas, *Brown Spot*, Hawar, Kresek, dan *Narrow Brown Spot* yang sebelumnya dilakukan segmentasi dengan menggunakan metode *K-Means*. Sehingga pada penelitian ini menghasilkan klasifikasi penyakit padi dengan tingkat akurasi sebesar 94.53%.

Kata Kunci : Klasifikasi, Penyakit Padi, *EfficientNetB4*, Segmentasi, *K-Means*

ABSTRACT

Nama : Fajar Sarasati
NIM : 14002387
Study of Program : Ilmu Komputer
Faculty : Teknologi Informasi
Levels : Strata Dua (S2)
Concentration : Image Processing
Title : *Rice Disease Image Classification Using CNN Based on K-Means Segmentation*

The agricultural sector, especially crop production, often experiences uncertainty due to diseases that attack. Infected rice plants if not immediately eradicated are at risk of spreading to other rice plants, causing crop failure. The risk of crop failure occurs due to the difficulties experienced by farmers in identifying diseases with the naked eye quickly and accurately. Technological trends with image processing applied in the field to detect rice diseases early so that it is expected to help farmers to control the disease. This study aims to classify diseases based on disease categories that have been agreed upon by 9 experts through a media questionnaire. Where in classifying rice diseases, this is done using the EfficientNetB4 architecture into 5 categories of diseases such as Blas, Brown Spot, Blight, Crackle, and Narrow Brown Spot which were previously segmented using the K-Means method. So that in this study resulted in the classification of rice diseases with an accuracy rate of 94.53%.

Keywords: Classification, Rice Disease, EfficientNetB4, Segmentation, K-Means

DAFTAR ISI

	Halaman
HALAMAN SAMPUL	i
HALAMAN JUDUL.....	ii
SURAT PERNYATAAN ORISINALITAS DAN BEBAS PLAGIARISME	iii
PERSETUJUAN TESIS	iv
PERSETUJUAN DAN PENGESAHAN.....	v
LEMBAR BIMBINGAN TESIS	vi
KATA PENGANTAR	vii
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS	ix
ABSTRAK	x
ABSTRACT.....	xi
DAFTAR ISI.....	xii
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
DAFTAR LAMPIRAN.....	xvi
BAB 1 PENDAHULUAN	
1.1. Latar Belakang Penulisan	1
1.2. Identifikasi Masalah.....	4
1.3. Tujuan Penelitian	4
1.4. Ruang Lingkup Penelitian.....	5
1.5. Sistematika Penulisan	6
BAB 2 LANDASAN/KERANGKA PEMIKIRAN	
2.1. Tinjauan Pustaka.....	7
2.2. Tinjauan Studi.....	32
2.3. Tinjauan Organisasi/Objek Penelitian	38
BAB 3 METODE PENELITIAN.....	41
BAB 4 HASIL PENELITIAN DAN PEMBAHASAN	45
BAB 5 PENUTUP	66
DAFTAR REFERENSI	68

DAFTAR RIWAYAT HIDUP.....	75
LAMPIRAN.....	76

DAFTAR TABEL

	Halaman
Tabel 2.1. Informasi <i>EfficientNet</i>	24
Tabel 2.2. <i>Channels EfficientNetB0-B4</i>	25
Tabel 2.3. <i>Literature Review</i>	34
Tabel 2.4. Riset Dataset Penyakit Padi (2016-2020)	38
Tabel 3.1. Spesifikasi <i>Hardware</i>	44
Tabel 3.2. Spesifikasi <i>Software</i>	44
Tabel 4.1. Sampel Hasil Akuisisi Citra.....	45
Tabel 4.2. Hasil Klasifikasi Manual oleh Ahli.....	49
Tabel 4.3. Hasil <i>Background Removal</i>	50
Tabel 4.4. Hasil <i>Image Augmentattion</i>	53
Tabel 4.5. Hasil Segmentasi.....	56
Tabel 4.6. <i>Hyperparameter Arsitektur EfficientNetB1</i>	58
Tabel 4.7. Evaluasi Model	61
Tabel 4.8. <i>Classification Report</i>	62
Tabel 4.9. Sampel Hasil Klasifikasi.....	64

DAFTAR GAMBAR

	Halaman
Gambar 2.1. <i>Stem</i>	22
Gambar 2.2. <i>Final Layers</i>	22
Gambar 2.3. <i>Module 1</i>	22
Gambar 2.4. <i>Module 2</i>	22
Gambar 2.5. <i>Module 3</i>	22
Gambar 2.6. <i>Module 4</i>	22
Gambar 2.7. <i>Module 5</i>	23
Gambar 2.8. <i>Sub-Blok 1</i>	23
Gambar 2.9. <i>Sub-Blok 2</i>	23
Gambar 2.10. <i>Sub-Blok 3</i>	24
Gambar 2.11. <i>Aristektur EfficientNetB1</i>	26
Gambar 2.12. <i>Confusion Matrix</i>	28
Gambar 2.13. <i>Kurva Underfit</i>	30
Gambar 2.14. <i>Kurva Overfit</i>	31
Gambar 2.15. <i>Kurva Good Fit</i>	32
Gambar 3.1. <i>Metode Penelitian</i>	41
Gambar 3.2. <i>Pengambilan Citra</i>	44
Gambar 4.2. <i>Hasil Klasifikasi Manual oleh Ahli</i>	48
Gambar 4.3. <i>Tahapan Background Removal</i>	50
Gambar 4.4. <i>Rancangan Model Usulan</i>	57
Gambar 4.5. <i>Training dan Validation Accuracy</i>	60
Gambar 4.6. <i>Training dan Validation Loss</i>	59
Gambar 4.7. <i>Confusion Matrix</i>	63
Gambar 4.8. <i>Hasil Klasifikasi</i>	64

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Pertanyaan Kuesioner	76
Lampiran 2. Hasil Pengisian Kuesioner	89
Lampiran 3. Konfirmasi ke Ahli	95
Lampiran 4. Citra Hasil Akuisisi	96
Lampiran 5. Dataset Penyakit Padi Hasil <i>Augmentation</i> dan Segmentasi	99
Lampiran 6. Tabel Hasil Pengujian Klasifikasi	110
Lampiran 7. <i>Source Code</i> Program	112

BAB I

PENDAHULUAN

1.1. Latar Belakang Penulisan

Padi menjadi salah satu tanaman penghasil bahan pangan pokok yang sangat penting untuk menunjang ketahanan pangan bagi mayoritas orang di dunia, menjadikan petani padi sebagai salah satu ujung tombak sumber pangan di berbagai negara tak terkecuali di Indonesia. Sektor pertanian khususnya produksi tanaman padi sering mengalami resiko ketidakpastian diantaranya gagal panen yang disebabkan oleh perubahan iklim seperti dampak terjadinya banjir, kekeringan, serangan hama bahkan penyakit atau organisme pengganggu tanaman yang berdampak pada kerugian yang dialami petani [1]. Penyakit tersebut jika tidak segera dibasmi maka akan dengan mudah menjalar ke tanaman padi lainnya dan berdampak pada menurunnya kualitas beras yang dihasilkan bahkan akan menyebabkan gagal panen. Penyakit pada tanaman padi juga mempengaruhi perubahan sosial masyarakat di suatu negara khususnya di Negara berkembang, sehingga berdampak terhadap adanya perubahan pada jenis, tingkat serangan, perkembangan, dan laju penyebaran penyakit pada tanaman itu sendiri. Puluhan penyakit tersebut juga dilaporkan dapat mengancam berbagai tanaman pangan budidaya termasuk tanaman padi [2].

Penyakit yang menginfeksi tanaman padi akan menunjukkan gejala berupa bercak dengan pola dan warna tertentu. Dimana gejala tersebut terlihat pada beberapa bagian padi diantaranya malai, daun, batang, dan akar. Namun gejala penyakit padi lebih sering diidentifikasi melalui daun padi karena memiliki penampang yang luas, selain itu perubahan warna dan bercak tampak lebih jelas. Oleh karena itu, daun padi digunakan sebagai tahap awal pada identifikasi penyakit padi [3].

Teknologi yang digunakan oleh petani dalam mengembangkan hasil produksi padi saat ini memerlukan beberapa perbaikan sesuai dengan permasalahan yang mengancam, hal tersebut juga turut mempertimbangkan sumber daya dan lingkungan, termasuk penyakit tanaman yang dapat berkembang dari waktu ke waktu [2]. Pengelolaan penyakit pada tanaman padi penting dilakukan sebagai upaya untuk menjaga stabilitas pangan, karena penyakit yang menyerang tanaman akan berkembang dengan cepat dari waktu ke waktu sehingga dapat mengancam pertumbuhan tanaman padi bahkan berpotensi menyebabkan gagal panen [2]. Balai Besar Penelitian Tanaman Padi (BB Padi) pada tahun 2018 telah merencanakan beberapa program untuk mengendalikan hama dan penyakit padi. Namun berdasarkan laporan tahunan yang dibuat oleh BB Padi menghadapi beberapa kendala dimana kondisi Sumber Daya Manusia (SDM) dari BB Padi yang sebagian besar sudah memasuki masa pensiun, selain itu peralatan yang digunakan di laboratorium maupun di lapangan masih menggunakan peralatan lama, serta kegiatan yang dilakukan tergantung pada musim tanam, kondisi iklim, curah hujan, gangguan hama, serta penyakit dan gulma yang menyerang sehingga pihak BB Padi membutuhkan kerja sama berupa penelitian dengan institusi lain yang telah memiliki peralatan modern [4].

Teknologi revolusi industri 4.0 di sektor pertanian saat ini memanfaatkan teknologi aplikasi pertanian presisi tinggi untuk mengatasi permasalahan dalam bidang pertanian seperti halnya kombinasi bioinformatika dan algoritma genetika untuk mencari bibit unggul dan proses penyilangan kromosom. Disisi lain penggunaan sistem cerdas interpretasi citra juga digunakan untuk perencanaan penggunaan lahan, begitu pula dengan sistem *fuzzy* untuk mendiagnosis hama dan penyakit tumbuhan. Kerusakan panen juga dihindari secara otomatis dengan menggunakan alat uji mutu, optimasi rute terpendek dalam pendistribusian komoditas, dan transparansi aliran barang dan uang dengan teknologi *blockchain* dan *big data* [5]. Salah satu tren penggunaan teknologi dalam bidang pertanian dalam mendeteksi penyakit pada tanaman padi adalah dengan penggunaan metode *classifier* untuk mendeteksi penyakit tanaman padi. Hal tersebut diharapkan dapat membantu

petani untuk mengklasifikasikan penyakit tanaman padi berdasarkan kelasnya atau jenisnya, melakukan pengendalian penyebaran penyakit pada tanaman padi, serta memberikan solusi dalam mengatasi penyakit tanaman padi.

Berdasarkan penelitian sebelumnya Alfarisy [6] mengeksplorasi metode *deep learning caffenet* untuk mengklasifikasikan penyakit padi. Penelitian lain yang dilakukan oleh Saputra [7] juga melakukan klasifikasi citra penyakit padi dengan mengusulkan metode *gray level co-occurrence matrix (GLCM)* sebagai ekstraksi fitur kemudian mengklasifikasikan penyakit menggunakan algoritma *k-nearest neighbors (KNN)* seperti halnya yang deteksi penyakit padi dilakukan oleh Suresha [8] dan Kawcher [9]. Klasifikasi yang dilakukan oleh Zhe [10], Sethy [11] dan Goluguri [12] memanfaatkan metode *deep learning* untuk mengklasifikasikan penyakit padi. Penelitian berikutnya juga melakukan pendeteksian penyakit dan hama citra padi menggunakan arsitektur berskala besar seperti metode *visual geometric group (VGG16)*, *InceptionNet*, *MobileNet*, dan berbagai metode arsitektur berskala besar lainnya namun kendalanya tidak cocok untuk perangkat seluler [13]. Deteksi dan klasifikasi penyakit padi juga dilakukan dengan menggunakan *machine learning* [14] [15] [16], deteksi penyakit juga dilakukan menggunakan arsitektur berskala besar [13], sedangkan Pothen [17] melakukan deteksi menggunakan segmentasi *local binary pattern (LBP)* dan *histogram of oriented gradient (HOG)* untuk selanjutnya dilakukan klasifikasi *support vector machine (SVM)*. Begitu pula dengan penelitian yang dilakukan dengan prediksi penyakit padi yang fokus pada metode deteksi segmentasi citra [18] [19].

Berdasarkan permasalahan yang dijabarkan di atas, maka peneliti tertarik melakukan penelitian dengan judul “klasifikasi citra penyakit padi menggunakan *CNN* berdasarkan segmentasi *k-means*”, dimana dalam laporan penelitian ini penulis mengusulkan metode *convolutional neural network (CNN)* untuk mengklasifikasikan penyakit padi menggunakan arsitektur *EfficientNetB4* dengan dilakukan proses segmentasi terlebih dahulu untuk memisahkan objek atau *foreground* dengan *background* pada daun padi, sehingga diperoleh tingkat keakuratan hasil klasifikasi lebih baik dibanding dengan tanpa disegmentasi terlebih dahulu. Metode *CNN* ini

dipilih sebagai upaya mengikuti perkembangan teknologi, sehingga diharapkan pengklasifikasian penyakit padi dengan metode tersebut dapat membantu petani dalam melakukan pendeteksian dini penyakit padi sebagai upaya pengendalian penyebaran penyakit sehingga dapat dilakukan pencegahan gagal panen.

1.2. Identifikasi Masalah

Berdasarkan permasalahan yang dibahas dalam latar belakang, maka rumusan masalah pada penelitian ini yaitu:

1. Bagaimana proses pengambilan citra penyakit padi?
2. Bagaimana proses penentuan penyakit padi berdasarkan citra yang telah diambil?
3. Apakah proses data *augmentation* dapat menghasilkan citra yang lebih beragam?
4. Bagaimana segmentasi citra dapat mendeteksi objek bercak penyakit padi?
5. Apakah metode *CNN* dapat melakukan klasifikasi terhadap penyakit padi?

1.3. Tujuan Penelitian

Tujuan yang ingin dicapai dari penelitian ini adalah:

1. Untuk melakukan pengambilan citra penyakit padi dengan bantuan kamera *handphone* Samsung A10.
2. Untuk melakukan klasifikasi manual terhadap citra yang telah diambil sebagai upaya memperoleh jenis penyakit padi yang disepakati oleh ahli dengan menyebar kuesioner.
3. Untuk melakukan manipulasi data sehingga diperoleh citra yang lebih beragam.
4. Untuk mendeteksi objek dalam hal ini bercak penyakit pada daun padi.
5. Untuk melakukan klasifikasi penyakit padi sebagai upaya pendeteksian dini terhadap penyakit padi

Sedangkan tujuan dari penyusunan tesis ini adalah untuk melengkapi salah satu syarat yang telah ditentukan dalam mencapai kelulusan Program Magister (S2) Program Studi Ilmu Komputer Universitas Nusa Mandiri.

1.4. Ruang Lingkup Penelitian

Ruang lingkup yang digunakan penulis pada penyusunan laporan tesis ini ditujukan untuk membatasi pokok permasalahan yang akan dibahas agar tidak menyimpang pada permasalahan pokok, maka batasan yang dibahas oleh penulis antara lain:

1. Permasalahan yang diangkat penulis hanya berkaitan dengan masalah penyakit pada tanaman padi, tidak termasuk hama yang menyerang tanaman padi.
2. Data yang digunakan oleh penulis merupakan data *private* yang sampelnya diambil di tiga lokasi di Indonesia, yakni di kabupaten Klaten di provinsi Jawa Tengah, kabupaten Sukabumi di provinsi Jawa Barat dan kabupaten Gunung Kidul di provinsi Daerah Istimewa Yogyakarta. Hasil pengambilan sampel diperoleh lima kategori penyakit padi yang meliputi penyakit blas, *brown spot*, hawar, kresak, dan *narrow brown spot* dimana masing-masing kategori memiliki 15 citra utama.
3. Pengambilan atau akuisisi citra menggunakan kamera *handphone* Samsung A10.
4. Proses klasifikasi manual oleh ahli dilakukan dengan menyebar kuesioner yang diisi oleh 9 responden yang merupakan ahli pertanian dari 7 Universitas atau instansi pertanian.
5. Citra yang digunakan pada penelitian ini menggunakan citra dengan tipe data *joint photographic group* (JPG), dengan jenis citra *red*, *green*, *blue* (RGB) dan berdimensi 774 x 1032 pixel.
6. Proses pengklasifikasian yang digunakan penulis menggunakan metode *CNN* dengan algoritma *EfficientNetB4* untuk melakukan klasifikasi terhadap penyakit padi.

1.5. Sistematika Penulisan

Sistematika penulisan yang dilakukan oleh penulis dalam penulisan laporan tesis ini meliputi:

BAB I	Pendahuluan Bab ini mencakup pembahasan mengenai latar belakang permasalahan yang diangkat oleh penulis untuk selanjutnya dijabarkan solusi yang diusulkan untuk mengatasi masalah, tujuan penelitian, ruang lingkup penelitian, dan sistematika penulisan.
BAB II	Landasan/Kerangka Penikiran Bab ini membahas mengenai tinjauan pustaka yang berisi teori pendukung, tinjauan studi yang mencakup penelitian terdahulu atau <i>literature review</i> yang berkaitan dengan penelitian yang dilakukan dan tinjauan obyek penelitian yang berisi obyek permasalahan yang akan diangkat oleh penulis pada penelitian ini
BAB III	Metodologi Penelitian Bab ini mengemukakan tentang metode penelitian serta alur penelitian sebagai acuan utama selama penelitian berlangsung termasuk didalamnya diagram tahapan penelitian, selain itu pada bab ini juga membahas pengolahan data awal, metode usulan dan evaluasi.
BAB IV	Hasil Penelitian dan Pembahasan Bab ini membahas hasil penelitian dan pembahasan yang berisi penjelasan hasil penelitian menyajikan hasil penelitian.
BAB V	Penutup Pada bab ini berisi kesimpulan dan saran dari hasil penelitian yang telah dilakukan. Kesimpulan berisi juga dibahas apakah identifikasi masalah dan tujuan yang telah diuraikan di bab I mampu terjawab didalam pembahasan serta pemberian saran untuk penelitian selanjutnya.

BAB II

LANDASAN/KERANGKA PEMIKIRAN

2.1 . Tinjauan Pustaka

Pada bagian ini penulis menguraikan teori-teori yang terkait dengan objek penelitian, pengenalan citra digital, teknik pengolahan citra, hingga metode-metode yang digunakan dalam mengklasifikasikan penyakit padi. Berikut beberapa teori yang digunakan oleh penulis:

2.1.1. Penyakit Tanaman

Penyakit merupakan permasalahan utama yang berakibat pada menurunkan produksi padi. Saat ini terdapat sekitar 60 macam penyakit yang berdampak pada menurunkan padi dan 33 jenis penyakit tersebut diantaranya disebabkan oleh infeksi jamur [20].

Beberapa macam penyakit padi yang digunakan dalam penelitian ini, meliputi:

1. Blas

Pada awalnya penyakit blas dikenal sebagai salah satu kendala yang terjadi pada padi gogo, namun sejak akhir tahun 1980-an penyakit ini mulai menyerang padi pada sawah beririgasi. Penyakit blas ini disebabkan oleh jamur *pathogen pyricularia*. Penyakit ini juga dapat menampakkan dua gejala khas, yaitu bercak coklat kehitaman berbentuk belah ketupat dimana pusat bercak pada daun berwarna putih [4].

Penyakit ini juga dapat menginfeksi tanaman padi pada semua stadia pertumbuhan. Bercak berbentuk belah ketupat lebar ditengah daun dan meruncing di kedua ujungnya menjadi gejala khas penyakit ini. Dimana bercak tersebut kira-kira berukuran 1-1,5 x 0,3-0,5 cm kemudian bercak tersebut menjadi berwarna abu-abu pada bagian tengah daunnya [20].

Cara pengendalian pada penyakit blas meliputi:

- a. Menggunakan varietas tahan blas secara bergantian
- b. Menggunakan pupuk nitrogen sesuai anjuran
- c. Mengupayakan waktu tanam dengan tepat, supaya waktu awal pembungaan tidak banyak embun dan hujan secara terus menerus.
- d. Menggunakan fungisida yang berbahan aktif seperti *metil tiofanat* atau *fosdifen* dan *kasugamisin*.
- e. Memperhatikan perlakuan pada benih

2. *Brown Spot*

Penyakit *Brown Spot* ini sering disebut bercak coklat dimana penyakit ini disebabkan oleh jamur *helminthosporium oryzae*. Penyakit ini dapat beresiko menyebabkan kematian pada tanaman muda dan menurunkan kualitas gabah. Bercak daun berwarna coklat, berbentuk oval hingga bulat berukuran sebesar biji wijen pada permukaan daun menjadi gejala yang paling umum pada penyakit ini [4].

3. Hawar

Hawar daun bakteri atau biasa disebut HDB merupakan penyakit yang disebabkan oleh bakteri yang tersebar luas dan dapat beresiko menurunkan hasil produksi hingga 36%. Penyakit ini sering terjadi pada musim hujan atau musim kemarau basah, terutama pada lahan sawah yang selalu tergenang. Penyakit HDB menghasilkan dua gejala khas yaitu kresak dan hawar [4].

Bakteri *xanthomonas campestris* pv *oryzae* ini merupakan penyebab penyakit HDB dengan gejala berupa bercak berwarna kuning sampai putih. Penyakit ini berawal dari terbentuknya garis lebam berair pada bagian tepi daun [20].

Cara pengendalian penyakit HDB adalah sebagai berikut:

- a. Menggunakan varietas padi tahan seperti conde dan angke
- b. Menggunakan pupuk nitrogen sesuai dengan kebutuhan tanaman
- c. Bersihkan tunggul-tunggul dan jerami-jerami yang terinfeksi

- d. Jarak tanam diusahakan jangan terlalu rapat
- e. Menggunakan benih atau bibit yang sehat

4. Kresek

Kresek merupakan gejala penyakit yang terjadi pada tanaman berumur <30 hari dengan gejala daun berwarna hijau kelabu, melipat dan menggulung [4].

5. *Narrow Brown Spot*

Narrow brown spot atau bercak *cercospora* merupakan penyakit tanaman padi yang disebabkan oleh jamur *cercospora oryzae*. Penyakit ini menampilkan gejala lurus sempit berwarna coklat pada helaian daun bendera, khususnya pada fase tumbuh pemasakan [4].

2.1.2. *Dataset*

Dataset dapat disebut sebagai kumpulan objek data. *Dataset* dengan kata lain sering digunakan adalah *record*, *point*, *vector*, *pattern*, *event*, *observation*, *case* atau bahkan data [21].

Dataset dapat berupa obyek yang meliputi kasus, *record* dan titik serta atribut yang meliputi fitur, *field*, karakteristik dan variabel [22].

Jenis dataset meliputi dua macam, yakni *private* dan *public* [22]:

- a. *Private dataset*, adalah *dataset* yang biasa diambil dari organisasi yang dijadikan obyek penelitian, sebagai contoh data dari bank, rumah sakit, pabrik, perusahaan jasa, dan lain-lain.
- b. *Public dataset*, adalah *dataset* yang biasa diambil dari repositori publik yang telah disepakati oleh para peneliti, sebagai contoh *dataset UCI Repository* (<http://www.ics.uci.edu/~mllearn/MLRepository.html>) dan *ACM KDD Cup* (<http://www.sigkdd.org/kddcup/>).

2.1.3. Citra Digital

Citra digital merupakan komponen multimedia yang berperan dalam penyajian informasi visual [23]. Citra atau *image* dapat disebut pula sebagai gambar pada bidang dwimatra atau dua dimensi atau biasa juga disebut sebagai kumpulan warna yang terlihat indah, memiliki pola, berbentuk abstrak dan ciri lainnya [24].

2.1.4. Format File Citra

File menyediakan mekanisme penting untuk penyimpanan, pengarsipan, dan bertukar data citra. Menggunakan format file standar meningkatkan kemudahan dalam pertukaran citra dan memungkinkan citra akan dapat dibaca oleh perangkat lunak lain dalam jangka panjang [25]. Berikut beberapa kriteria format file citra yang digunakan penulis:

1. *JPEG (Joint Photographic Experts Group)*

Standar *JPEG* mendefinisikan metode kompresi untuk *continuous* citra skala abu-abu dan berwarna. Standar *JPEG* mendukung citra dengan komponen warna hingga 256 dan yang semakin penting adalah dukungan untuk citra *CMYK* [25].

2. *TIFF (Tagged Image File Format)*

Format file ini banyak digunakan dan fleksibel untuk memenuhi kebutuhan profesional di berbagai bidang. Format file ini mendukung berbagai citra skala abu-abu, diindeks, dan warna asli tetapi juga tipe citra khusus. File *TIFF* ini dapat berisi sejumlah citra dengan property. Spesifikasi *TIFF* menyediakan berbagai metode kompresi yang berbeda (*LZW*, *ZIP*, *CCITT*, dan *JPEG*) dan ruang warna [25].

3. *GIF (Graphics Interchange Format)*

GIF pada dasarnya adalah format file citra terindeks yang dirancang untuk warna dan skala abu-abu dengan kedalaman maksimum 8 bit dan akibatnya tidak mendukung citra warna asli. Format ini menawarkan dukungan efisien untuk palet pengkodean yang berisi dari 2 hingga 256 warna, salah satunya dapat ditandai untuk transparansi [25].

4. *PNG (Portable Network Graphics)*

PNG mendukung tiga jenis citra, yakni citra berwarna asli hingga 3 x 16 bit/piksel, citra skala abu-abu hingga 16 bit/piksel, dan citra berwarna yang diindeks hingga 256 warna [25].

5. *JFIF (JPEG File Interchange Format)*

JFIF menentukan format file berdasarkan standar *JPEG* dengan mendefinisikan elemen penting yang tersisa dari format file [25].

6. *EXIF (Exchangeable Image File Format)*

Format file citra *EXIF* adalah varian dari format *JPEG (JFIF)* yang dirancang untuk menyimpan data citra yang berasal dari kamera digital dan untuk mendukung penyimpanan metadata seperti jenis kamera, tanggal dan waktu, parameter fotografi seperti aperture dan waktu eksposur serta geografis data (*GPS*) [25].

7. *BPM (Windows Bitmap)*

BPM merupakan format sederhana di bawah Windows yang banyak digunakan untuk mendukung skala abu-abu, diindeks dan citra berwarna [25].

8. *PBM (Portable Bitmap Format)*

PBM merupakan format file yang sangat sederhana karena disimpan dalam format teks yang dapat dibaca oleh manusia dengan mudah atau cukup diedit menggunakan *editor teks* [25].

2.1.5. *Google Colaboratory*

Proyek *colab* dari *Google* membuat *platform* yang diadopsi dari *Jupyter Notenook* namun memanfaatkan *Google* sebagai sumber dayanya. Didalam *Google Colab* juga tersedia fitur *sharing* yang menjadi solusi untuk pembatasan akses. Oleh karena itu *Google Colab* juga mendukung *sharing* seperti halnya *Google Drive* [26].

2.1.6. *Image Processing*

Istilah “*image processing*” sering digunakan secara bergantian dengan istilah “*image editing*”. Pengeditan citra digital atau disebut pencitraan digital merupakan

manipulasi citra digital yang memanfaatkan aplikasi perangkat lunak yang tersedia secara *open source* seperti Adobe Photoshop atau Corel Paint. Pengolahan citra digital di sisi lain merupakan konsepsi, desain, pengembangan dan program peningkatan pencitraan digital [25].

Dalam lingkungan modern saat ini pengolahan citra digital dilakukan dengan *APIs (Application programming interfaces)* yang meliputi hampir setiap aspek komputasi, basis data, grafik, suara atau pencitraan [25].

2.1.7. Akuisisi Citra

Akuisisi citra merupakan proses mendapatkan representasi digital dari sebuah adegan. Dimana representasi ini dikenal sebagai sebuah citra dan elemen-elemennya disebut piksel atau elemen citra [27]. Akuisisi citra juga dapat disebut tahap awal dengan menangkap atau memindai suatu citra analog sehingga diperoleh citra digital [28].

Beberapa teknologi yang biasa digunakan untuk menangkap sensor pencitraan meliputi *charge-coupled device (CCD)* dan *complementary metal oxide semiconductor (CMOS)* yang merupakan teknologi paling luas yang digunakan dalam sensor citra [27].

Akuisisi citra juga bisa disebut proses dimana adegan diubah menjadi citra digital yang bervariasi dan rumit. Dasar dari sebagian besar metode akuisisi citra adalah variasi pada kamera optik yang dimulai dengan memeriksanya secara rinci [25].

2.1.8. Pre-Processing

Pre-processing merupakan tahapan dalam pengolahan citra [28]. Peningkatan resolusi citra dan teknik *pre-processing* untuk membersihkan atau menghilangkan *noise*. Menghapus bagian yang terkena *noise* dari citra tanpa merusak tepi citra merupakan bagian akhir dari *pre-processing* [29].

Beberapa proses yang dilakukan dalam tahap *pre-processing* meliputi:

1. *Image Resize*

Image resize merupakan proses memperoleh resolusi citra yang ideal untuk dapat digunakan dalam proses pengolahan citra. Proses mengubah ukuran citra ini dimaksudkan untuk mempermudah proses eksekusi, proses ini memiliki tujuan karena semakin besar jumlah pixel pada citra yang diproses maka semakin lama program tersebut mengeksekusi citra [24].

2. *Background Removal*

Salah satu situs yang dapat digunakan untuk menghapus *background* ialah situs <https://www.remove.bg/>. Dimana situs ini digunakan untuk menghapus latar belakang gambar secara otomatis dan mempermudah penyimpanan dalam format *PNG* ataupun *JPG*. Situs tersebut juga dilengkapi dengan menu mengganti *background* lain sesuai keinginan [30]. Cara menghapus *background* gambar pada situs *Remove BG* adalah sebagai berikut [30]:

- a. Buka situs <https://www.remove.bg/>
- b. Klik *upload image* dan pilih foto yang akan dihilangkan *background*-nya
- c. Setelah citra terunggah dengan sempurna, *background* pada citra yang diunggah akan langsung hilang dan dapat langsung diunduh.

3. *Image Augmentation*

Augmentation merupakan proses mengubah atau memodifikasi citra sedemikian rupa sehingga computer akan mendeteksi bahwa citra yang diubah adalah citra yang berbeda, namun dari pandangan manusia mengatakan bahwa citra yang diubah itu masih sama. Proses *augmentation* sendiri dapat meningkatkan akurasi model *CNN* yang dilatih karena melalui model *augmentation* ini dapat menambah data yang berguna untuk membuat model yang dapat menggeneralisasi lebih baik [23].

Selain itu proses *image augmentation* yang digunakan untuk menambah jumlah citra dalam kumpulan data *train* ini juga dimaksudkan untuk mencegah kemungkinan terjadinya *overfitting* dalam proses *training*. Dimana

overfitting adalah masalah ketika jaringan mempelajari data daripada pola umum dari kumpulan data. *Image augmentation* dilakukan dengan memperkenalkan variasi tertentu dalam citra seperti rotasi, pergeseran lebar, pergeseran tinggi, rentang geser, *flip horizontal* [31].

Augmentasi data diperlukan untuk menambah kumpulan data yang dikumpulkan untuk meningkatkan ukuran dataset dan memperkenalkan beberapa variasi distorsi pada citra yang tujuan utamanya untuk mencegah masalah *overfit* yang disebabkan oleh dataset kecil untuk melatih model *deep learning* [6].

Sejumlah cara yang digunakan pada teknik augmentasi data diantaranya: [32].

a. *Flip*

Flip yaitu pembalikan citra yang dilakukan baik secara horizontal dan vertikal. Misalnya citra wajah yang diubah menjadi menghadap ke kiri kemudian dibalik menjadi menghadap kanan.

b. *Random Cropping*

Random cropping digunakan untuk memperoleh variasi data dari berbagai sudut pandang.

c. *Color Jitter*

Color jitter digunakan untuk memperoleh variasi warna secara acak. Dalam teknik ini cara paling sederhananya adalah dengan melakukan variasi kontras secara acak.

d. *Random Combination*

Random combination merupakan berbagai teknik pengolahan citra diantaranya *translation* atau pembesaran atau pengecilan citra, *rotation* atau pemutaran citra dengan derajat tertentu, *stretching* atau peregangan dan pengerutan citra, *shearing*, dan *lens distortions* pada citra yang akan diolah.

2.1.8. Segmentasi

Tujuan segmentasi membagi citra ke dalam segmen-segmen yang berbeda, membagi daerah yang membutuhkan fokus utama dibandingkan daerah lainnya yakni latar belakang [33].

1. *K-Means Clustering*

K-Means clustering merupakan metode yang digunakan untuk memperoleh deskripsi dari sekelompok data dengan mengungkapkan kecenderungan pada setiap data lainnya [34].

Teknik ini memiliki ide dasar untuk menemukan pusat data dari sekelompok data untuk dapat mengelompokkan data secara individu kedalam salah satu dari sekelompok data berdasarkan jaraknya. Seperti hitungan matematis berikut:

$$d_{ij} = \sqrt{\sum_{k=1}^p \{x_{ik} - x_{jk}\}^2} \dots \dots \dots (2.1)$$

Sumber: [34]

Dimana:

d_{ij} = Jarak antar Objek

p = Dimensi Data

x_{ik} = Dimensi i dan k

x_{jk} = Dimensi j dan k

K-means clustering digunakan untuk klasifikasi objek berdasarkan sekumpulan fitur menjadi K sejumlah kelas. Klasifikasi dilakukan dengan meminimumkan jumlah kuadrat jarak antara benda dan klaster yang sesuai [19]. Algoritma *K-Means* ini juga disebut sebagai *flat clustering*, karena kelompok satu memiliki kedudukan sejajar dengan kelompok lainnya [35].

Algoritma *K-Means clustering* terdiri dari beberapa tahapan sebagai berikut [19]:

- a. Menghitung penyebaran intensitas (histogram)
- b. Inisialisasi *centroid* dengan intensitas acak k
- c. Mengulang langkah samp label *cluster* citra tidak berubah lagi
- d. Titik *cluster* berdasarkan atas jarak intensitas *centroid*
- e. Hitung *centroid* baru untuk masing-masing *cluster*

2.1.9. Partisi Data

Pada umumnya suatu data memiliki data *training*, *validation* dan *testing*. Mesin yang dilatih menggunakan data *training* diukur berdasarkan kemampuan mengenali atau mengeneralisasi data *validation* [35].

1. Data Train

Data train atau *training set* adalah himpunan data yang digunakan untuk membangun model atau melatih model [35].

2. Data Test

Data *test* atau *testing set* merupakan himpunan data yang digunakan untuk menguji model setelah proses latihan selesai. *Testing set* disini merupakan *unseen data* artinya model atau manusia tidak boleh melihat sampel saat proses latihan [35].

3. Data Validation

Data *validation* merupakan himpunan data yang digunakan untuk mengoptimisasi saat melatih model. Model ini dilatih menggunakan *training set* dan pada umumnya kinerja saat latihan diuji dengan *validation set*. Dalam hal ini berguna untuk generalisasi agar model mampu mengenali pola secara generik [35].

Dalam pembagian data set pengujian harus memenuhi dua kondisi sebagai berikut [36]:

1. Cukup menghasilkan hasil yang bermakna secara statistik.

2. Mewakili kumpulan data secara keseluruhan. Dengan kata lain tidak diperkenankan memilih data *test* dengan karakteristik yang berbeda dari data *train*.

2.1.10. Classification

Classification merupakan proses melakukan generalisasi struktur atau model yang telah diketahui untuk dapat diaplikasikan pada data-data baru. Contohnya, klasifikasi penyakit ke dalam sejumlah kategori, klasifikasi email ke dalam *spam* atau bukan [37].

Classification adalah proses pengelompokan, yang berarti mengumpulkan objek atau entitas yang sama dan memisahkan objek atau entitas yang tidak sama [38].

2.1.11. Transfer Learning

Dalam *neural network* pengumpulan data *train* penting dilakukan, namun persyaratan tersebut sulit diperoleh karena data berlabel hanya diperoleh melalui proses manual yang memakan waktu dan rentan terhadap kesalahan. Oleh karena itu tujuan dari perancangan *transfer learning* sebagai cara efektif untuk mengeksplor pengetahuan yang diekstraksi dari domain target semula [39].

Transfer learning juga memiliki tujuan untuk mentransfer pengetahuan dari kumpulan data besar yang dikenal sebagai domain sumber ke kumpulan data yang lebih kecil bernama domain target [40]. *Transfer learning* menyediakan cara yang efektif dalam melatih jaringan besar menggunakan pelatihan yang langka tanpa ada *overfitting* data [40].

2.1.12. Deep Learning (DL)

Deep learning adalah evolusi selanjutnya dari *machine learning*. Sehingga sebagian pakar *machine learning* menggolongkan *DL* dalam wilayah berbeda. Sebuah model *machine learning* yang umum perlu diketahui bagaimana dia menghasilkan

prediksi dengan cara diinputkan data. Sementara model *DL* dapat mempelajari metode komputasinya sendiri [41].

Deep learning merupakan subbidang *machine learning* yang berkaitan dengan algoritma yang terinspirasi oleh struktur dan fungsi otak yang disebut *artificial neural network* [42].

Deep learning sudah dikembangkan dalam berbagai model atau arsitektur diantaranya [43]:

1. *Recurrent Neural Networks (RNN)*

Berbagai pengaplikasian *RNN* seperti *speed recognition*, *handwriting recognition*

2. *LSTM/GRU Networks*

Berbagai pengaplikasian *LSTM/GRU Networks* meliputi *natural language text compression*, *handwriting recognition*, *speech recognition*, *gesture recognition*, *image captioning*

3. *Convolutional Neural Networks (CNN)*

Pengaplikasian *CNN* meliputi *image recognition*, *video analysis*, *natural language processing*

4. *Deep Belief Networks (DBN)*

Pengaplikasian *DBN* meliputi *image recognition*, *information retrieval*, *natural language understanding failure prediction*

5. *Deep Stacking Networks (DSN)*

Pengaplikasian *DSN* meliputi *information retrieval*, *continuous speech recognition*

2.1.13. Convolutional Neural Network (CNN)

Pada awal kemunculan arsitektur *CNN* didesain dengan hanya berisi sedikit lapisan (kurang dari sepuluh). Para ahli mendesain *CNN* dengan ratusan bahkan ribuan lapisan. *CNN* dikembangkan berdasarkan *Multilayer Layer Perceptron (MLP)* yang dirancang untuk memproses data dua dimensi dalam bentuk citra [23].

Arsitektur yang mampu mengenali informasi prediktif suatu objek (gambar, teks, potongan suara, dan sebagainya) meskipun objek tersebut dapat diposisikan dimanapun saat diinput merupakan kemampuan utama dalam *CNN* [35].

CNN atau *ConvNet* juga dapat disebut sebagai salah satu model di dalam *deep learning* yang digunakan untuk keperluan analisa citra atau visual. *CNN* memiliki neuron yang disusun secara tiga dimensi, sehingga memiliki panjang, lebar dan tinggi. Sehingga *ConvNet* efektif dan efisien untuk menganalisa citra [43].

2.1.14. Layer CNN

Secara umum cara kerja *CNN* dibagi menjadi empat bagian/tahap/layer [41], yaitu:

1. *Convolution*

Layer-layer pertama yang berfungsi untuk menerima *input signal*. Pada *CNN*, *convolution* merupakan sebuah proses di mana *network* mencoba untuk memberi label pada sinyal *input* dengan cara merujuk apa yang sudah dipelajari di masa lalu. Secara umum *convolution* adalah operasi antar dua *function* sehingga menghasilkan *function* ketiga yang merupakan hasil modifikasi dari kedua *function* [41].

2. *Subsampling*

Input yang berasal dari *convolution layer* diperhalus untuk mengurangi tingkat sensitifitas terhadap *noise* dan *variations*. Proses memperhalus ini disebut *subsampling*, dan dapat dicapai dengan cara mengambil nilai rata-rata atau mengambil nilai maksimum dan beberapa buah sinyal. Contoh *subsampling* yaitu dengan mengurangi ukuran gambar atau mengurangi kontras warna (*RGB channel*) [41].

3. *Activation*

Layer activation mengendalikan bagaimana sinyal mengalir dari satu *layer* ke *layer* berikutnya, mengikuti cara kerja *neuron-neuron*. Sinyal *output* yang terkait dengan referensi sebelumnya akan mengaktifkan banyak *neuron* yang lain, sehingga sinyal dapat dipropagasi secara efisien [41].

Activation Function dihitung setelah operasi konvolusi dilakukan. *Activation Function* ini digunakan dalam *CNN* termasuk didalamnya *tanh()*, *ReLU* (*Rectified Linear Unit*), *sigmoid*, dan *softmax* [23].

a. Relu

ReLU layer atau lapisan *Rectified Linear Units* ini mengaplikasikan fungsi aktivasi $f(x) = \max(0, x)$. fungsi dari *activation function* ini untuk meningkatkan sifat nonlinearitas fungsi keputusan dan jaringan secara keseluruhan tanpa memengaruhi bidang-bidang reseptif pada *convolution layer* [41].

Fungsi Relu merupakan nilai keluaran suatu neuron yang dinyatakan sebagai 0 jika nilai masukannya negatif, begitu juga sebaliknya jika nilai masukannya positif maka keluaran dari neuron adalah nilai masukannya itu sendiri [23]. Berikut perhitungan matematis dari fungsi Relu:

$$f(x) = \max(0, x) \dots\dots\dots(2.2)$$

Sumber : [23]

b. Softmax

Softmax berfungsi untuk mengubah *output* dalam *neural network* menjadi probabilitas distribusi dasar. Aktivasi *softmax* biasa diterapkan pada lapisan terakhir pada *neural network*. *Softmax* ini lebih umum digunakan daripada *Relu*, *sigmoid* atau *tanh()* [44].

$$f(x) = \frac{e^{x_i}}{\sum_{j=1}^k e^{x_j}} \text{ for } i = 1, 2, 3, \dots, k \dots\dots\dots(2.3)$$

Sumber : [44]

4. Fully Connected

Fully Connected layer adalah *multi layer perceptron (MLP)* yang menggunakan *softmax activation function*. *Fully Connected* menunjukkan bahwa setiap *neuron* pada *layer* sebelumnya terkoneksi dengan setiap *neuron* pada *layer* berikutnya. Tujuan dari *Fully Connected* adalah untuk klasifikasi (*class*) *image* [41].

2.1.15. Optimasi

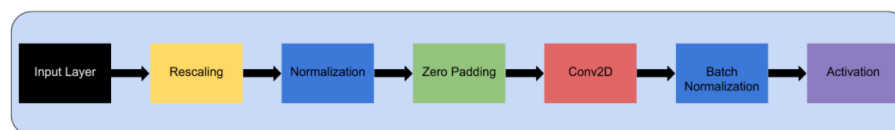
Adam optimizer adalah algoritma optimasi yang digunakan sebagai pengganti prosedur stokstik gradient klasik yang memperbarui bobot network berdasarkan iteratif tanpa mengubah *learning rate* [44]. *Adam optimizer* digunakan untuk meminimalisir *loss function* [45].

2.1.16. EfficientNet

Model *EfficientNet* terdiri dari 8 model mulai dari B0 hingga B7, dimana masing-masing nomor model berikutnya mengacu pada varian dengan lebih banyak parameter dan akurasi yang lebih tinggi. Arsitektur *EfficientNet* menggunakan *transfer learning* untuk menghemat waktu dan kekuatan komputasi, sehingga memberikan akurasi yang lebih tinggi dari model pesaing yang dikenal. Hal tersebut karena penggunaan penskala yang cerdas pada kedalaman, lebar dan resolusi [46]. Jumlah total lapisan dalam *EfficientNet-B0* adalah 237 sedangkan dalam *EfficientNet-B7* totalnya menjadi 813 [47].

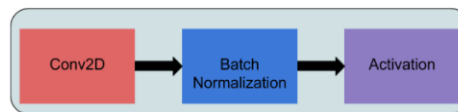
Berdasarkan pengamatan yang dilakukan oleh Linh T. Duong [39], akurasi dan efisiensi yang lebih baik diperoleh dengan memaksakan keseimbangan antara semua dimensi jaringan. Model *EfficientNet* telah diusulkan untuk meningkatkan kinerja *CNN* dengan penskalaan dalam tiga dimensi, yakni lebar, kedalaman, dan resolusi menggunakan sebuah set koefisien skala tetap yang memenuhi beberapa kendala.

Arsitektur umum pada model *EfficientNet* dapat digambarkan dalam bentuk *stem*, *final layers*, *module* 1-5, seperti pada Gambar 2.1 - 2.7 berikut:



Sumber : [47]

Gambar 2.1 Stem



Sumber : [47]

Gambar 2.2 Final Layers



Sumber : [47]

Gambar 2.3 Module 1

Module1 sebagai titik awal untuk sub-blok



Sumber : [47]

Gambar 2.4 Module 2

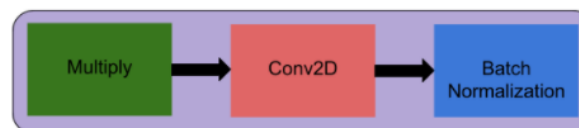
Module 2 sebagai titik awal untuk sub-blok pertama dari semua 7 blok utama kecuali yang pertama



Sumber : [47]

Gambar 2.5 Module 3

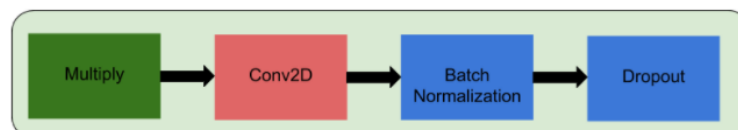
Module 3 sebagai koneksi yang melewati semua sub-blok



Sumber : [47]

Gambar 2.6 Module 4

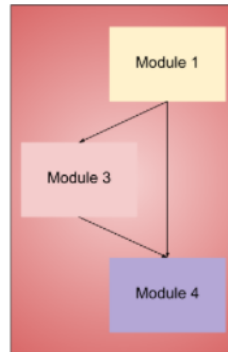
Module 4 berfungsi untuk menggabungkan koneksi yang melewati sub-blok pertama



Sumber : [47]

Gambar 2.7 Module 5

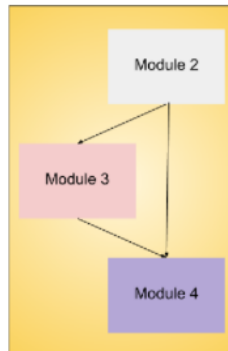
Module 5 sebagai sub-blok yang terhubung ke sub-blok sebelumnya dalam koneksi melewati dan digabungkan menggunakan modul ini.



Sumber : [47]

Gambar 2.8 Sub-Blok 1

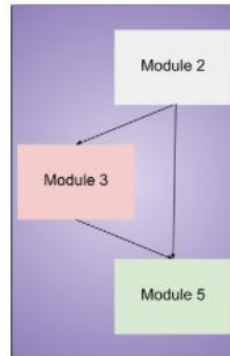
Sub-Blok 1 digunakan sebagai sub-blok pertama di blok pertama



Sumber : [47]

Gambar 2.9 Sub-Blok 2

Sub-Blok 2 digunakan sebagai sub-blok pertama di semua blok lainnya



Sumber : [47]

Gambar 2.10 Sub-Blok 3

Sub-Blok 3 digunakan untuk setiap sub-blok kecuali yang pertama di semua blok

Tabel 2.1 menunjukkan struktur jaringan dasar yang terdiri dari ukuran kernel, resolusi, *channel*, dan nomor dari informasi *layers* pada model *EfficientNetB0*:

Tabel 2.1 Struktur Jaringan Dasar *EfficientNetB0*

<i>Stage</i>	<i>Operator</i>	<i>Resolution</i>	<i>Layers</i>	<i>Channels</i>
1	<i>Conv3x3</i>	224 x 224	1	32
2	<i>MBConv1, k3x3</i>	112 x 112	1	16
3	<i>MBConv6, k3x3</i>	112 x 112	2	24
4	<i>MBConv6, k5x5</i>	56 x 56	2	40
5	<i>MBConv6, k5x5</i>	28 x 28	3	80
6	<i>MBConv6, k5x5</i>	14 x 14	3	112
7	<i>MBConv6, k5x5</i>	14 x 14	4	192
8	<i>MBConv6, k3x3</i>	7 x 7	1	320
9	<i>Conv1x1 & Pooling & FC</i>	7 x 7	1	1280

Sumber : [47]

Sedangkan pada Tabel 2.2 menginformasikan mengenai perbedaan *channel* pada arsitektur *EfficientNetB0* hingga *EfficientNetB4*:

Tabel 2.2 Channel EfficientNet B0-B7

B0	B1	B2	B3	B4	B5	B6	B7
32	32	32	40	48	48	56	64
16	16	16	24	24	24	32	32
24	24	24	32	32	40	40	48
40	40	48	48	56	64	72	80
80	80	88	96	112	128	144	160
112	112	120	136	160	176	200	224
192	192	208	232	272	304	344	384
320	320	352	384	448	512	576	640
1280	1280	1408	1536	1792	2304	2304	2560

Sumber : [47]

1. *EfficientNetB4*

EfficientNetB4 menerima *inputan* berupa *float tensor pixel* dengan nilai dalam rentang [0-255]. Dalam *EfficientNetB4* terdiri dari beberapa argumen diantaranya:

a. *Include_top*

Dalam *include_top* menyertakan lapisan yang sepenuhnya terhubung di bagian atas jaringan. *Default to True*

b. *Weights*

Salah satu dari tidak adanya inisialisasi acak '*imagenet*' (*pre-training* di *imagenet*), atau jalur ke bobot file yang akan dimuat. *Default to 'imagenet'*.

c. *Input_tensor*

Tensor keras opsional yakni *output* dari *layers.input()* digunakan untuk sebagian *input* citra untuk model

d. *Input_shape*

Tuple berbentuk opsional hanya ditentukan jika *include_top* adalah *false*. Dalam hal ini harus memiliki tepat 3 *input channels*.

e. *Pooling*

Mode penyatuan opsional untuk ekstraksi fitur ketika *include_top* adalah *false*. *Default to None*. *None* berarti keluaran model akan menjadi keluaran tensor 4D dari lapisan konvolusi terakhir. *-avg* berarti penyatuan rata-rata global akan diterapkan ke lapisan keluaran konvolusi terakhir, dengan demikian keluaran model akan menjadi tensor 2D. *-max* berarti penggabungan maks global akan diterapkan.

f. *Classes*

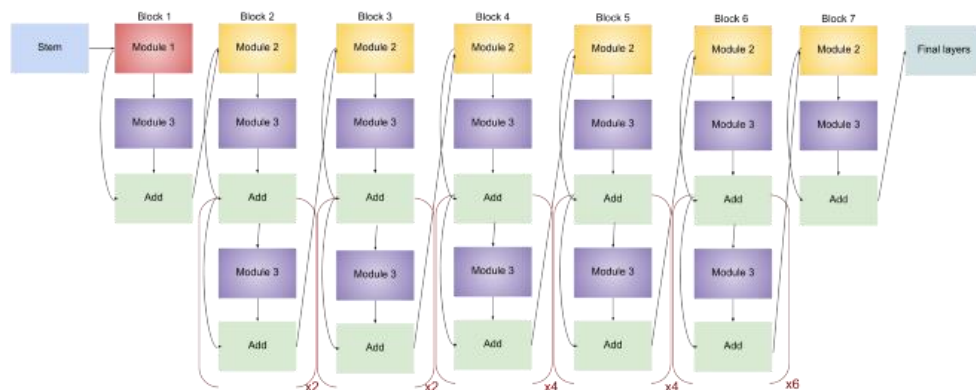
Jumlah kelas opsional digunakan untuk mengklasifikasikan citra, hanya untuk ditentukan jika *include_top* adalah *True*. Dan jika tidak ada argument bobot yang ditentukan. *Default to 1000* (jumlah kelas dalam *ImageNet*).

g. *Classifier_activation*

Fungsi aktivasi digunakan pada lapisan atas. Kecuali *include_top=True*. Set *classifier_activation=None* untuk mengembalikan logit dari lapisan atas.

Default to 'softmax', saat memuat bobot yang telah dilatih sebelumnya, *classifier* hanya dapat berupa *None* atau *'softmax'*.

Berikut arsitektur model *EfficientNetB4*:



Sumber: [47]

Gambar 2.11 Arsitektur *EfficientNetB4*

Keterangan:

Steam : Gambar 2.1

Final layers : Gambar 2.2

Module1 : Gambar 2.3

Module 2 : Gambar 2.4

Module 3 : Gambar 2.5

2.1.17. Evaluasi Kinerja Model

Evaluasi kinerja model tidak berhenti pada saat proses *training* telah mencapai nilai optimal, namun harus dilakukan juga saat mengevaluasi kinerja model pada *validation*. Terakhir juga harus dilakukan evaluasi prediksi model dengan mengukur *performance measure* pada data *test* [35].

Untuk melakukan evaluasi pada suatu model klasifikasi diperlukan himpunan data uji yang tidak digunakan dalam proses pelatihan. Evaluasi dapat dilakukan menggunakan suatu ukuran tertentu yang diilustrasikan pada penjelasan berikut [32]:

- a. *TP* adalah *true positives*, jumlah *tuple* positif yang dilabeli dengan benar oleh model klasifikasi.
- b. *TN* adalah *true negatives*, jumlah *tuple* negatif yang dilabeli dengan benar oleh model klasifikasi
- c. *FP* adalah *false positives*, jumlah *tuple* negatif yang salah dilabeli oleh model klasifikasi
- d. *FN* adalah *false negative*, jumlah *tuple* positif yang salah dilabeli oleh model klasifikasi.

Dalam pengukuran kinerja model yang telah dibuat dalam beberapa metode evaluasi model, diantaranya:

1. *Confusion Matrix*

Confusion matrix merupakan metode yang digunakan untuk mengukur kinerja dari suatu metode klasifikasi (*performance of a classification model or classifier*). Metode ini pada dasarnya mengandung informasi yang

membandingkan hasil klasifikasi yang telah dilakukan oleh sistem dengan hasil klasifikasi yang sebenarnya [43].

Confusion matrix dapat dinyatakan melalui *matrix* berikut:

		<i>Actual Class</i>	
<i>Predicted Class</i>	<i>Positive (1)</i>	TP	FP
	<i>Negative (0)</i>	FN	TN

Sumber : [43]

Gambar 2.12 *Confusion Matrix*

Pada Gambar 2.12 dapat dilihat *actual class* (sebagai kolom) diletakkan di atas dan *predicted class* diletakkan di samping (sebagai baris). Pada contoh di atas, ada dua *class* yang dicantumkan pada *confusion matrix*, yaitu *class positive* atau *yes* atau 1 dan *class negative* atau *no* atau 0.

Confusion Matrix ini berfungsi untuk menganalisis kualitas model klasifikasi sehingga dapat mengenali tuple-tuple dari kelas yang ada. Dimana *TP* dan *TN* menyatakan bahwa model klasifikasi dapat mengenali tuple dengan benar, sebaliknya *FP* dan *FN* menyatakan bahwa model klasifikasi salah dalam mengenali *tuple* [32].

2. *Accuracy*

Akurasi merupakan *performance measure* yang paling sederhana dan digunakan untuk mengevaluasi kinerja model. Akurasi ini juga didefinisikan sebagai proporsi prediksi yang benar dengan banyaknya sampel [35].

Accuracy menggambarkan keakuratan model dalam mengklasifikasikan. Dengan kata lain *accuracy* merupakan tingkat kedekatan nilai prediksi dengan nilai aktual (sebenarnya).

Persamaan nilai *accuracy*:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots(2.4)$$

Sumber : [43]

3. *Precision (Positive Prediction Value)*

Precision menggambarkan hasil prediksi yang dihasilkan oleh model berdasarkan tingkat keakuratan antara data yang diminta [49].

Persamaan nilai *precision*:

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots(2.5)$$

Sumber : [45]

4. *Recall (True Positive Rate)*

Recall menunjukkan keberhasilan suatu model dalam menemukan kembali sebuah informasi. *Recall* juga dapat disebut rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif [50].

Persamaan nilai *recall*:

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots(2.6)$$

Sumber : [45]

5. *F1-Score*

Nilai terbaik *F1-Score* adalah 1.0 dan nilai terburuknya adalah 0. Secara representative, jika *F1-Score* memiliki skor baik mengindikasikan bahwa model klasifikasi mempunyai *precision* dan *recall* yang baik pula.

Persamaan nilai *F1-Score*:

$$\frac{1}{F1} = \frac{1}{2} \left(\frac{1}{precision} + \frac{1}{recall} \right) \dots\dots\dots(2.7)$$

Sumber: [51]

6. Diagnosa Perilaku Model

Ada tiga dinamika kurva pembelajaran yang digunakan untuk mendiagnosis perilaku model [52]:

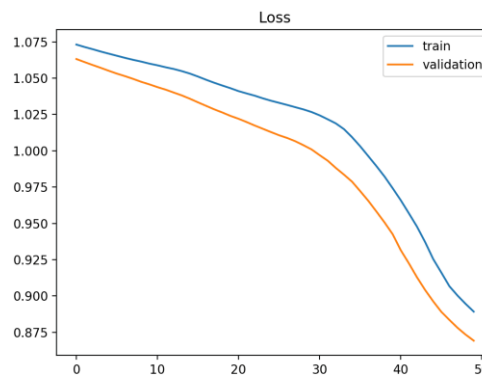
a. *Underfit*

Model *underfit* dapat diidentifikasi dari kurva belajar hanya *training loss* saja. *Underfit* ini menunjukkan garis datar dari *loss* yang relative tinggi, sehingga menunjukkan bahwa model tidak dapat mempelajari data *train* sama sekali.

Plot kurva menunjukkan *underfitting* jika:

- *Training loss* tetap datar walaupun telah melalui proses *training*
- *Training loss* terus menurun hingga akhir *training*.

Gambar 2.13 merupakan contoh kurva pembelajaran pelatihan menampilkan model *underfit* yang membutuhkan pelatihan lebih lanjut



Sumber : [52]

Gambar 2.13 Kurva *Underfit*

b. *Overfit*

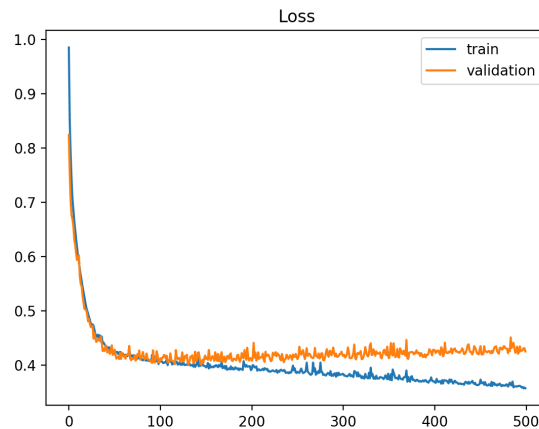
Masalah pada *overfitting* adalah bahwa semakin terspesialisasi model pada data *training*, semakin kurang baik untuk bisa digeneralisasikan ke data baru.

Plot kurva pembelajaran menunjukkan *overfitting* apabila:

- Plot *training loss* terus berkurang dengan pengalaman

- Plot *loss* validasi berkurang ke suatu titik dan mulai meningkat lagi

Gambar 2.14 menunjukkan contoh kurva pembelajaran dengan menampilkan model *overfit* pada *train* dan *validation*



Sumber : [52]

Gambar 2.14 Kurva Overfit

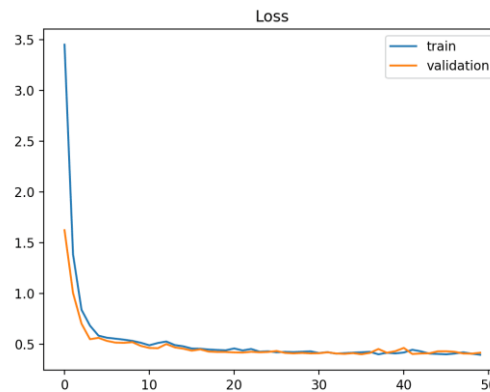
c. *Good Fit*

Good fit diidentifikasi oleh *loss training* dan *validation* yang menurun ke titik stabilitas dengan kesenjangan minimal antara dua nilai *loss* akhir.

Plot kurva pembelajaran menunjukkan *good fit* apabila:

- Plot *train loss* menurun ke titik stabilitas
- Plot *validation loss* berkurang ke titik stabilitas dan memiliki sedikit celah dengan *train loss*.

Pada Gambar 2.15 menampilkan contoh kurva pembelajaran *train* dan *validation* yang menunjukkan kecocokan.



Sumber : [52]

Gambar 2.15 Kurva Good Fit

2.2. Tinjauan Studi

Beberapa penelitian yang terkait dengan klasifikasi dan deteksi penyakit padi, diantaranya:

1. Berdasarkan penelitian yang dilakukan oleh Khirade dan Patil [19] dalam penelitiannya mensegmentasi bagian penyakit tanaman dan juga melakukan ekstraksi fitur dan teknik klasifikasi untuk mengekstrak ciri daun yang terinfeksi dan untuk mengklasifikasikan penyakit menggunakan metode ANN.
2. Identifikasi penyakit juga dilakukan oleh Suresha dkk [8] yang mengusulkan metode untuk mengidentifikasi penyakit *Blast* dan *Brown Spot* menggunakan metode KNN.
3. Penelitian lainnya terkait deteksi penyakit padi oleh Prajapati dkk [14] dimana dalam penelitian ini menyajikan sistem *prototype* untuk mendeteksi dan mengklasifikasikan penyakit padi yang terinfeksi, kemudian menggunakan metode *k-means* berbasis *centroid feeding* untuk segmentasi bagian penyakit, pada penelitian ini juga mengklasifikasikan kelas jamak menggunakan SVM.
4. Kodama dan Hata [16] juga melakukan pengembangan sistem untuk mengklasifikasikan tanaman padi yang sehat dengan yang sakit dengan mengolah citra padi, dalam pengklasifikasiannya menggunakan metode SVM.

5. Penelitian lainnya melakukan eksplorasi metode *deep learning CaffeNet* untuk mengklasifikasikan hama penyakit padi. Dimana model yang dikembangkan mampu mengklasifikasikan 13 kelas penyakit padi [6].
6. Archana dan Sahayadhas [18] juga melakukan penelitian untuk memenuhi tantangan dalam memprediksi penyakit padi yang difokuskan pada metode deteksi otomatis dengan segmentasi citra dan metode deteksi otomatis untuk mengidentifikasi penyakit tertentu.
7. Penelitian yang dilakukan oleh Daniya dan Vigneshwari [15] dengan mendeteksi citra padi yang terinfeksi dengan menggunakan *machine learning* dengan algoritma *probabilistic neural network (PNN)*, *genetic algorithm (GA)*, dan *SVM* yang dipilih sebagai teknik klasifikasi.
8. Kawcher Ahmed dkk [9], melakukan penelitian dengan referensi mendeteksi penyakit tanaman padi menggunakan teknik *machine learning* dengan menggunakan algoritma *Logistic Regression*, *K-NN*, *Decision Tree* dan *Naïve Bayes*, sehingga memperoleh akurasi terkecil dihasilkan oleh algoritma *Naïve Bayes* dengan akurasi sebesar 50%.
9. Klasifikasi citra penyakit padi juga dilakukan oleh Saputra dkk [7] dengan mengusulkan metode *GLCM* sebagai ekstraksi fitur untuk analisis teks dengan nilai fitur kontras, energy, entropi, homogenitas dan korelasi. Kemudian klasifikasi penyakit menggunakan algoritma *KNN* sehingga menghasilkan akurasi nilai maksimum sebesar 65,83%.
10. Penelitian dengan klasifikasi menggunakan *DCNN* untuk mendiagnosis kekurangan gizi beras melalui citra daun padi, namun dalam mengidentifikasi sedikit mengalami ketidak akuratan, sehingga peneliti harus mengeksplorasi keakuratan *DCNN* [10].
11. Penelitian lain Chowdhury dkk [13] mendeteksi penyakit dan hama citra padi menggunakan arsitektur berskala besar yakni *VGG16* dan *InceptionV3* untuk mengenali penyakit padi, namun dalam penelitian ini pada arsitektur yang digunakan berskala besar sehingga tidak cocok untuk perangkat seluler.

12. Sethy, et al [11] melakukan pre-processing, segmentasi, ekstraksi fitur, hingga klasifikasi untuk mendiagnosa penyakit padi menggunakan arsitektur *CNN*.
13. Begitu pula dengan Goluguri dkk [12] Mengidentifikasi penyakit padi dengan teknologi terbaru menggunakan metode *deep learning*, *SVM*, *Multiclass SVM*, *CNN*, *Back propagation neural network*.
14. Deteksi penyakit padi juga dilakukan dengan memanfaatkan segmentasi *LBP*, dan *HOG* kemudian diklasifikasikan dengan algoritma *SVM* [17].

Berdasarkan penelitian terdahulu yang terkait dengan obyek penelitian yang dilakukan, maka berbagai penelitian terkait penyakit padi seperti pada tabel *literature review* berikut:

Tabel 2.3 Literature Review

No	Penulis	Judul	Tipe Data	Metode	Deskripsi	Hasil
1	Ahmad Arib Alfarisy et al [6]	<i>Deep Learning Based Classification for Paddy Pests & Diseases Recognition</i>	<i>Paddy diseases (Public)</i>	<i>CaffeNet</i>	Penelitian ini mengeksplorasi metode <i>deep learning CaffeNet</i> untuk mengklasifikasikan hama dan penyakit padi. Model yang dikembangkan mampu mengklasifikasikan 13 kelas penyakit padi	Hasil perbandingan dengan metode lain yang menggunakan <i>deep learning</i> pada tanaman selain padi hasilnya masih kurang
2	Harshadkumar B. Prajapati et al [14]	<i>Detection and classification of rice plant diseases</i>	<i>Rice Plant (Private)</i>	<i>K-Means, SVM</i>	Penelitian ini menyajikan sistem prototype untuk mendeteksi dan mengklasifikasi penyakit padi yang terinfeksi, kemudian menggunakan metode <i>K-Means</i> berbasis <i>centroid feeding</i> untuk segmentasi bagian penyakit. Serta menggunakan metode <i>SVM</i> untuk klasifikasi kelas jamak.	Akurasi pada data uji rendah dan lebih spesifik akurasi pada citra Leaf Smut menjadi lebih rendah.
3	Daniya dan Vigneshwari [15]	<i>A Review on Machine Learning Techniques for Rice Plant Disease Detection in</i>	<i>Rice plant (Private)</i>	<i>Klasifikasi Neuro Fuzzy expert system, PCA based</i>	Deteksi citra padi terinfeksi dengan menggunakan <i>Machine Learning</i> , algoritma <i>Probabilistic Neural Network (PNN)</i> , <i>Genetic Algorithms</i>	

No	Penulis	Judul	Tipe Data	Metode	Deskripsi	Hasil
		<i>Agricultural Research</i>		<i>SVM, Deep CNN, faseter R-CNN, SVM</i>	(GA), dan SVM dipilih sebagai teknik klasifikasi.	
4	Archana dan Sahayadhas [18]	<i>Automatic Rice Leaf Disease Segmentation Using Image Processing Techniques</i>	<i>Rice leaves (Private)</i>	<i>K-Means</i>	Penelitian ini untuk memenuhi tantangan memprediksi penyakit padi yang difokuskan pada metode deteksi otomatis segmentasi gambar dan metode deteksi otomatis untuk mengidentifikasi penyakit tertentu.	Segmentasi tidak dapat memenuhi identifikasi penyakit padi secara otomatis
5	Kodama, Takuya dan Yutaka Hata [16]	<i>Development of Classification System of Rice Disease Using Artificial Intelligence</i>	<i>Image of rice (Private)</i>	<i>SVM, PCA, LOOCV, AI</i>	Pengembangan sistem untuk mengklasifikasikan tanaman padi yang sehat dan sakit dengan mengolah citra padi, dalam pengklasifikasiannya menggunakan SVM sehingga memperoleh akurasi 90%.	Hasil klasifikasi menunjuka akurasi yang cukup besar namun hasilnya perlu konfirmasi apakah perbedaan bagian yang sakit ditangkap secara akurat sebagai fitur.
6	Khirade Sachin D dan Patil [19]	<i>Plan Disease Detection Using Image Processing</i>	<i>Plant disease (Private)</i>	<i>SVM, ANN</i>	penelitian ini untuk mensegmentasi bagian penyakit tanaman, dan juga melakukan ekstraksi fitur dan teknik klasifikasi untuk mengekstrak ciri daun yang terinfeksi, dan untuk mengklasifikasikan penyakit menggunakan metode ANN.	
7	Suresha M and Shreekanth K N dan Thirumalesh B V [8]	<i>Recognition of Diseases in Paddy Leaves Using kNN Classifier</i>	<i>Paddy Leaves (private)</i>	<i>KNN</i>	Penelitian ini mengusulkan metode untuk mengidentifikasi penyakit Blast dan Brown Spot menggunakan metode KNN	Hasil akurasi masih rendah yakni sebesar 76,59%

No	Penulis	Judul	Tipe Data	Metode	Deskripsi	Hasil
8	Saputra et al [7]	<i>Rice Leaf Disease Image Classifications Using KNN Based On GLCM Feature Extraction</i>	<i>Rice Leaf Diseases (Private)</i>	<i>GLCM, KNN</i>	Klasifikasi citra penyakit padi dengan mengusulkan metode <i>GLCM</i> sebagai ekstraksi fitur untuk analisis teks dengan nilai fitur kontras, energi, entropi, homogenitas dan korelasi. Kemudian klasifikasi penyakit menggunakan algoritma <i>KNN</i>	Hasil akurasi nilai maksimum yakni 65,83%
9	Kawcher Ahmed et al [9]	<i>Rice Leaf Disease Detection Using Machine Learning Techniques</i>	tidak disebutkan	<i>Logistic Regression, KNN (K1 & 3), Decision tree, Naïve bayes</i>	Sedangkan pada penelitian selanjutnya dengan referensi mendeteksi penyakit tanaman padi menggunakan teknik <i>machine learning</i> dengan menggunakan algoritma <i>Logistic Regression, K-NN, Decision tree</i> dan <i>Naïve Bayes</i>	Hasil penelitian memperoleh akurasi terkecil dihasilkan oleh algoritma <i>naïve bayes</i> dengan akurasi sebesar 50%.
10	Zhe Xu et al [10]	<i>Using Deep Convolutional Neural Networks for Image-Based Diagnosis of Nutrient Deficiencies in Rice</i>	Gejala kekurangan unsur hara (Private)	<i>DCNNs & DenseNet1 21</i>	Penelitian ini melakukan klasifikasi dengan menggunakan <i>DCNN</i> untuk mendiagnosis kekurangan gizi beras melalui citra daun padi	Penggunaan metode <i>DCNN</i> untuk mengidentifikasi sedikit mengalami ketidakakuratan, sehingga peneliti harus mengeksplorasi keakuratan <i>DCNN</i>
11	Chowdhury R. Rahman et al [13]	<i>Identification and recognition of rice diseases and pests using convolutional neural networks</i>	<i>Rice Plant (Private)</i>	<i>VGG16, InceptionV3, MobileNet v2, NasNet Mobile, SqueezeNet, Simple CNN</i>	Penelitian ini mendeteksi penyakit dan hama citra padi menggunakan arsitektur berskala besar yakni <i>VGG16</i> dan <i>InceptionV3</i> untuk mengenali penyakit padi	Karena arsitektur yang digunakan berskala besar sehingga tidak cocok untuk perangkat seluler

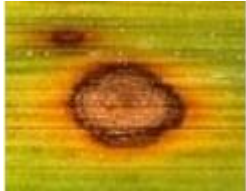
No	Penulis	Judul	Tipe Data	Metode	Deskripsi	Hasil
12	Sethy, et al [11]	<i>Image Processing Techniques for Diagnosing Rice Plant Disease: A Survey</i>	<i>Rice Disease (Private)</i>	<i>SVM</i>	Penelitian ini melakukan <i>pre-processing</i> , segmentasi, ekstraksi fitur, hingga klasifikasi untuk mendiagnosa penyakit padi menggunakan arsitektur <i>CNN</i>	
13	Goluguri et al [12]	<i>Image classifiers and image deep learning classifiers evolved in detection of Oryza sativa diseases: survey</i>	<i>Oryza sativa fungal pathogens (public)</i>	<i>SVM, multiclass SVM, CNN, Back propagation on neural network</i>	Mengidentifikasi penyakit padi dengan teknologi terbaru menggunakan <i>metode deep learning, SVM, Multiclass SVM, CNN, Back propagation neural network</i>	
14	Pothen dan Maya [17]	<i>Detection of Rice Leaf Diseases Using Image Processing</i>	<i>Rice disease (public)</i> Terdiri dari 120 citra dengan masing-masing kelas 40 citra (<i>Bacterial leaf blight, brown spot, dan leaf smut</i>)	<i>LBP, HOG dan SVM</i>	Deteksi penyakit padi menggunakan segmentasi <i>LBP</i> , dan <i>HOG</i> kemudian diklasifikasikan dengan algoritma <i>SVM</i>	Akurasi yang dihasilkan dari algoritma <i>SVM</i> adalah 94,6%
15	Mingxing Tan dan Quoc V.Le [53]	<i>EfficientNet: Rethinking Model Scaling for Convolutional Neural Network</i>	<i>Cifar, Flowers, Birdsnap, Stanford Cars, FGVC Aircraft, Oxford-IIIT Pets, Food-101 (Public)</i>	<i>EfficientNet, MobileNet dan ResNet</i>	Dalam penelitian ini dilakukan penskalaan model dan identifikasi terhadap <i>network depth, width, dan resolution</i> yang menghasilkan kinerja yang lebih baik	

2.3. Tinjauan Organisasi/Obyek Penelitian

Objek penelitian yang penulis angkat pada penelitian ini adalah mengenai penyakit pada tanaman padi dengan berbagai kategori atau kelas penyakit seperti pada riset jenis dataset penyakit padi pada Tabel 2.4 berikut:

Tabel 2.4 Riset Dataset Penyakit Padi (2016-2020)

No	Tahun	Peneliti	Tippe Dataset	Jenis Penyakit	Sampel Citra
1	2016	Jitesh P. Shah etc [54]	145 citra: 30 <i>helathy</i> , 46 <i>bacterial leaf blight</i> , 44 <i>leaf smut</i> dan 25 <i>brown spot</i> (private)	<i>Brown Spot</i> , <i>Leaf Blast</i> , <i>Narrow Brown Spot</i> , <i>Leaf Scald</i> , <i>Sheath Blight</i> , <i>Bacterial leaf Blight</i>	 (Brown Spot)
2	2017	Harshadkumar B. Prajapati etc [14]	120 citra: 40 citra dari setiap penyakit (private)	<i>Bacterical Leaf Blight</i> , <i>Brown Spot</i> , <i>Leaf Smut</i>	 (Brown Spot)
3		Yang Lu, [55]	500 citra (private)	<i>Rice blast(RB)</i> , <i>rice false smut (RFS)</i> , <i>rice brown spot (RBS)</i> , <i>rice bakanae disease (RBD)</i> , <i>rice sheath blight (RSHB)</i> , <i>rice sheath rot (RSR)</i> , <i>rice bacterial leaf blight (RBLB)</i> , <i>rice bacterial sheath rot (RBSR)</i> , <i>rice seeding blight (RSEB)</i> dan <i>rice bacterial wilt (RBW)</i>	

No	Tahun	Peneliti	Tipe Dataset	Jenis Penyakit	Sampel Citra
4	2018	K.S. Archana dan Arun Sahayadhas [18]	Jumlah citra tidak disebutkan (private)	<i>Brown Spot, Bacterial Leaf Blight</i>	 (Bacterial Leaf Blight)
5	2019	Wan-jie Liang [56]	5080 citra: 2906 positif dan 2902 negatif (private)	Bias positif dan negatif	
6		Md. Jahid Hasan [57]	1080 citra (public)	<i>Brown Spot, False Smut, Bacterial Leaf Blight, Leaf Scald, Leaf Smut, Red Stripe, Sheath Blight, Tungro Virus, Blast</i>	 (brown Spot)
7		V. K. Shrivastava [58]	619 citra (private)	<i>Rice Blast (RB); (ii) Bacterial Leaf Blight (BLB); (ii) Sheath Blight (SB); and (iv) Healthy Leaves (HL)</i>	 (Rice Blast)
8	2020	Chowdhury R. Rahman etc [13]	1426 citra (private)	<i>False smut, Brown Plant Hopper (BPH), Bacterial Leaf Blight (BLB), Neck Blast, Stem borer, Hispa, Sheath Blight, Brown Spot, Others</i>	 (Bacterial Leaf Blight)

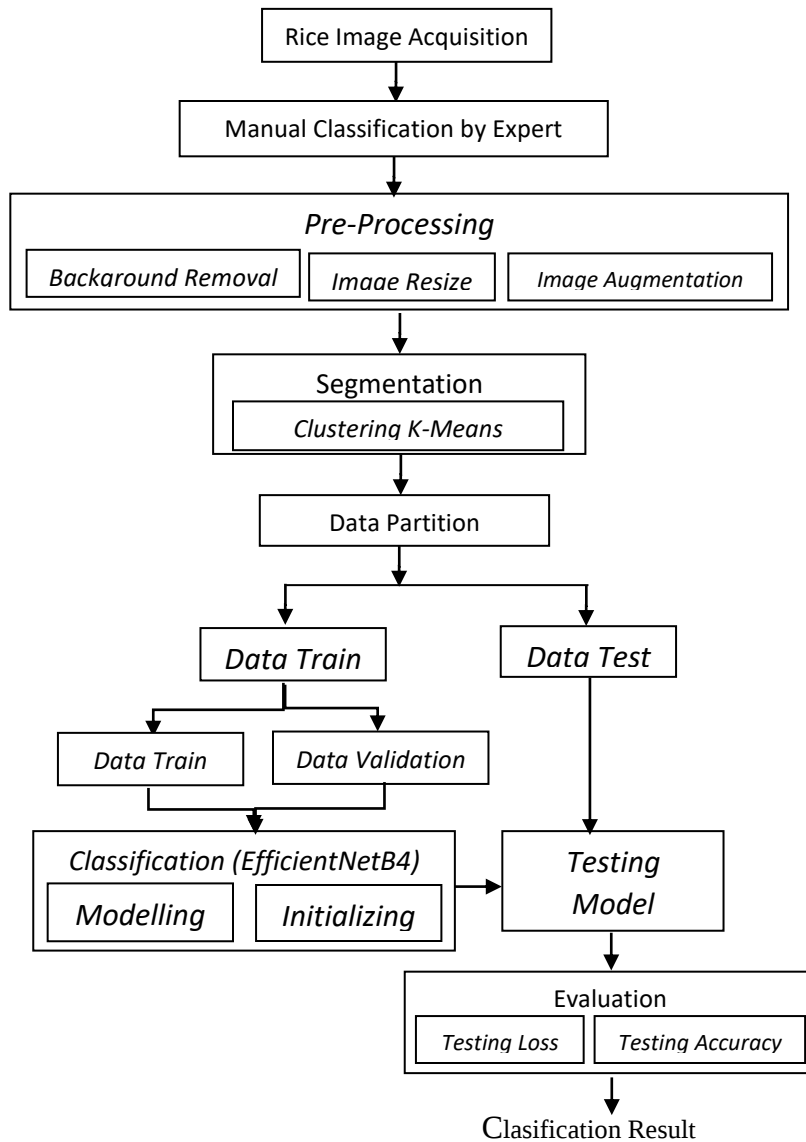
No	Tahun	Peneliti	Tipe Dataset	Jenis Penyakit	Sampel Citra
9		Feng Jiang etc [59]	8911 citra (private)	Healthy, blast, bacterial spot, streak leaf spot, dan sheath blight	 (Blast)
10		Junde Chen [60]	500 citra (private)	Leaf smut, leaf scald, stackburn, white tip dan bacterial leaf streak	 (Leaf Smut)
11		Dengshan Li [61]	5320 citra (private)	Rice sheath blight, rice stem borer symptoms, rice brown spot	 (Brown Spot)
12		N. V. Raja Reddy Goluguri [62]	1600 citra (public)	Leaf Blast, Brown spot, Hispa dan Healthy	

BAB III

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Dalam melakukan sebuah penelitian penulis menggunakan rancangan yang menggambarkan prosedur atau langkah-langkah yang harus ditempuh selama penelitian berlangsung. Tahapan penelitian yang dilakukan penulis meliputi tahapan berikut ini:



Sumber: (Hasil Penelitian, 2021)

Gambar 3.1 Metode Penelitian

Berdasarkan Gambar 3.1 penulis menyusun metode penelitian yang diusulkan dengan penjelasan sebagai berikut:

1. Akuisisi Citra Padi

Pada proses akuisisi citra padi penulis melakukan pengambilan citra penyakit padi sesuai dengan fokus permasalahan yang akan diteliti. Proses pengambilan citra sampel padi dilakukan langsung di sawah dari 3 kabupaten di provinsi yang berbeda yakni di Kabupaten Klaten Jawa Tengah, Kabupaten Gunung Kidul DIY, dan Kabupaten Sukabumi Jawa Barat. Pada proses pengambilan citra ini penulis menggunakan bantuan alat berupa kamera *handphone* Samsung A10.

2. Klasifikasi Manual oleh Ahli Tanaman

Pada proses klasifikasi manual oleh ahli dilakukan dengan menyebar kuesioner ke 9 orang ahli tanaman dengan bantuan *Google Formulir*. Kuesioner yang disebar oleh penulis terdiri dari 194 pertanyaan yang berisi foto daun padi beserta pertanyaannya yang terlampir dalam lampiran 1.

Tak hanya itu penulis juga melakukan pencocokan hasil klasifikasi dari ahli dengan buku saku hama dan penyakit tanaman padi dari balai pengkajian teknologi pertanian kepulauan Riau serta melakukan konfirmasi langsung ke beberapa ahli untuk memastikan bahwa hasil klasifikasi telah benar.

3. *Pre-Processing*

Pada tahap *pre-processing* penulis melakukan *background removal*, *image resize* dan *image augmentation*.

4. Segmentasi

Pada proses segmentasi dilakukan dengan metode klustering *K-Means Classifier* dengan tujuan untuk membagi objek agar fokus sehingga dapat terlihat perbedaan antara bercak pada daun padi. Dalam proses klustering ini penulis membagi menjadi 3 segmen berdasarkan warna yakni warna hijau, coklat dan hitam.

5. Partisi Data

Pada tahap ini penulis melakukan pembagian atau *split* data menjadi data *train* dan data *test*. Kemudian data *train* tersebut dibagi kembali menjadi data *validation*.

6. Klasifikasi

Pada tahap klasifikasi penyakit padi dilakukan dengan metode *CNN* untuk mengklasifikasikan penyakit padi berdasarkan klasifikasi manual kategori penyakit padi. Pada tahap ini penulis menggunakan arsitektur *EfficientNetB4*. Seluruh proses klasifikasi penyakit padi dilakukan dengan menggunakan bantuan *google colabatory*. Pada tahap klasifikasi ini penulis melakukan *modelling* dan *initializing* untuk membangun model arsitektur yang akan digunakan dalam mengklasifikasikan penyakit padi.

7. Evaluasi

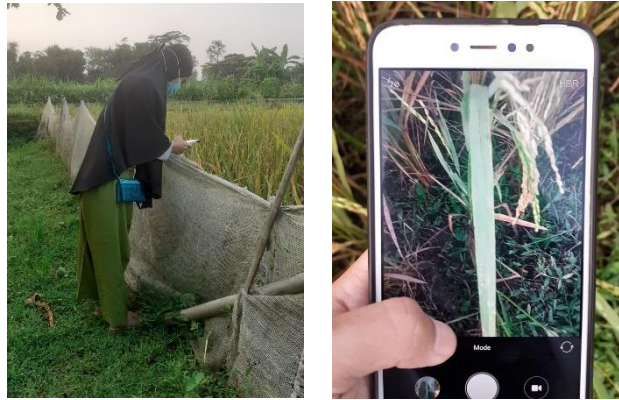
Pada tahap ini penulis melakukan evaluasi terhadap model yang telah dibangun, evaluasi disini penulis menghasilkan hasil *testing loss* dan *testing accuracy*.

8. Hasil Klasifikasi

Pada tahap ini penulis melakukan visualisasi hasil klasifikasi dalam bentuk *confusion matrix*, *classification report* dan grafik.

3.2. Proses Akuisisi Citra

Pada proses akuisisi citra padi penulis mengambil citra penyakit padi dengan menggunakan kamera *handphone* merek Samsung dengan tipe A10. Pengambilan citra difokuskan pada daun padi yang mempunyai bercak penyakit. Pada proses pengambilan citra dilakukan dengan jarak ± 10 cm antara kamera dan objek yang diambil dengan pencahayaan matahari sekitar pukul 10.00-12.00 Waktu Indonesia Barat (WIB). Proses akuisisi citra yang penulis lakukan tampak seperti pada Gambar 3.2 berikut:



Sumber: (Hasil Penelitian, 2021)

Gambar 3.2 Pengambilan Citra

3.3. *Tools* Pendukung

Dalam penelitian diperlukan beberapa *tools* yang mendukung selama penelitian berlangsung. Beberapa *tools* yang dibutuhkan oleh penulis dalam penelitian ini meliputi *hardware* dan *software* yang tersaji dalam Tabel 3.1 dan Tabel 3.2 berikut:

Tabel 3.1 Spesifikasi *Hardware*

Perangkat	Spesifikasi
Laptop	Asus Laptop A409J
<i>Handphone</i>	Samsung A10 RAM 2 GB Memory 32 GB

Sumber: (Hasil Penelitian, 2021)

Tabel 3.2 Spesifikasi *Software*

<i>Software</i>	Spesifikasi
Sistem Operasi	<i>Windows 10</i>
Pengolah Kata	<i>Microsoft Word 2010</i>
<i>Hyper Background Eraser</i>	https://www.remove.bg/
<i>Code Editting</i>	<i>Google Colaboratory</i>

Sumber: (Hasil Penelitian, 2021)

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

Pada bab ini akan dijelaskan mengenai pembahasan yang terdiri dari akuisisi citra padi, klasifikasi manual oleh ahli, *pre-processing* yang meliputi *background removal*, *resize image*, *image augmentation*, segmentasi, klasifikasi, dan evaluasi sebagai berikut:

4.1. Akuisisi Citra Padi

Tahap akuisisi citra ini penulis lakukan dengan tujuan memperoleh citra penyakit padi yang akan digunakan dalam penelitian, sehingga melalui tahap penangkapan atau pemindaian citra ini diperoleh citra berupa daun padi yang berpenyakit. Namun citra asli hasil akuisisi pada citra masih banyak *noise* pada *background* citra dikarenakan pengambilan citra dilakukan langsung di sawah. Hasil dari proses akuisisi memperoleh citra sebanyak 194 citra dengan ukuran 774 x 1032 pixel. Untuk hasil akuisisi citra penyakit padi selengkapnya terlampir pada halaman lampiran 4. Berdasarkan hasil akuisisi citra yang dilakukan, penulis memperoleh 7 jenis penyakit dan hama pada tanaman padi seperti tampak pada sampel hasil akuisisi citra Tabel 4.1 di bawah ini:

Tabel 4.1 Sampel Hasil Akuisisi Citra

No	Kelas	Citra Hasil Akuisisi	Keterangan
1	Blas		Penyakit Padi

No	Kelas	Citra Hasil Akuisisi	Keterangan
2	<i>Brown Spot</i>		Penyakit Padi
3	Hawar		Penyakit Padi
4	Kresek		Penyakit Padi
5	<i>Narrow Brown Spot</i>		Penyakit Padi
6	Nekrosis		Hama Padi

No	Kelas	Citra Hasil Akuisisi	Keterangan
7	Putih Palsu		Hama Padi

Sumber : (Hasil Penelitian, 2021)

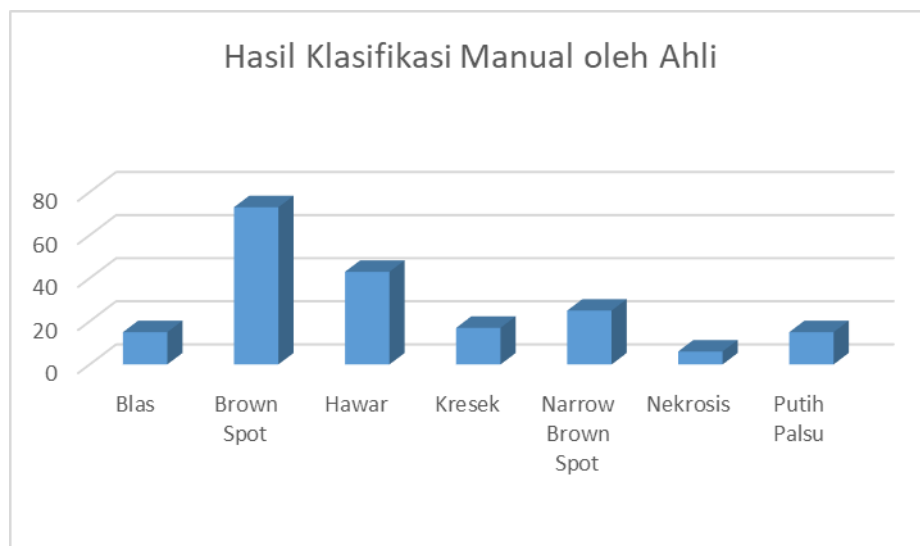
Pada Tabel 4.1 disajikan sampel citra hasil akuisisi yang terdiri dari penyakit dan hama pada tanaman padi. Pada citra pertama diperoleh penyakit Blas dimana ciri khusus dari penyakit ini adanya bercak berbentuk belah ketupat. Sedangkan pada citra kedua diperoleh penyakit *Brown Spot* dengan ciri khusus bercak coklat berbentuk oval dan bulat. Citra ketiga diperoleh jenis penyakit Hawar dengan ciri khusus tepi daun berwarna hijau kelabu, melipat, menggulung dan akhirnya mongering. Citra keempat diperoleh jenis penyakit Kresak memiliki gejala hamper sama dengan Hawar namun Kresak terjadi pada tanaman dini (vegetatif) sedangkan Hawar terjadi pada tanaman dewasa (generatif). Jenis penyakit kelima yang diperoleh dari hasil akuisisi yakni penyakit *Narrow Brown Spot* dengan gejala bercak sama dengan *Brown Spot* hanya saja bercaknya lebih sempit dan memanjang. Pada citra keenam diperoleh jenis hama pada padi yakni Nekrosis atau biasa disebut kerusakan mekanis yang disebabkan oleh hama seperti belalang. Sedangkan hama pada citra ketujuh yakni Putih Palsu menunjukkan gejala daun padi terlipat secara longitudinal dan larva ulat berada dalam lipatan tersebut. larva tersebut akhirnya akan mengikis jaringan hijau daun hingga menjadi berwarna putih dan mongering.

4.1. Klasifikasi Manual oleh Ahli

Tahap klasifikasi manual ini penulis lakukan dengan tujuan untuk memperoleh jenis penyakit dari citra daun padi yang sebelumnya telah diambil secara acak pada tahap sebelumnya, sehingga melalui tahap klasifikasi manual ini diperoleh jenis atau kategori dari penyakit padi yang sesuai dengan keilmuan

terkait yakni ilmu pertanian. Pada proses klasifikasi manual ini dilakukan oleh ahli tanaman dengan bantuan 9 orang ahli dari berbagai Universitas atau Instansi pertanian melalui media *Google Formulir*. Tahap klasifikasi manual ini penulis menyebar kuesioner melalui 3 tahapan yang disampaikan ke ahli melalui link *Google Formulir*. Dimana pada tahap pertama penulis menyebar link kuesioner http://bit.ly/Form_RiceDiseases, tahap kedua dengan link kuesioner http://bit.ly/ClaimPenyakitPadi_2 dan tahap pengiriman kembali hasil dugaan atau hasil klasifikasi manual dengan link http://bit.ly/KonfirmasiHasilDugaan_PenyakitPadi. Pada proses klasifikasi manual ini penulis melaksanakannya selama ± 2 bulan secara bertahap mulai dari bulan april hingga bulan mei 2021.

Selain melalui kuesioner penulis juga melakukan pencocokan hasil klasifikasi sementara dengan gejala penyakit yang ada pada panduan buku saku hama dan penyakit padi dari balai pengkajian teknologi pertanian Riau tahun 2018. Tahap terakhir dari klasifikasi manual oleh ahli, penulis melakukan konfirmasi langsung ke ahli terkait melalui media *chat* dan telepon aplikasi *WhatsApp*. Berdasarkan hasil klasifikasi manual dapat digambarkan dalam bentuk grafik hasil klasifikasi manual oleh ahli seperti Gambar 4.2 di bawah ini:



Sumber: (Hasil Penelitian, 2021)

Gambar 4.2 Hasil Klasifikasi Manual oleh Ahli

Berdasarkan hasil klasifikasi manual oleh ahli yang ditampilkan pada Gambar 4.2 yang diperoleh dari kuesioner, pencocokan buku saku hama dan penyakit tanaman serta konfirmasi langsung ke ahli, maka diperoleh hasil klasifikasi manual dengan total jumlah 194 citra dari 7 kelas, yang terdiri dari 15 citra dari kelas Blas, 73 citra kelas *Brown Spot*, 43 citra kelas Hawar, 17 citra kelas Kresek, 25 citra kelas *Narrow Brown Spot*, 6 citra kelas Nekrosis, dan 15 citra dari kelas Putih Palsu.

Dalam penelitian ini penulis hanya menggunakan 5 kelas dari 7 kelas hasil klasifikasi manual, dimana 5 kelas yang diambil merupakan kelas penyakit padi dan masing-masing kelas penyakit penulis hanya mengambil 15 citra dari masing-masing kelas penyakit yang dimaksudkan agar jumlah citra dari masing-masing kelas seragam, sehingga diambil dari jumlah citra terkecil yakni dari kelas Blas. Informasi lebih jelas mengenai hasil klasifikasi manual dapat dilihat pada Tabel 4.2 berikut:

Tabel 4.2 Hasil Klasifikasi Manual oleh Ahli

No	Hasil Klasifikasi	Jumlah Citra	Keterangan
1	Blas	15	Penyakit
2	<i>Brown Spot</i>	73	Penyakit
3	Hawar	43	Penyakit
4	Kresek	17	Penyakit
5	<i>Narrow Brown Spot</i>	25	Penyakit
6	Nekrosis	6	Hama
7	Putih Palsu	15	Hama
	Total Citra	194	

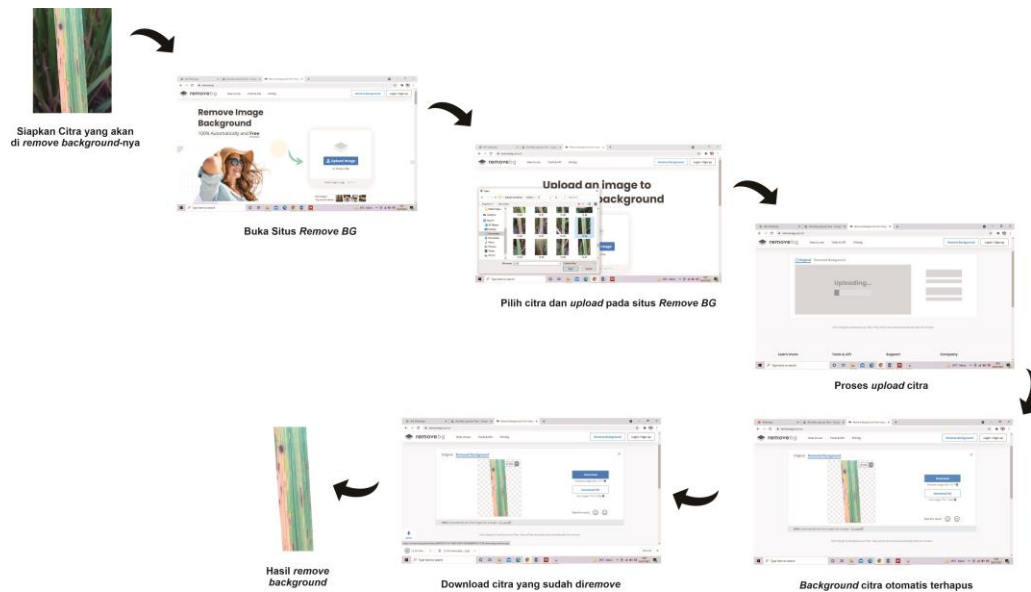
Sumber: (Penelitian, 2021)

4.2. Pre-Processing

Pada tahap *pre-processing* penulis melakukan beberapa tahapan berikut:

4.2.1. Background Removal

Tujuan dilakukan tahap ini adalah untuk memperoleh citra yang lebih jelas serta mengurangi derau atau *noise* pada citra, sehingga dampaknya proses membaca citra lebih mudah pada saat akan dilakukan pengolahan citra. Pada proses *remove background* penulis menggunakan bantuan alat yakni *website* <https://www.remove.bg/>. Pada proses *remove background* ini penulis melalui beberapa tahapan yang tercantum pada Gambar 4.3 berikut:



Sumber : (Penelitian, 2021)

Gambar 4.3 Tahapan Background Removal

Dari tahapan di atas penulis menyiapkan terlebih dahulu citra yang akan dihapus *background*-nya kemudian penulis membuka situs <https://www.remove.bg/>, langkah selanjutnya penulis upload citra yang sudah disiapkan. Maka situs akan memproses *upload* citra dan secara otomatis *background* pada citra yang di-*upload* terhapus. Selanjutnya penulis melakukan *download* citra yang telah berhasil di *remove bg*, maka citra hasil *remove* siap digunakan. Dari beberapa proses tersebut penulis menampilkan hasil *background removal* pada citra penyakit padi dengan sampel masing-masing kelas seperti pada Tabel 4.3 berikut:

Tabel 4.3 Hasil *Background Removal*

No	Kelas	Citra Asli	Hasil
1	Blas		
2	<i>Brown Spot</i>		
3	Hawar		
4	Kresek		
5	<i>Narrow Brown Spot</i>		

Sumber : (Penelitian, 2021)

4.2.2. *Resize image*

Tahap *resize image* ini penulis lakukan dengan tujuan untuk memperkecil piksel-piksel pada citra sehingga dampak dari tahapan ini dapat mempermudah sistem dalam membaca dan mengolah citra, selain itu penyimpanan yang dibutuhkan juga lebih kecil. Dalam tahap ini penulis mengubah berbagai ukuran citra menjadi citra berukuran 224 x 224 piksel dengan tujuan untuk mempermudah proses pengolahan citra.

4.2.3. *Image augmentation*

Proses *augmentation* dilakukan penulis karena suatu permasalahan yang dihadapi penulis yakni jumlah citra dalam data *train* masih minim sehingga mengakibatkan *overfitting*. Dampak dari proses ini penulis memperoleh citra yang lebih beragam dan bervariasi sehingga kemungkinan terjadinya *overfitting* dapat dicegah dalam proses *training*. Dari hasil dari proses *image augmentation* ini penulis memperoleh citra sebanyak 240 citra dari masing-masing kelas dengan citra awal sebanyak 15 citra. Berdasarkan hasil *augmentation* penulis memperoleh total 1200 citra serta 75 citra asli sehingga total dataset yang diolah yakni 1275 citra. Dalam proses *augmentation* ini penulis menggunakan *open source computer vision library (OpenCV)* dengan berbagai teknik di antaranya:

1. *Horizontal shift*

Penulis melakukan pergeseran atau translasi horizontal adalah menggeser citra ke kiri atau ke kanan berdasarkan rasio yang menentukan seberapa besar pergeseran maksimum. Dalam *augmentation horizontal shift* ini penulis menggunakan rasio sebesar 0.0 dan 0.2.

2. *Vertical shift*

Penulis menerapkan pergeseran citra ke atas atau bawah berdasarkan rasio pergeseran. Pada *augmentation vertical shift* ini menggunakan rasio 0.0 dan 0.1

3. *Brightness*

Pada teknik ini penulis menggunakan ruang warna *hue, saturation, value (HSV)*, sehingga semakin banyak nilai *saturation* dan *value matrix* semakin

besar tingkat kecerahannya. Dimana pada teknik ini penulis menggunakan nilai *saturation* dan *value matrix* sebesar (0.5, 3) dan (0.7, 5).

4. *Zoom*

Pada teknik ini penulis melakukan teknik *zoom* untuk memperbesar citra, dimana dalam teknik ini penulis menggunakan nilai *zoom* sebesar 0.5 yang berarti 50% dari keseluruhan citra dan nilai *zoom* sebesar 0.3 yang berarti 30% dari citra keseluruhan.

5. *Channel shift*

Teknik ini penulis melakukan pergeseran *channel* secara acak sengan sejumlah nilai dari rentang yang ditentukan, sehingga hasilnya seperti fungsi kecerahan. Pada teknik ini penulis menggunakan nilai *channel shift* sebesar 60 dan 40.

6. *Horizontal flip*

Pada teknik ini penulis membalik citra menjadi horizontal. Dimana pada teknik ini penulis menggunakan fungsi *OpenCV cv2.flip* dengan argumen bernilai 1 dan 0.

7. *Vertical flip*

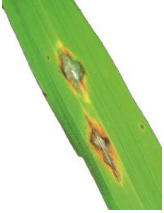
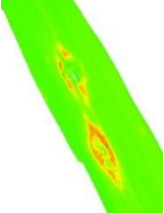
Sedangkan pada teknik *vertical flip* yang merupakan kebalikan dari *horizontal flip*, dimana penulis membalik citra menjadi vertical dengan fungsi *cv2.flip* bernilai 0 dan 1.

8. *Rotation*

Pada teknik ini penulis melakukan rotasi di *OpenCV* dengan membuat matriks rotasi dan menerapkannya pada transformasi affine. Pada teknik ini penulis menggunakan nilai rotasi sebesar 30 dan 50.

Hasil *image augmentation* dari 8 teknik di atas yang penulis lakukan dapat dilihat dapat Tabel 4.4 di bawah ini:

Tabel 4.4 Hasil *Image Augmentation*

No	Teknik Augmentation	Citra Asli (RGB)	Hasil Augmentation
1	Horizontal shift		
2	Vertical shift		
3	Brightness		
4	Zoom		
5	Channel shift		
6	Horizontal flip		
7	Vertical flip		

No	Teknik Augmentation	Citra Asli (RGB)	Hasil Augmentation
8	Rotation		

Sumber : (Hasil Penelitian, 2021)

4.2.4. Partisi Data

Tahap partisi data ini dilakukan penulis dengan tujuan untuk membagi dataset menjadi data *train*, *test*, dan *validation*, sehingga dampak dari pembagian data ini diperoleh data yang digunakan untuk melatih model serta data yang digunakan untuk mengetahui performa model yang sudah dilatih sebelumnya ketika menemukan data baru yang belum pernah dilihat sebelumnya.

Pada tahap ini penulis membagi dataset hasil *augmentation* dengan pembagian data *train* sebesar 90% dan data *test* sebesar 10%. Sehingga hasil pembagian dataset dari 1275 citra diperoleh pembagian citra sebanyak 1147 pada data *train* dan 128 data *test*. Sedangkan pada proses *fitting model* data *train* yang telah dipartisi diambil 10% untuk menjadi data *validation* sehingga diperoleh 114 data *validation*. Dimana data *validation* ini digunakan untuk mengoptimisasi saat melatih model sehingga model mampu mengenali pola secara generik

4.3. Metode Usulan

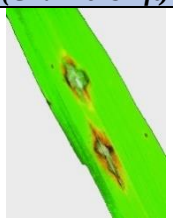
Berdasarkan permasalahan yang diangkat yakni mengenai penyakit padi, penulis mengusulkan metode *deep learning CNN* dalam pengklasifikasian penyakit padi untuk melakukan pendeteksian dini terhadap penyakit tersebut. dalam hal ini penulis menggunakan arsitektur *EfficientNet B4*. Berikut beberapa tahapan yang dilakukan beserta metode yang diusulkan pada penelitian ini:

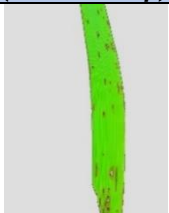
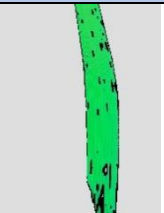
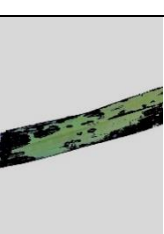
4.3.1. Segmentasi

Pada tahap ini penulis melakukan segmentasi dengan tujuan untuk memisahkan objek daun padi dan objek lainnya berupa bercak penyakit pada daun tersebut. Sehingga dampak yang dihasilkan dari segmentasi ini bercak pada daun padi yang mengindikasikan penyakit terlihat lebih jelas termasuk daun dengan bercak sempit seperti pada penyakit *narrow brown spot*. Meskipun pada kenyataannya ada beberapa citra yang tidak dapat tersegmentasi dengan baik dikarenakan bercaknya menyebar hampir diseluruh daun sehingga hasil segmentasi mengalami kebocoran selain itu *background* pada citra hasil *augmentation* khususnya pada teknik *rotation* yang berwarna hitam juga mempengaruhi hasil segmentasi. Hasil segmentasi selengkapnya penulis laporkan pada halaman lampiran 5 pada poin dataset hasil *augmentation* dan segmentasi.

Tahap segmentasi ini penulis menggunakan algoritma *K-Means clustering* dengan komponen citra *RGB* sebanyak 3 segmen warna sesuai dengan warna utama dari penyakit padi yang diyakini oleh ahli untuk menentukan jenis penyakit padi yakni warna hijau, coklat dan hitam. Hasil segmentasi yang penulis tampilkan merupakan kelanjutan dari proses sebelumnya yakni segmentasi dari hasil *augmentation* dengan teknik *channel shift* dari masing-masing kelas seperti pada Tabel 4.5 berikut:

Tabel 4.5 Hasil Segmentasi

No	Jenis	Citra Asli (RGB)	Hasil Augmentation (Channel shift)	Hasil Segmentasi K-Means
1	Blas			

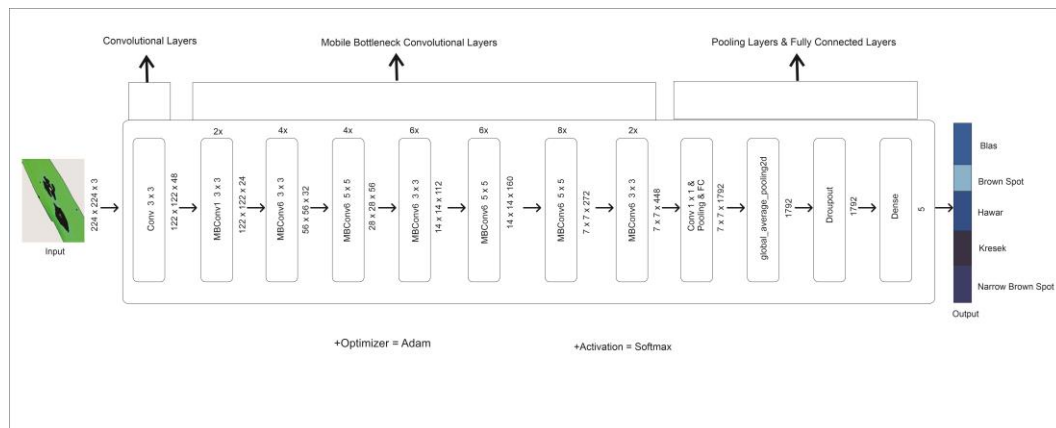
No	Jenis	Citra Asli (RGB)	Hasil Augmentation (Channel shift)	Hasil Segmentasi K-Means
2	Brown Spot			
3	Hawar			
4	Kresek			
5	Narrow Brown Spot			

Sumber : (Penelitian, 2021)

4.3.2. Klasifikasi

Tahap klasifikasi ini penulis lakukan dengan tujuan untuk dapat mengelompokkan penyakit padi secara otomatis dari setiap piksel ke kelas tertentu. Pada tahap ini penulis menggunakan salah satu arsitektur *CNN* yakni *EfficientNetB4* karena berdasarkan *eksperiment* yang telah dilakukan penulis dengan dataset yang sama arsitektur *EfficientNetB4* memperoleh tingkat akurasi lebih baik dibanding arsitektur lain. Hal ini juga turut didukung dengan penelitian sebelumnya yang mengungkapkan bahwa arsitektur *EfficientNet* lebih

baik dibanding arsitektur sejenis lainnya seperti *ResNet*, *Xception*, *Inception* dan arsitektur lainnya. Berdasarkan hasil klasifikasi dengan arsitektur *EfficientNetB4* yang diterapkan pada dataset penyakit padi ini maka penulis memperoleh akurasi dan hasil klasifikasi yang baik, untuk hasil akurasi dan klasifikasi dibahas pada pembahasan selanjutnya mengenai poin evaluasi model. Hasil pengklasifikasian tersebut dipengaruhi oleh *hyperparameter* yang terdapat pada arsitektur terkait. Dalam penelitian ini penulis merancang metode yang digunakan dalam mengklasifikasikan penyakit padi seperti pada Gambar 4.4 berikut:



Sumber : (Penelitian, 2021)

Gambar 4.4 Rancangan Metode Usulan

Gambar 4.4 menggambarkan metode usulan, dimana pada jaringan *EfficientNetB4* yang pertama dilakukan *input* citra dengan resolusi 224x224x3 (RGB). Kemudian pada arsitektur *EfficientnetB4* memiliki 2 *convolution layers*, kemudian masuk ke *layers* selanjutnya yaitu *mobile bottleneck convolutional (MBConv)* yang terdiri dari 7 *layers*. Didalam *layers MBConv* terdapat beberapa operasi, diantaranya inputan dari lapisan sebelumnya yang menghubungkan jumlah saluran yang jauh lebih sedikit dibanding dengan lapisan ekspansi, dikombinasikan dengan konvolusi *depthwise* yang secara efektif mengurangi komputasi sebesar 2 kali kernel dibanding dengan lapisan tradisional. Kernel adalah yang menentukan tinggi dan lebar jendela konvolusi 2D. Pada *layers* ini juga ditambahkan pengoptimalan *squeeze* dan *excitation* (SE) yang berkontribusi

pada peningkatan kinerja lebih lanjut. Struktur ini membantu mengurangi jumlah keseluruhan operasi yang diperlukan serta ukuran model.

Selanjutnya 1 *pooling layers* dan 1 *fully connected* saya menggunakan mengadaptasi *Global average pooling2D* untuk mengubah tensor 4D (*batch_size*, *rows*, *cols*, *channels*) menjadi tensor 2D dengan bentuk (*batch_size*, *channels*) *Global average pooling* sehingga menghasilkan jumlah fitur yang jauh lebih sedikit dibanding dengan *flatten*, yang secara efektif mengurangi jumlah parameter. Kemudian pada layers *fully connected* juga tambahkan operasi *dropout* sebagai salah satu usaha untuk mencegah *overfitting* dan mempercepat proses *learning*.

Beberapa *hyperparameter* yang digunakan dalam mengklasifikasikan citra berbasis *deep learning* dengan arsitektur *EfficientNetB4* dapat dilihat pada Tabel 4.6 di bawah ini:

Tabel 4.6 Hyperparameter Arsitektur EfficientNet B4

Variabel	Nilai
<i>Include_top</i>	<i>False</i>
<i>Weight</i>	<i>Imagenet</i>
<i>Input_shape</i>	<i>(image_size, image_size, 3)</i>
<i>Pooling</i>	1280
<i>Classes</i>	1000 (jumlah kelas pada <i>imagenet</i>)
<i>Classifier_activation</i>	<i>Softmax</i>
<i>Image size</i>	224
<i>Batch size</i>	32
<i>loss</i>	<i>Categorical_crossentropy</i>
<i>optimizer</i>	<i>Adam</i>
<i>Learning rate</i>	0.001
<i>epochs</i>	25

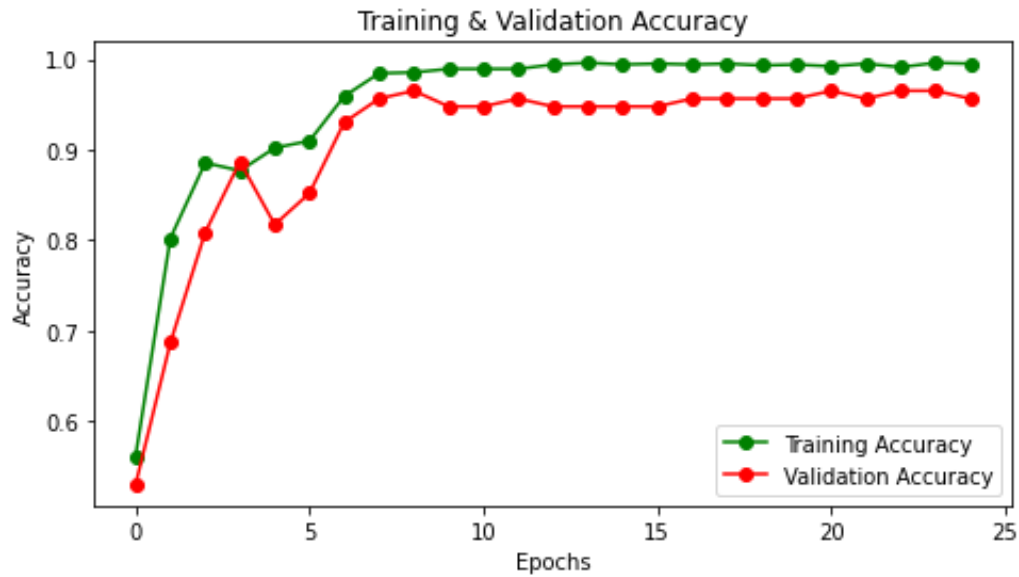
Variabel	Nilai
<i>Model.save</i>	<i>EffNet.h5</i>

Sumber: (Penelitian, 2021)

Pada Tabel 4.6 penulis menerapkan *transfer learning*, sehingga pada *include_top* penulis menggunakan *False* sehingga 3 *layer* terakhir tidak digunakan, sedangkan pada *Weight* penulis menggunakan *imagenet* yang berisi 1000 citra yang tersedia. *Loss* yang digunakan dalam penelitian ini adalah *categorical_crossentropy*. Karena salah satu syarat *categorical_crossentropy* dapat digunakan adalah proses klasifikasi menggunakan fungsi *softmax*. Pada arsitektur yang digunakan penulis juga menerapkan *learning rate* dengan nilai ambang batas 0.001. Serta melakukan *save model* dengan menggunakan *EffNet.h5*.

4.4. Evaluasi

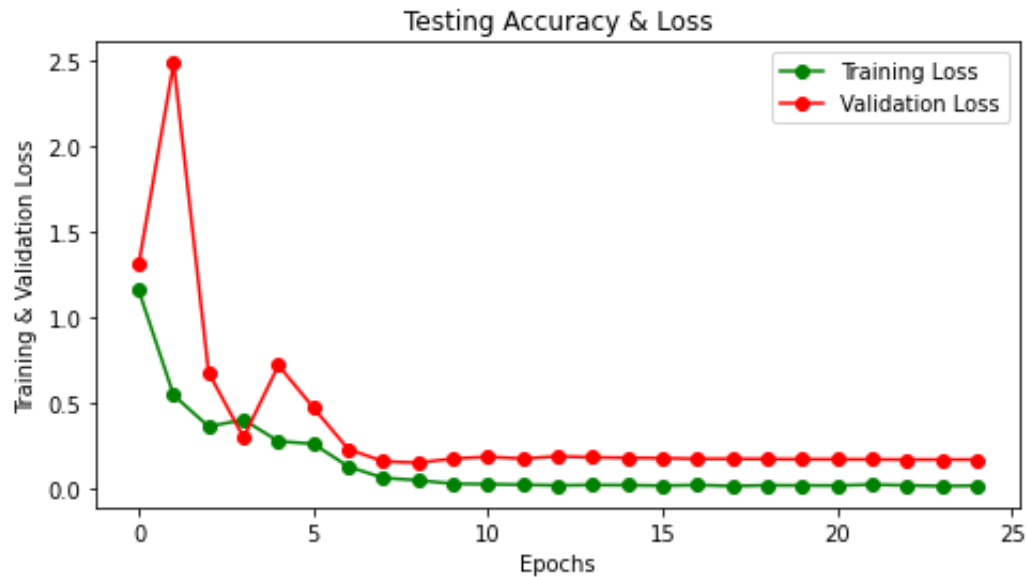
Pada tahap evaluasi ini penulis melakukan evaluasi dengan tujuan untuk melakukan pengujian terhadap keakuratan model yang dibangun. Melalui tahap ini dapat terlihat perbandingan hasil klasifikasi yang dilakukan oleh model dengan hasil klasifikasi sebenarnya. Dalam tahap ini penulis memvisualisasikan dalam bentuk *confusion matrix* seperti pada Gambar 4.5. Dimana *confusion matrix* ini ditampilkan dalam berbentuk tabel matriks yang menggambarkan kinerja model klasifikasi pada serangkaian data uji yang nilai sebenarnya diketahui. Penulis menggunakan evaluasi model berdasarkan nilai *accuracy* dalam bentuk grafik seperti pada Gambar 4.5 dan Gambar 4.6. Selain itu penulis juga menggunakan *testing accuracy* dan *testing loss* untuk mengevaluasi model seperti pada Tabel 4.7 dan *classification report* pada Tabel 4.8.



Sumber : (Penelitian, 2021)

Gambar 4.5 Training dan Validation Accuracy

Gambar 4.5 di atas menunjukkan perbandingan *accuracy* antara *test* dan *validation* dari setiap *epoch*. Dimana *validation accuracy* ini diambil pada proses *validation_split* pada saat *fitting model* sehingga diperoleh dataset terpisah yakni data latih dan data uji yang datasetnya dibagi secara acak. Berdasarkan grafik tersebut terlihat bahwa tingkat akurasi *validation* lebih rendah dari *train* dikarenakan selama proses *validation* tidak menggunakan *droupout*. Berdasarkan grafik tersebut pula dapat dilihat bahwa hasil terabaran plot antara *train accuracy* dan *validation accuracy* tidak terlampau jauh sehingga kurva pembelajaran tersebut menunjukkan *good fit*.



Sumber : (Penelitian, 2021)

Gambar 4.6 Training Loss dan Validation Loss

Jika pada grafik sebelumnya menampilkan tingkat akurasi antara *testing* dan *validation*, maka pada Gambar 4.6 menampilkan nilai *loss* dari *train* dan *validation*. Dimana pada grafik tersebut terlihat bahwa pada *epoch* pertama hingga *epoch* ke 5 relatif belum stabil namun plot *train loss* sudah mulai menurun ke titik stabilitas pada *epoch* kedua. Begitu pula dengan plot *validation loss* yang berkurang ke titik stabilitas dan memiliki sedikit celah dengan *train loss*, selain itu *loss* model data *train* lebih rendah daripada dataset *validation*. Sehingga berdasarkan kurva pembelajaran menunjukkan *good fit*.

Tabel 4.7. Evaluasi Model

<i>Testing Loss</i>	<i>Testing Accuracy</i>
19.19%	94.53 %

Sumber : (Penelitian, 2021)

Berdasarkan hasil prediksi penyakit padi, maka penulis melaporkan *classification report* yang terdiri dari; nilai *precision*, *recall*, dan nilai *f1-score* seperti pada Tabel 4.8 berikut:

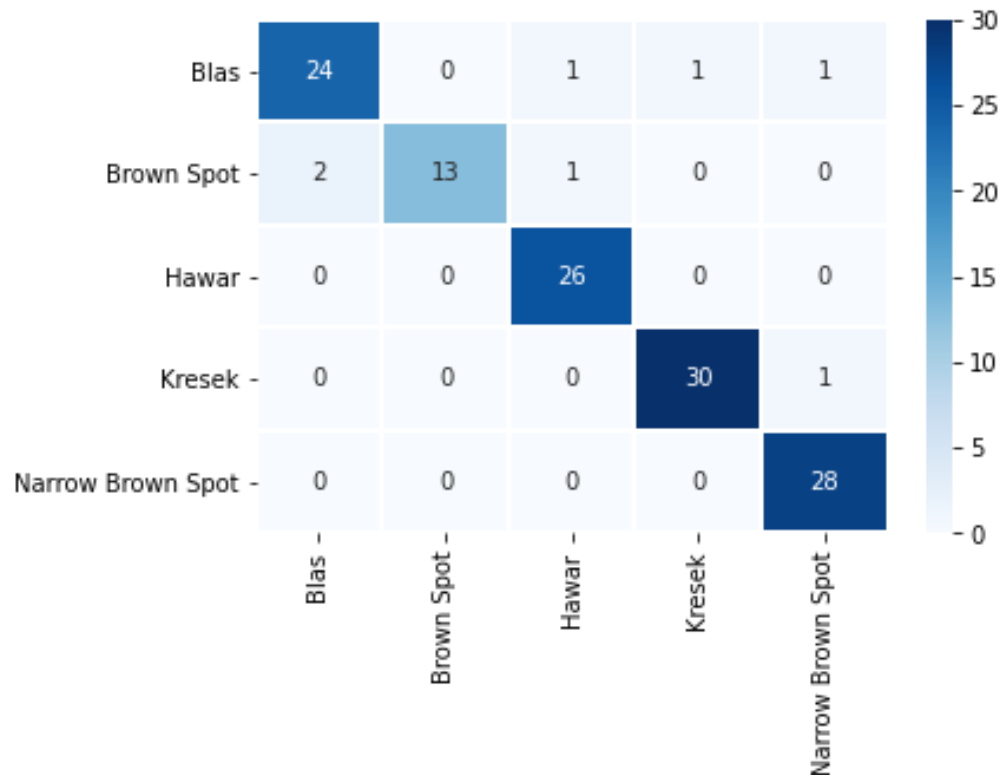
Tabel 4.8. Classification Report

Kelas	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Blas	0.92	0.89	0.91
<i>Brown Spot</i>	1.00	0.81	0.90
Hawar	0.93	1.00	0.96
Kresek	0.97	0.97	0.97
<i>Narrow Brown Spot</i>	0.93	1.00	0.97

Sumber : (Penelitian, 2021)

Berdasarkan Tabel 4.8 diatas penulis melaporkan evaluasi kinerja model kedalam bentuk *precision*, *recall* dan *f1-score*. Berdasarkan informasi pada tabel tersebut nilai *precision* yang dihasilkan menggambarkan bahwa tingkat keakuratan antara data penyakit padi dengan hasil prediksi dari model menunjukkan keakuratan yang tinggi yakni diatas 90%. Selanjutnya untuk nilai *recall* terlihat bahwa tingkat keberhasilan model dalam menemukan informasi atau memprediksi *true positive (TP)* dibanding keseluruhan data *true positive (TP)*. Tingkat keberhasilan model dalam memprediksi kelas *Brown Spot* masih lemah dibandingkan pada saat model memprediksi kelas lainnya. Model ini juga dinilai berhasil memprediksi data yang benar terbukti dapat memprediksi hingga 100% pada kelas Hawar dan *Narrow Brown Spot*. Laporan klasifikasi selanjutnya yakni nilai *f1-score* dimana berdasarkan tabel tersebut *f1-score* memiliki skor yang baik yakni memiliki nilai diatas 90% sehingga mengindikasikan bahwa model klasifikasi yang digunakan mempunyai *precision* dan *recall* yang baik.

Berdasarkan informasi perbandingan yang telah diperoleh dari hasil klasifikasi penyakit padi yang dilakukan oleh sistem dengan hasil klasifikasi yang seharusnya, maka penulis menggunakan *confusion matrix* untuk mengukur kinerja metode seperti di bawah ini:



Sumber : (Penelitian, 2021)

Gambar 4.7 Confusion Matrix

Berdasarkan hasil akurasi yang ditampilkan dalam bentuk *confusion matrix* diatas dapat diketahui bahwa kelas pertama yakni Blas memiliki akurasi sebesar 89% dimana 24 citra terprediksi Blas sedangkan 3 citra Blas masing-masing terprediksi pada kelas Hawar, Kresek dan *Narrow Brown Spot*. Kelas kedua yakni *Brown Spot* menghasilkan akurasi terendah dibanding kelas lainnya yakni sebesar 81% karena 13 citra terprediksi kelas *Brown Spot* sedangkan 2 citra *brown Spot* terprediksi Blas dan 1 citra terprediksi Hawar. Kelas ketiga yakni Hawar menghasilkan tingkat akurasi sebesar 100%, dimana 26 citra seluruhnya terprediksi Hawar. Kelas keempat yaitu Kresek menghasilkan tingkat akurasi sebesar 97% dimana 30 citra terprediksi Kresek dan 1 citra Kresek terprediksi pada kelas *Narrow Brown Spot*. Kelas terakhir yakni *Narrow Brown Spot*

Tabel 4.9 Sampel Hasil Klasifikasi

Data Testing	Jenis Penyakit	Hasil Klasifikasi	Keterangan
1	Blas	Blas	<i>True</i>
2	Blas	Blas	<i>True</i>
3	Blas	Hawar	<i>False</i>
4	Blas	Kresek	<i>False</i>
5	Blas	<i>Narrow Brown Spot</i>	<i>False</i>
6	<i>Brown Spot</i>	<i>Brown Spot</i>	<i>True</i>
7	<i>Brown Spot</i>	<i>Brown Spot</i>	<i>True</i>
8	<i>Brown Spot</i>	Blas	<i>False</i>
9	<i>Brown Spot</i>	Blas	<i>False</i>
10	<i>Brown Spot</i>	Hawar	<i>False</i>
11	Hawar	Hawar	<i>True</i>
12	Hawar	Hawar	<i>True</i>
13	Hawar	Hawar	<i>True</i>
14	Hawar	Hawar	<i>True</i>
15	Hawar	Hawar	<i>True</i>
16	Kresek	Kresek	<i>True</i>
17	Kresek	Kresek	<i>True</i>
18	Kresek	Kresek	<i>True</i>
19	Kresek	Kresek	<i>True</i>
20	Kresek	<i>Narrow Brown Spot</i>	<i>False</i>
21	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
22	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
23	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
24	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
25	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>

Sumber : (Penelitian, 2021)

BAB V PENUTUP

5.1. Kesimpulan

Berdasarkan permasalahan terkait penyakit pada tanaman padi yang sering dialami oleh petani padi sehingga mengakibatkan gagal panen. Maka pada penelitian ini dilakukan diagnosis terhadap penyakit padi yang telah dilakukan pada 1275 citra dari 5 jenis penyakit padi yang diklasifikasi dengan metode yang telah diusulkan sehingga berhasil memenuhi tujuan yang diinginkan diantaranya:

1. Proses pengambilan citra atau proses akuisisi citra penyakit padi dapat dilakukan dengan bantuan kamera *handphone* Samsung A10, sehingga diperoleh sebanyak 194 citra asli.
2. Proses klasifikasi manual yang dilakukan oleh ahli melalui media kuesioner terhadap 194 citra yang telah diambil, maka diperoleh 5 jenis penyakit padi diantaranya Blas, *Brown Spot*, Hawar, Kresek, dan *Narrow Brown Spot* serta 2 jenis hama tanaman padi yakni Nekrosis dan Putih Palsu.
3. Dari proses *augmentation* yang dilakukan dengan 8 teknik *augmentation* diperoleh variasi citra yang lebih beragam sehingga diperoleh 1275 citra yang siap diolah.
4. Proses segmentasi dengan metode *K-Means* berhasil mensegmentasi bercak pada daun yang sakit sehingga objek yang diinginkan yaitu bercak pada daun dapat terdeteksi.
5. Berdasarkan klasifikasi penyakit yang dilakukan dengan metode *CNN* arsitektur *EfficientNetB4*, maka diperoleh tingkat akurasi sebesar 94.53% dari 128 citra *testing*. Dari 128 citra tersebut diperoleh 24 citra terklasifikasi penyakit Blas, sedangkan 3 citra Blas terklasifikasi penyakit Hawar, Kresek dan *Narrow Brown Spot*. 13 citra terklasifikasi penyakit *Brown Spot* sedangkan 2 citra *Brown Spot* terklasifikasi penyakit Blas dan Hawar, 26 citra terklasifikasi penyakit Hawar. Selanjutnya 30 citra terklasifikasi penyakit Kresek dan 1 citra Kresek terklasifikasi *Narrow Brown Spot*. Pada Kelas terakhir 28 citra terklasifikasi penyakit

Narrow

Brown

Spot.

5.2. Saran

Proses penelitian yang telah dilakukan oleh penulis pada pengklasifikasian penyakit padi ini dapat dikembangkan menjadi penelitian yang lebih baik, sehingga penulis menyampaikan beberapa saran sebagai berikut:

1. Jenis penyakit pada tanaman padi masih banyak macamnya, sehingga masih banyak kemungkinan dilakukan penelitian lain.
2. Penyebab gagal panen pada tanaman padi sendiri tidak hanya disebabkan oleh penyakit, namun juga hama tanaman padi sehingga penelitian mengenai hama tanaman padi juga menarik untuk diteliti.
3. Mengingat citra awal yang diperoleh penulis jumlahnya tidak mencukupi untuk diolah dan jumlah antar kelas tidak seimbang, maka pada penelitian berikutnya dapat memperbanyak jumlah citra dari masing-masing kelas untuk meningkatkan penelitian.
4. Dapat dilakukan *eksperiment* menggunakan metode lain untuk melakukan klasifikasi penyakit padi agar dapat ditemukan metode yang paling cocok.
5. Pada penelitian berikutnya dapat dibuat pengaplikasian dari klasifikasi penyakit padi dalam bentuk *website* maupun *mobile app*.

REFERENSI

- [1] K. P. R. INDONESIA, “Asuransi Usaha Tani Padi, Solusi Kegagalan Panen.” <https://www.pertanian.go.id/home/?show=news&act=view&id=1609>.
- [2] B. Nuryanto, “Pengendalian Penyakit Tanaman Padi Berwawasan Lingkungan Melalui Pengelolaan Komponen Epidemik,” *J. Penelit. dan Pengemb. Pertan.*, vol. 37, no. 1, p. 1, 2018, doi: 10.21082/jp3.v37n1.2018.p1-8.
- [3] S. Zahrah, R. Saptono, and E. Suryani, “Identifikasi Gejala Penyakit Padi Menggunakan Operasi Morfologi Citra,” *Snik*, no. Snik, pp. 100–106, 2016.
- [4] M. N. Yudhistira Nugraha, Suprihanto, Ainur Farid, Agus Guswara, Idrus Hasmi, Suharna, “Laporan Tahunan 2018,” *Balai Besar Penelit. Tanam. Padi*, pp. 1–78, 2019.
- [5] D. A. Satria, “Gagasan & Peran IPB untuk Pertanian 4.0.” IPB, Bogor, p. 21, 2018, [Online]. Available: http://drn.go.id/files/2018/SP_1_DRN/Sesi_B_4__Dr__Arif_Satria_Pertanian_4_0_Semnas_DRN-compressed.pdf.
- [6] A. A. Alfariy, Q. Chen, and M. Guo, “Deep learning based classification for paddy pests & diseases recognition,” *ACM Int. Conf. Proceeding Ser.*, pp. 21–25, 2018, doi: 10.1145/3208788.3208795.
- [7] R. A. Saputra, Suharyanto, S. Wasiyanti, D. F. Saefudin, A. Supriyatna, and A. Wibowo, “Rice Leaf Disease Image Classifications Using KNN Based on GLCM Feature Extraction,” *J. Phys. Conf. Ser.*, vol. 1641, no. 1, 2020, doi: 10.1088/1742-6596/1641/1/012080.
- [8] M. Suresha, K. N. Shreekanth, and B. V. Thirumalesh, “Recognition of diseases in paddy leaves using knn classifier,” *2017 2nd Int. Conf. Conver. Technol. I2CT 2017*, vol. 2017-Janua, pp. 663–666, 2017, doi: 10.1109/I2CT.2017.8226213.
- [9] K. Ahmed, T. R. Shahidi, S. M. Irfanul Alam, and S. Momen, “Rice leaf disease detection using machine learning techniques,” *2019 Int. Conf. Sustain. Technol. Ind. 4.0, STI 2019*, vol. 0, pp. 1–5, 2019, doi:

10.1109/STI47673.2019.9068096.

- [10] Z. Xu *et al.*, “Using deep convolutional neural networks for image-based diagnosis of nutrient deficiencies in rice,” *Comput. Intell. Neurosci.*, vol. 2020, 2020, doi: 10.1155/2020/7307252.
- [11] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, “Image Processing Techniques for Diagnosing Rice Plant Disease: A Survey,” *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 516–530, 2020, doi: 10.1016/j.procs.2020.03.308.
- [12] N. V. R. R. Goluguri, K. Suganya Devi, and N. Vadaparathi, *Image classifiers and image deep learning classifiers evolved in detection of Oryza sativa diseases: survey*, vol. 54, no. 1. Springer Netherlands, 2021.
- [13] C. R. Rahman *et al.*, “Identification and recognition of rice diseases and pests using convolutional neural networks,” *Biosyst. Eng.*, vol. 194, pp. 112–120, 2020, doi: 10.1016/j.biosystemseng.2020.03.020.
- [14] H. B. Prajapati, J. P. Shah, and V. K. Dabhi, “Detection and classification of rice plant diseases,” *Intell. Decis. Technol.*, vol. 11, no. 3, pp. 357–373, 2017, doi: 10.3233/IDT-170301.
- [15] T. Daniya and S. Vigneshwari, “A review on machine learning techniques for rice plant disease detection in agricultural research,” *Int. J. Adv. Sci. Technol.*, vol. 28, no. 13, pp. 49–62, 2019.
- [16] T. Kodama and Y. Hata, “Development of Classification System of Rice Disease Using Artificial Intelligence,” *Proc. - 2018 IEEE Int. Conf. Syst. Man, Cybern. SMC 2018*, pp. 3699–3702, 2019, doi: 10.1109/SMC.2018.00626.
- [17] M. E. Pothen and D. M. L. Pai, “Detection of Rice Leaf Diseases Using Image Processing,” *Proc. 4th Int. Conf. Comput. Methodol. Commun. ICCMC 2020*, no. Iccmc, pp. 424–430, 2020, doi: 10.1109/ICCMC48092.2020.ICCMC-00080.
- [18] K. S. Archana and A. Sahayadhas, “Automatic rice leaf disease segmentation using image processing techniques,” *Int. J. Eng. Technol.*, vol. 7, no. 3.27 Special Issue 27, pp. 182–185, 2018, doi: 10.14419/ijet.v7i3.27.17756.
- [19] S. D. Khirade and A. B. Patil, “Plant disease detection using image

- processing,” *Proc. - 1st Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2015*, pp. 768–771, 2015, doi: 10.1109/ICCUBEA.2015.153.
- [20] A. Z. Siregar, *Karakteristik Hama Padi Pasang Surut*. Intimedia, 2018.
- [21] D. Nababan, “Data Mining Data Set,” 2018. <https://slideplayer.info/slide/13913621/>.
- [22] D. (Dian Nuswantoro University), “Dataset Proses Utama pada Data Mining.”
- [23] I. Weni, P. E. P. Utomo, B. F. Hutabarat, and M. Alfalah, “Detection of Cataract Based on Image Features Using Convolutional Neural Networks,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 15, no. 1, p. 75, 2021, doi: 10.22146/ijccs.61882.
- [24] A. Zarkasi and H. Ubaya, “Vision Sebagai Pengolahan Citra Api,” *Konf. Nas. Teknol. Inf. Apl.*, vol. 4, pp. 39–44, 2016.
- [25] W. B. dan M. J. Burge, *Digital Image Processing An Algorithmic Introduction Using Java Secong Edition*, vol. 19, no. 3. 2016.
- [26] D. F. Sengkey, F. D. Kambey, S. P. Lengkong, S. R. Joshua, and H. V. F. Kainde, “Pemanfaatan Platform Pemrograman Daring dalam Pembelajaran Probabilitas dan Statistika di Masa Pandemi CoVID-19,” *J. Inform.*, vol. 15, no. 4, pp. 217–224, 2020.
- [27] F. Perez-Sanz, P. J. Navarro, and M. Egea-Cortines, “Plant phenomics: An overview of image acquisition technologies and image data analysis algorithms,” *Gigascience*, vol. 6, no. 11, pp. 1–18, 2017, doi: 10.1093/gigascience/gix092.
- [28] M. Fatchan and C. E. Dharma, “Identifikasi Salah Pemasangan Komponen Dengan Algorithma Template Matching,” pp. 4–5, 2019.
- [29] S. Perumal and T. Velmurugan, “Preprocessing by Contrast Enhancement Techniques for Medical Images,” *Int. J. Pure Appl. Math.*, vol. 118, no. 18, pp. 3681–3688, 2018.
- [30] N. Disri, “Edit Background Foto Jadi Gampang Banget dengan 4 situs Gratis Ini!,” 2020. <https://teknologi.id/insight/edit-background-foto-jadi-gampang-banget-dengan-4-situs-gratis-ini>.

- [31] I. Z. Mukti and D. Biswas, "Transfer Learning Based Plant Diseases Detection Using ResNet50," *2019 4th Int. Conf. Electr. Inf. Commun. Technol. EICT 2019*, no. December, pp. 1–6, 2019, doi: 10.1109/EICT48899.2019.9068805.
- [32] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut*. Bandung: Informatika, 2018.
- [33] A. Atina, "Segmentasi Citra Paru Menggunakan Metode k-Means Clustering," *J. Pendidik. Fis. dan Keilmuan*, vol. 3, no. 2, p. 57, 2017, doi: 10.25273/jpfk.v3i2.1475.
- [34] A. Premana, R. M. H. Bhakti, and D. Prayogi, "Segmentasi K-Means Clustering Pada Citra Menggunakan Ekstraksi Fitur Warna dan Tekstur," *J. Ilm. Intech Inf. Technol. J. UMUS*, vol. 2, no. 01, 2020, doi: 10.46772/intech.v2i01.190.
- [35] J. W. G. Putra, "Pengenal Konsep Pembelajaran Mesin dan Deep Learning," *Comput. Linguist. Nat. Lang. Process. Lab.*, vol. 4, pp. 1–235, 2019, [Online]. Available: <https://www.researchgate.net/publication/323700644>.
- [36] M. L. C. Course, "Training and Test Sets: Splitting Data," 2020. <https://developers.google.com/machine-learning/crash-course/training-and-test-sets/splitting-data?hl=id>.
- [37] Suyanto, *Data Mining Untuk Klasifikasi dan Klasterisasi Data*. Bandung: Informatika, 2017.
- [38] E. Firasari, N. Khasanah, U. Khultsum, D. N. Kholifah, R. Komarudin, and W. Widyastuty, "Comparison of K-Nearest Neighbor (K-NN) and Naive Bayes Algorithm for the Classification of the Poor in Recipients of Social Assistance," *J. Phys. Conf. Ser.*, vol. 1641, no. 1, 2020, doi: 10.1088/1742-6596/1641/1/012077.
- [39] L. T. Duong, P. T. Nguyen, C. Di Sipio, and D. Di Ruscio, "Automated fruit recognition using EfficientNet and MixNet," *Comput. Electron. Agric.*, vol. 171, no. February, p. 105326, 2020, doi: 10.1016/j.compag.2020.105326.
- [40] Z. Huang, Z. Pan, and B. Lei, "Transfer learning with deep convolutional neural network for SAR target classification with limited labeled data," *Remote*

- Sens.*, vol. 9, no. 9, pp. 1–21, 2017, doi: 10.3390/rs9090907.
- [41] Suyanto, *Deep Learning Modernisasi Machine Learning untuk Big Data*. Bandung: Informatika Bandung, 2019.
 - [42] J. Brownlee, *Deep Learning for Natural Language Processing*, V1.1. 2017.
 - [43] B. Purnama, *Pengantar Machine Learning*. Bandung: Informatika, 2019.
 - [44] M. F. Dzulqarnain, S. Suprpto, and F. Makhrus, “Improvement of Convolutional Neural Network Accuracy on Salak Classification Based Quality on Digital Image,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 2, p. 189, 2019, doi: 10.22146/ijccs.42036.
 - [45] R. Prathivi, “Optimasi Model TL-CNN Untuk Klasifikasi Citra CIFAR-10,” *J. Rekayasa Sist. dan Teknol. Inf.*, vol. 1, no. 10, pp. 3–7, 2019.
 - [46] G. Marques, D. Agarwal, and I. de la Torre Díez, “Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network,” *Appl. Soft Comput. J.*, vol. 96, p. 106691, 2020, doi: 10.1016/j.asoc.2020.106691.
 - [47] “Complete Architectural Details of all EfficientNet Models,” *Toward Data Science*, 2020. <https://towardsdatascience.com/complete-architectural-details-of-all-efficientnet-models-5fd5b736142>.
 - [48] Keras, “EfficientNet B0 to B7,” *Keras*. <https://keras.io/api/applications/efficientnet/>.
 - [49] S. Narkhede, “Understanding Confusion Matrix,” *Toward Data Science*, 2018. <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62>.
 - [50] J. Brownlee, “What is a Confusion Matrix in Machine Learning,” *Machine Learning Mastery*, 2016. <https://machinelearningmastery.com/confusion-matrix-machine-learning/>.
 - [51] S. Setiawan, “Membicarakan Precision, Recall, dan F1-Score,” 2020. <https://stevkarta.medium.com/membicarakan-precision-recall-dan-f1-score-e96d81910354>.
 - [52] J. Brownlee, “How to use Learning Curves to Diagnose Machine Learning

- Model Performance,” 2019. <https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/>.
- [53] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.
 - [54] B. S. Prajapati, V. K. Dabhi, and H. B. Prajapati, “A survey on detection and classification of cotton leaf diseases,” *Int. Conf. Electr. Electron. Optim. Tech. ICEEOT 2016*, pp. 2499–2506, 2016, doi: 10.1109/ICEEOT.2016.7755143.
 - [55] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, “Identification of rice diseases using deep convolutional neural networks,” *Neurocomputing*, vol. 267, pp. 378–384, 2017, doi: 10.1016/j.neucom.2017.06.023.
 - [56] W. jie Liang, H. Zhang, G. feng Zhang, and H. xin Cao, “Rice Blast Disease Recognition Using a Deep Convolutional Neural Network,” *Sci. Rep.*, vol. 9, no. 1, pp. 1–10, 2019, doi: 10.1038/s41598-019-38966-0.
 - [57] M. J. Hasan, S. Mahbub, M. S. Alom, and M. Abu Nasim, “Rice Disease Identification and Classification by Integrating Support Vector Machine with Deep Convolutional Neural Network,” *1st Int. Conf. Adv. Sci. Eng. Robot. Technol. 2019, ICASERT 2019*, vol. 2019, no. Icasert, pp. 1–6, 2019, doi: 10.1109/ICASERT.2019.8934568.
 - [58] V. K. Shrivastava and M. K. Pradhan, “Rice plant disease classification using color features: a machine learning paradigm,” *J. Plant Pathol.*, vol. 103, no. 1, pp. 17–26, 2021, doi: 10.1007/s42161-020-00683-3.
 - [59] F. Jiang, Y. Lu, Y. Chen, D. Cai, and G. Li, “Image recognition of four rice leaf diseases based on deep learning and support vector machine,” *Comput. Electron. Agric.*, vol. 179, no. October, p. 105824, 2020, doi: 10.1016/j.compag.2020.105824.
 - [60] J. Chen, D. Zhang, Y. A. Nanekaran, and D. Li, “Detection of rice plant diseases based on deep transfer learning,” *J. Sci. Food Agric.*, vol. 100, no. 7, pp. 3246–3256, 2020, doi: 10.1002/jsfa.10365.
 - [61] D. Li *et al.*, “A recognition method for rice plant diseases and pests video

detection based on deep convolutional neural network,” *Sensors (Switzerland)*, vol. 20, no. 3, 2020, doi: 10.3390/s20030578.

- [62] N. V. R. R. Goluguri, K. S. Devi, and P. Srinivasan, “Rice-net: an efficient artificial fish swarm optimization applied deep convolutional neural network model for identifying the *Oryza sativa* diseases,” *Neural Comput. Appl.*, vol. 33, no. 11, pp. 5869–5884, 2020, doi: 10.1007/s00521-020-05364-x.

DAFTAR RIWAYAT HIDUP

A. Biodata Mahasiswa

NIM : 14002387
 Nama : Fajar Sarasati
 Tempat dan Tanggal Lahir : Klaten, 27 September 1997
 Jenis Kelamin : Perempuan
 Alamat : Mranggen, Borangan, Manisrenggo, Klaten

B. Riwayat Pendidikan Formal dan Non-Formal

1. SD N 1 BORANGAN, lulus tahun 2009
2. SMP N 2 MANISRENGGO, lulus tahun 2012
3. SMK N 1 KLATEN, lulus tahun 2015
4. DIII Manajemen Informatika, lulus tahun 2018
5. S1 Sistem Informasi, lulus tahun 2019

C. Riwayat Pengalaman Organisasi/Pekerjaan

1. Anggota UKM Paduan Suara Mahasiswa UBSI Yogyakarta Sinfonia D'Amore tahun 2015/2018
2. Sekretaris Karang Taruna FORMUDA/Forum Muda Mudi RW Mranggen tahun 2018
3. Pengajar TK Miftahul Ulum Desa Borangan, Manisrenggo, Klaten Tahun 2019
4. Staf dan Dosen S1 Universitas Nusa Mandiri, 2019-Sekarang



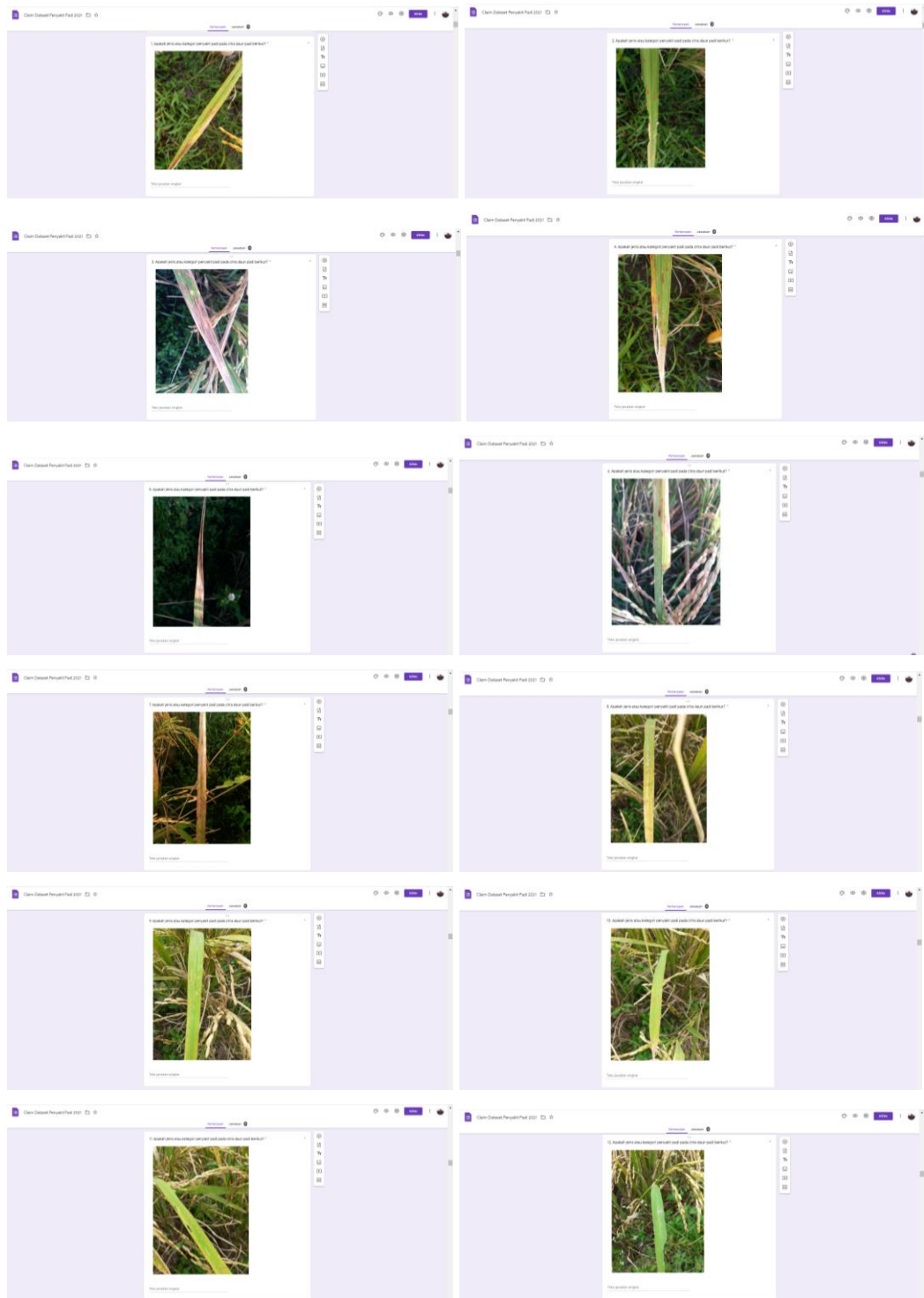
Jakarta, 23 Agustus 2021

(Fajar Sarasati)

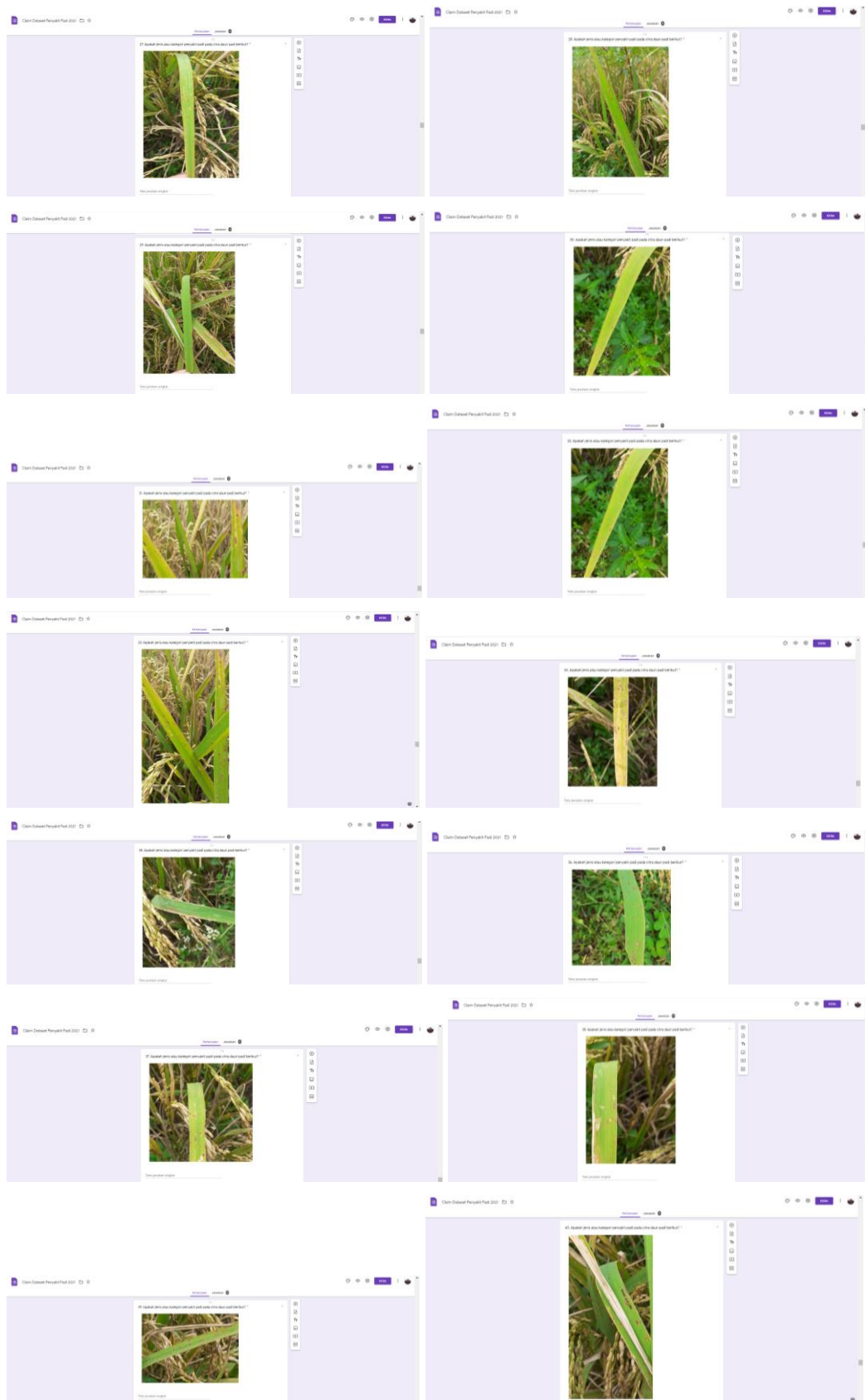
LAMPIRAN-LAMPIRAN

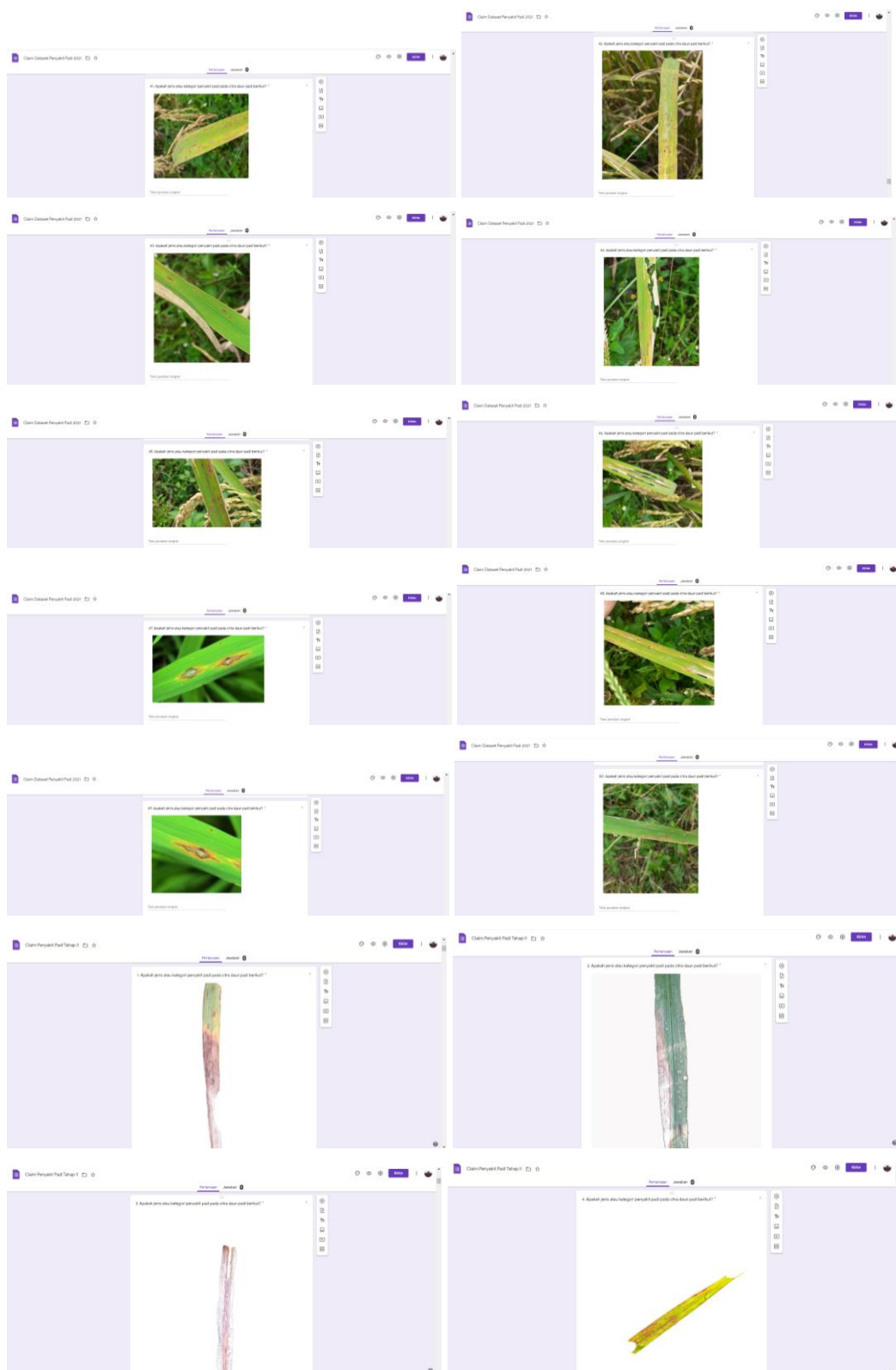
Lampiran 1. Pertanyaan Kuesioner

Link kuesioner : http://bit.ly/Form_RiceDiseases, dan
http://bit.ly/ClaimPenyakitPadi_2





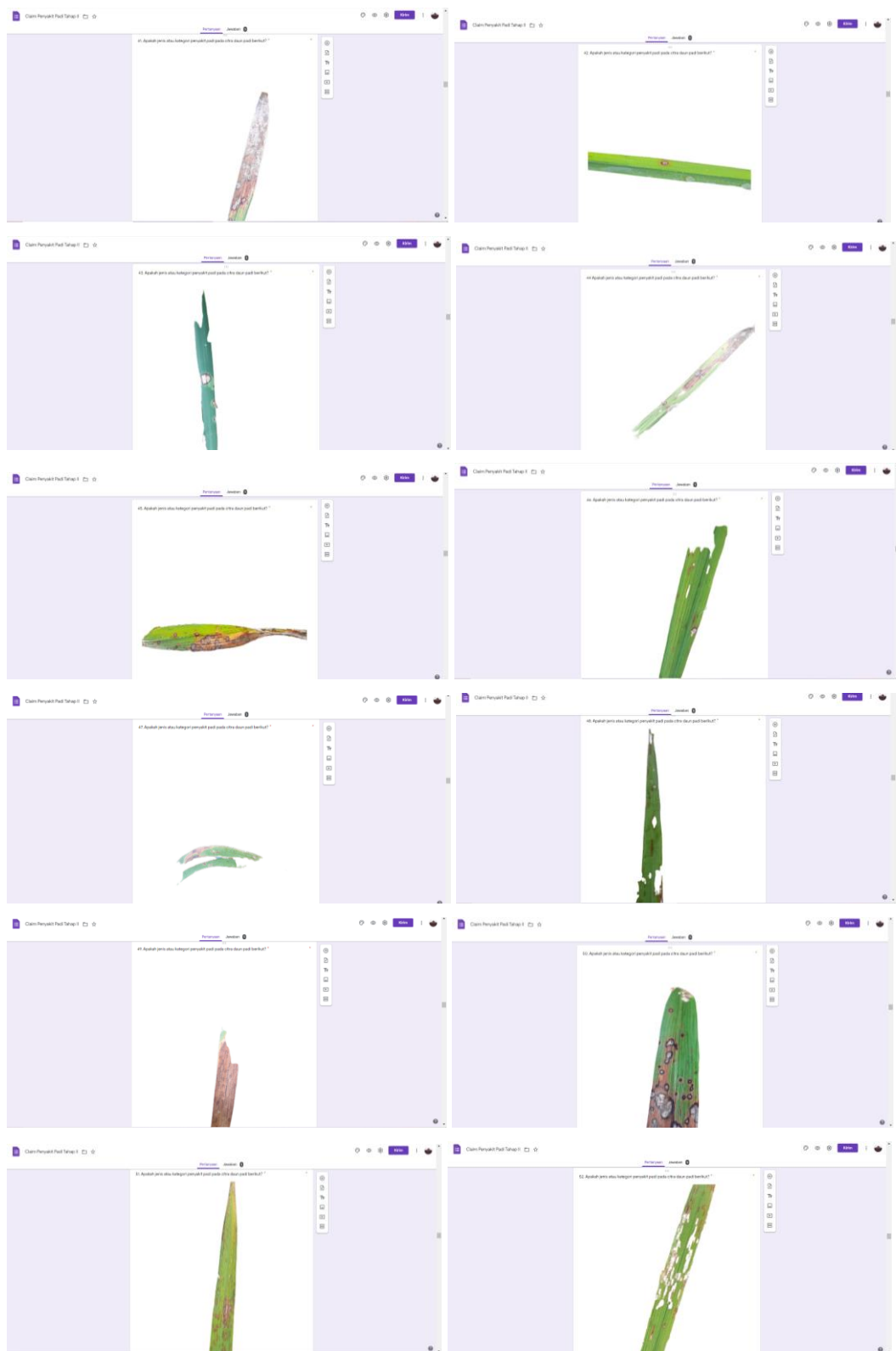






















Lampiran 2. Hasil Pengisian Kuesioner

Tahap 1

Nama Lengkap		Anisa Rahma Untari	Jaka Susilo	Lufi Indriyani	Khaterine Jane Yasmine	Ir. Darmo Agus Nurcahyo	Danang Taruno
Jurusan/ Fakultas		Penyuluhan pertanian berkelanjutan /pertanian	Pertanian	Agroteknologi/Pertanian	Agroteknologi	Budidaya Pertanian/Pertanian	Agroteknologi/Pertanian
Nama Instansi/ Kampus		Politeknik pembangunan pertanian yogyakarta-magelang	UGM	Universitas Sarjanawiyata Tamansiswa	Universitas Sarjanawiyata Tamansiswa	DPUPR Kab. Magelang/ Universitas Slamet Riyadi	UNS
Tahun lulus		2019	1990	2019	Blm lulus	1992	2017
Keahlian dan Pengetahuan		Penyuluhan Pertanian	HamaTanaman	Agronomi	Budidaya tanaman	Pola dan Rencana Tata Tanam pada Operasi Pemeliharaan Irigasi	Ilmu Tanah
Pertanyaan	1	Kresek/hawar daun	Brown leaf spot	Hawar daun	Tungro	Penyakit hawar daun bakteri atau kresek	Bercak padi
	2	Bercak daun	Brown leaf spot	Hama belalang	Blas daun	Penyakit Blas	Hama
	3	Kresek/hawar daun dan fusarium	Hawar daun	Hawar daun	Tungro	Penyakit Blas	Fungi
	4	Bercak daun dan hawar	Hawar daun	Hawar daun	Ulat penggerak	Penyakit Hawar Daun Bakteri	Defisiensi hara
	5	Kresek/hawar daun	Hawar daun	Hawar daun	Tungro	Penyakit Kresek	Defisiensi
	6	Blast dan ulat daun	Brown leaf spot	Hama belalang	Blas daun	Penyakit Blas	Hama
	7	Kresek/hawar daun	Hawar daun	Hawar daun	Hawar daun	Penyakit Kresek	Fungi
	8	Bercak daun	Brown leaf spot (brown spot)	Blast	Bercak daun	Penyakit Blas	Fungi
	9	Bercak daun	Brown leaf spot	Blast	Bercak daun	Penyakit Blas	Hama
	10	Kresek	Brown leaf spot	Hawar daun	Blas daun	Penyakit Blas	Hama
	11	Bercak daun	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Hama
	12	Bercak daun	Brown leaf spot	Hama belalang	Blas daun	Penyakit Blas	Hama
	13	Hawar dan bercak daun	Brown leaf spot	Blast	Bercak daun	Penyakit Blas	Fungi
	14	Hawar	Brown leaf spot	Hawar daun	Blas daun	Penyakit Kresek	Defisiensi
	15	Hawar tungro	Hawar daun	Hawar daun	Tungro	Penyakit Kresek	Defisiensi
	16	Bercak daun hawar dan hispa	Brown leaf spot	Hawar daun	Hawar	Penyakit Blas	Hama
	17	Bercak daun	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Kresek	Hama
	18	Hawar	Hawar daun	Hawar daun	Tungro	Penyakit Blas	Defisiensi
	19	Hawar dan bercak daun	Brown leaf spot /bercak daun coklat	Hawar daun	Tungro	Penyakit Blas	Fungi
	20	Bercak daun	Brown leaf spot	Blast	Bercak daun	Penyakit Blas	Hama
	21	Hispa dan bercak daun	Kerusakan mekanis / necrosis	Hama belalang	Hawar daun	Penyakit Blas	Hama
	22	Hispa dan bercak daun	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Hama
	23	Bercak daun	Brown leaf spot	Hawar daun	Blas daun	Penyakit Blas	Hama

	Hawar daun dan bercak daun					
24		Brown leaf spot	Hawar daun	Tungro	Penyakit Blas	Defisiensi
25	Hispa	Kerusaka mekanis hama / necrosis	Hama belalang	Ulat penggerak	Penyakit Blas	Hama
26	Bercak daun	Brown leaf spot	Blast	Bercak daun	Penyakit Blas	Fungi
27	Bercak daun dan hispa	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Hama
28	Bercak daun	Brown leaf spot	Hawar daun	Blas daun	Penyakit Blas	Fungu
29	Bercak daun	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Hama
30	Bercak daun	Brown leaf spot	Hawar daun	Blas daun	Penyakit Blas	Hama
31	Tungro blast dan hawar	Brown leaf spot dan hawar daun	Hawar daun	Tungro	Penyakit Blas	Fungi
32	Bercak daun	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Fungi
33	Blast	Brown leaf spot	Hawar daun	Blas daun	Penyakit Blas	Fungi
34	Tungro dan blast	Brown leaf spot dan hawar daun	Blast	Blas daun	Penyakit Blas	Fungi
35	Hawar	Brown leaf spot	Hawar daun	Tunhro	Penyakit Blas	Fungu
36	Bercak daun	Brown leaf spot	Blast	Bercak daun	Penyakit Blas	Hama
37	Blast	Brown leaf spot	Hawar daun	Blas daun	Penyakit Blas	Hama
38	Bercak daun dan hawar	Penyakit blas dan brown leaf spot	Hawar daun	Tungro	Penyakit Blas	Hama
39	Bercak daun	Brown leaf spot	Hawar daun	Blas daun	Penyakit Blas	Fungi
40	Hawar	Hawar daun	Hawar daun	Hawar	Penyakit Blas	Hama
41	Bercak daun	Brown leaf spot	Blast	Bercak daun	Penyakit Blas	Fungi
42	Bercak daun dan fu	Brown leaf spot	Blast	Blas daun	Penyakit Blas	Defisiensi
43	Fusarium dan bercak daun	Hawar daun dan brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Fungi
44	Bercak daun	Kerusakan mekanis dari hama	Hama	Hawar daun	Penyakit Blas	Hama
45	Bercak daun blast	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Fungi
46	Hispa	Penyakit kerusakan mekanis hama	Hama belalang	Tungro	Penyakit Blas	Hama
47	Blast	Penyakit blas	Blast	Blas daun	Penyakit Blas	Defisiensi
48	Tungro	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Defisiensi
49	Blast	Penyakit blas	Blast	Blas daun	Penyakit Blas	Defisiensi
50	Bercak daun	Brown leaf spot	Hawar daun	Bercak daun	Penyakit Blas	Fungi

Tahap 2

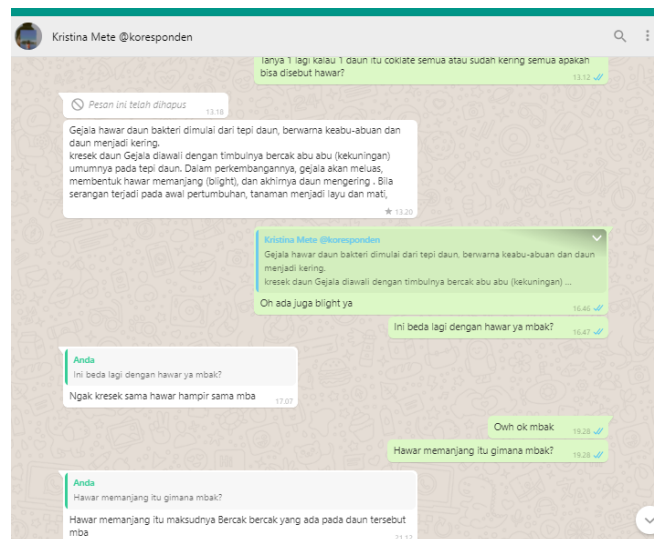
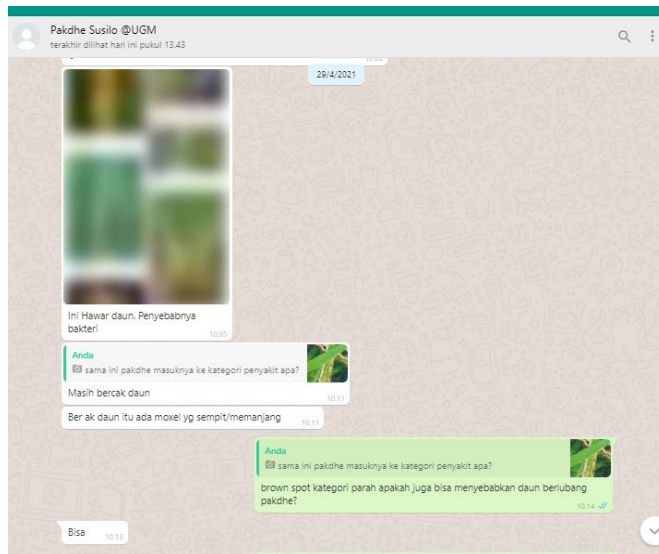
Nama Lengkap		Dzurriyatul Millah	Lufi Indriyani	Anisa Rahma Untari	Kristina Mete	Nursyamsih Taufik
Jurusan/Fakultas		Proteksi Tanaman	Agroteknologi/ Pertanian	pertanian	Kehutanan	Budidaya Pertanian/Pertanian
Nama Instansi/Kampus		IPB	Universitas Sarjanawiyata Tamansiswa	polbangtan yo-ma	Institut Pertanian	Balai Pengkajian Teknologi Pertanian Sulawesi barat/Universitas Hasanuddin
Tahun Lulus		2017	2019	2019	2020	2018
Keahlian dan Pengetahuan		Virology	Agronomi	tanaman pangan	Analisis berbagai vegetasi hutan kota	Budidaya Tanaman
Pertanyaan	1	Bercak cokelat	Brown spot	hawar	Hawar daun	Kresek
	2	Kresek	Brown spot	hawar	Kresek daun	Hawar daun
	3	Penyakit daun bergores bakteri	Brown spot	hawar	Hawar daun	Kresek
	4	Bercak cokelat sempit	Brown spot	brownspot	Narrow brown left spot	Brown spot
	5	Tungro RTBV	Brown spot	brownspot	Penyakit Bercak coklat daun	Brown spot
	6	Kerdil rumput	Brown spot	hawar	Penyakit blas	Kresek
	7	Bercak cokelat sempit	Brown spot	brownspot	Penyakit blas	Brown spot
	8	Tungro	Brown spot	hawar	Penyakit kresek daun	Hawar daun
	9	Bercak cokelat sempit	Brown spot	brownspot sempit	Penyakit cochliobolus miyabeanus	Brown spot
	10	Blas	Brown spot	blast	Penyakit kresek daun	Blas
	11	Bercak cokelat sempit	Brown spot	serangga, brownspot	Hama pengerek	Brown spot
	12	Gosong palsu	Brown spot	hawar	Penyakit hawar daun	Kresek
	13	Hawar seludang	Brown spot	hawar	Narrow Brown spot	Brown spot
	14	Stackburn	Brown spot	blast	Penyakit blas	Brown spot
	15	Bercak cokelat sempit	Brown spot	brownspot	Narrow Brown spot	Brown spot
	16	Stackburn	Brown spot	blast	Penyakit tungro	Brown spot
	17	Kresek	Brown spot	hawar	Penyakit hawar daun	Kresek
	18	Bercak cokelat sempit	Brown spot	serangga, brownspot	Penyakit Bercak coklat	Brown spot
	19	Blas	Brown spot	serangga	Hama penggerek	Brown spot
	20	Kresek	Brown spot	hawar	Hawar daun	Kresek
	21	Hawar bakteri	Brown spot	serangga	Wereng daun	Brown spot
	22	Blas	Brown spot	blast, hawar	Penyakit blas	Blas
	23	Blas	Brown spot	blast	Hawar daun	Blas
	24	Bercak cokelat sempit	Brown spot	brownspot	Narrow brown spot	Brown spot
	25	Bercak cokelat	Brown spot	blast	Hama wereng	Brown spot
	26	Hawar bakteri	Brown spot	serangga	Kresek daun	Blas
	27	Bercak cokelat sempit	Brown spot	brownspot	Hawar daun	Brown spot
	28	Kerdil kuning	Brown spot	hawar	Hawar daun	Kresek
	29	Blas	Brown spot	hawar, blast	Narrow brown spot	Hawar daun
	30	Bercak cokelat	Brown spot	blast	Hama penggerek	Blas

31	Blas	Brown spot	blast	Hama pengerek	Blas
32	Bercak coklat	Brown spot	blast	Hama penggerek	Brown spot
33	Blas	Brown spot	blast, hawar, serangga	Wereng daun	Brown spot
34	Bercak coklat	Brown spot	hawar	Penyakit kresek daun	Blas
35	Bercak coklat	Brown spot	hawar	Penyakit kresek daun	Kresek
36	Bercak coklat	Brown spot	hawar	Hama penggerek daun	Kresek dan blas
37	Kresek	Brown spot	hawar	Penyakit kresek daun	Kresek
38	Bercak coklat	Brown spot	blast	Hawar daun	Brown spot
39	Blas	Brown spot	hawar	Penyakit Bercak coklat	Kresek
40	Bercak coklat sempit	Brown spot	brownspot, hawar	Blas daun	Brown spot
41	Kresek	Brown spot	hawar	Kresek daun	Kresek
42	Bercak coklat	Brown spot	blast	Blas daun	Brown spot
43	Bercak coklat	Brown spot	blast	Penyakit hawar daun	Blas
44	Bercak coklat	Brown spot	hawar	Kresek daun	Brown spot
45	Bercak coklat	Brown spot	hawar	Narrow Brown spot	Blas
46	Bercak coklat	Brown spot	serangga, blast	Blas daun	Brown spot
47	Bercak coklat	Brown spot	serangga	Blas daun	Brown spot
48	Bercak coklat sempit	Brown spot	serangga	Hama penggerek	Kresek
49	Bercak coklat	Brown spot	hawar	Hawar daun	Blas
50	Bercak coklat	Brown spot	blast	Hawar daun	Blas
51	Bercak coklat sempit	Brown spot	brownspot	Wereng daun	Brown spot
52	Bercak coklat sempit	Brown spot	serangga	Hama penggerek	Brown spot
53	Bercak coklat sempit	Brown spot	brownspot	Hama penggerek	Brown spot
54	Bercak cokelat	Brown spot	hawar	Blas daun	Kresek
55	Bercak coklat sempit	Brown spot	brownspot	Hama penggerek	Brown spot
56	Kerdil kuning	Brown spot	serangga	Hama penggerek	Hawar daun
57	Kresek	Brown spot	hawar	Hawar daun	Kresek
58	Bercak coklat	Brown spot	brownspot	Narrow brown spot	Brown spot
59	Bercak coklat	Brown spot	brownspot	Narrow brown spot	Brown spot
60	Bercak coklat	Brown spot	blast	Brown cochliobolus	Blas
61	Bercak coklat	Brown spot	blast	Brown cochliobolus	Blas
62	Bercak coklat	Brown spot	serangga	Bercak coklat wereng	Brown spot
63	Kresek	Brown spot	hawar	Wereng daun	Hawar daun
64	Bercak coklat	Brown spot	hawar	Hawar daun	Kresek dan brown spot
65	Tungro	Brown spot	hawar	Hama penggerek	Brown spot
66	Tungro	Brown spot	hawar	Hama penggerek	Brown spot
67	Kresek	Brown spot	hawar	Brown spot	Hawar daun
68	Bercak coklat sempit	Brown spot	brownspot	Chochliobolus miyabeanus	Brown spot
69	Kerdil kuning	Brown spot	hawar	Chochliobolus miyabeanus	Hawar daun dan brown spot
70	Bercak coklat sempit	Brown spot	hawar	Chochliobolus miyabeanus	Kresek

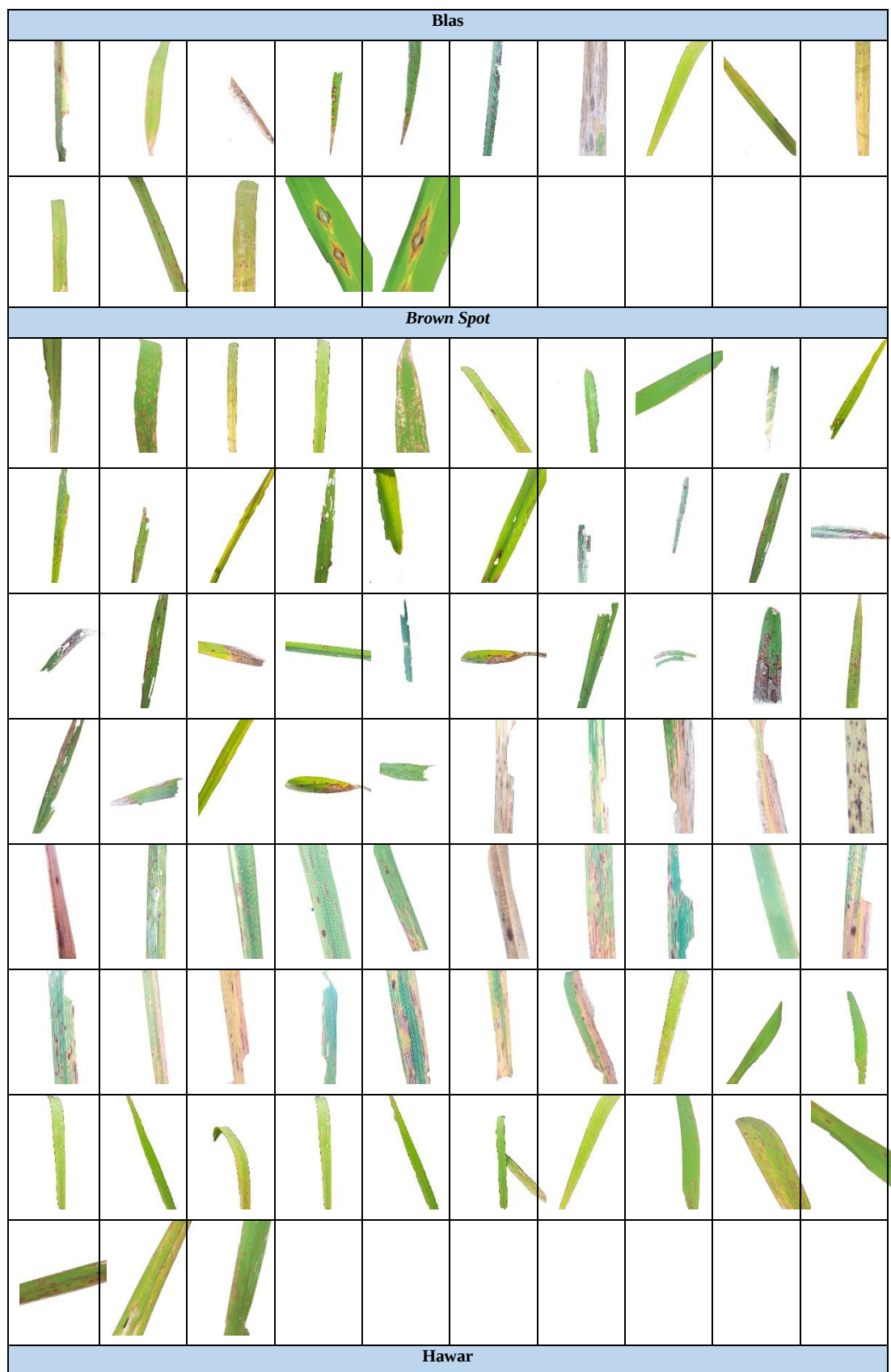
71	Tungro	Brown spot	brownspot	Wereng daun	Bulai dan Brown spot
72	Kresek	Brown spot	hawar	Brown left spor	Hawar daun
73	Kresek	Brown spot	hawar	Wereng daun	Kresek
74	Bercak coklat	Brown spot	brownspot	Hama penggerek	Brown spot
75	Tungro	Brown spot	hawar	Hama penggerek	Kresek
76	Bercak coklat	Brown spot	blast	Narrow brown spot	Brown spot
77	Bercak coklat sempit	Brown spot	brownspot	Blas daun	Brown spot
78	Tungro	Brown spot	hawar	Blas daun	Brown spot
79	Tungro	Brown spot	hawar	Hama penggerek	Hawar daun
80	Tungro	Brown spot	hawar	Hama penggerek	Brown spot
81	Bercak coklat sempit	Brown spot	brownspot	Chochliobolus miyabeanus	Brown spot
82	Kresek	Brown spot	hawar	Wereng daun	Hawar daun
83	Kresek	Brown spot	hawar	Hawas daun	Hawar daun
84	Bercak coklat sempit	Brown spot	brownspot	Chochliobolus miyabeanus	Brown spot
85	Kresek	Brown spot	hawar	Kresek daun	Kresek
86	Bercak coklat sempit	Brown spot	brownspot	Chochliobolus miyabeanus	Brown spot
87	Kresek	Brown spot	hawar	Chochliobolus miyabeanus	Hawar daun
88	Kerdil rumput	Brown spot	hawar	Hama penggerek	Hawar daun
89	Hawar	Brown spot	brownspot	Wereng daun	Kresek
90	Hawar	Brown spot	hawar	Chochliobolus miyabeanus	Kresek
91	Blas	Brown spot	blast	Kresek daun	Kresek
92	Bercak coklat sempit	Brown spot	brownspot	Hama penggerek	Brown spot
93	Bercak coklat sempit	Brown spot	hawar	Hama penggerek	Hawar daun
94	Bercak coklat sempit	Brown spot	brownspot	Chochliobolus miyabeanus	Brown spot
95	Bercak coklat sempit	Brown spot	hawar	Kresek daun	Brown spot
96	Bercak coklat	Brown spot	brownspot	Wereng daun	Brown spot
97	Bercak coklat	Brown spot	brownspot	Wereng daun	Brown spot
98	Bercak coklat sempit	Brown spot	serangga	Hama penggerek	Brown spot
99	Bercak coklat	Brown spot	brownspot	Chochliobolus miyabeanus	Brown spot
100	Bercak coklat	Brown spot	brownspot	Chochliobolus miyabeanus	Brown spot
101	Kresek	Brown spot	hawar	Kresek daun	Brown spot
102	Bercak coklat	Brown spot	hawar	Kresek daun	Brown spot
103	Hawar	Brown spot	hawar	Chochliobolus miyabeanus	Brown spot
104	Hawar	Brown spot	brownspot	Blas daun	Brown spot
105	Kerdil kuning	Brown spot	hawar	Kresek daun	Hawar daun
106	Tungro	Brown spot	hawar	Kresek daun	Hawar daun
107	Kresek	Brown spot	hawar	Kresek daun	Hawar daun
108	Daun bergores bakteri	Brown spot	hawar	Narrow brown spot	Hawar daun
109	Bercak coklat	Brown spot	blast hawar	Wereng daun	Brown spot

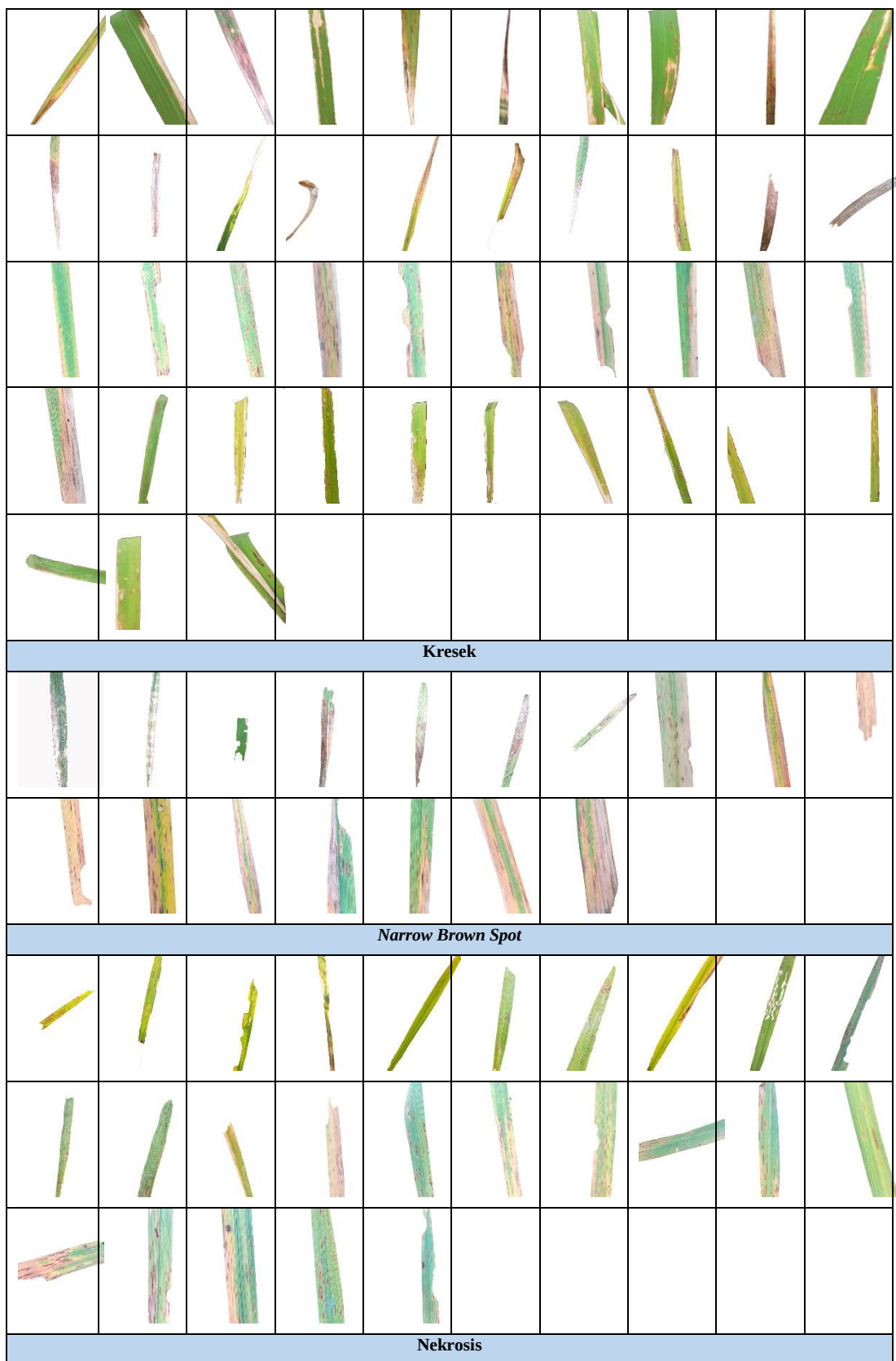
110	Bercak coklat	Brown spot	brownspot	Hama penggerek	Brown spot
111	Bercak coklat	Brown spot	hawar	Wereng daun	Brown spot
112	Kerdil kuning	Brown spot	hawar	Wereng daun	Brown spot
113	Hawar	Brown spot	hawar	Kresek daun	Brown spot
114	Tungro	Brown spot	hawar	Kresek daun	Hawar daun
115	Bercak coklat	Brown spot	brownspot,	Kresek daun	Brown spot
116	Hawar	Brown spot	blast	Wereng daun	Brown spot
117	Hawar	Brown spot	hawar	Kresek daun	Brown spot dan hawar daun
118	Tungro	Brown spot	hawar	Wereng daun	Brown spot
119	Bercak coklat	Brown spot	hawar	Hama penggerek	Brown spot

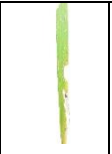
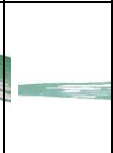
Lampiran 3. Konfirmasi ke Ahli



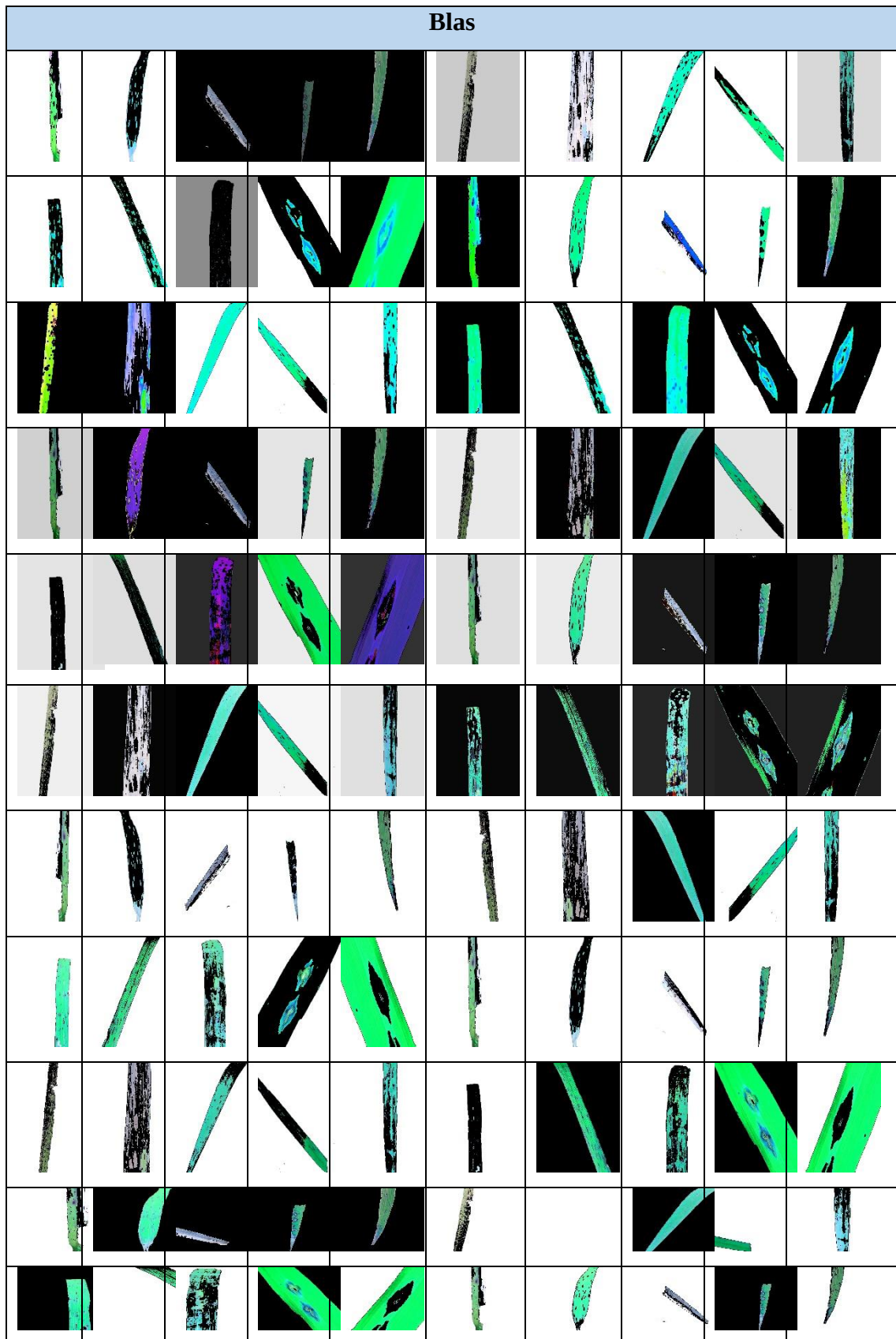
Lampiran 4. Citra Hasil Akuisisi

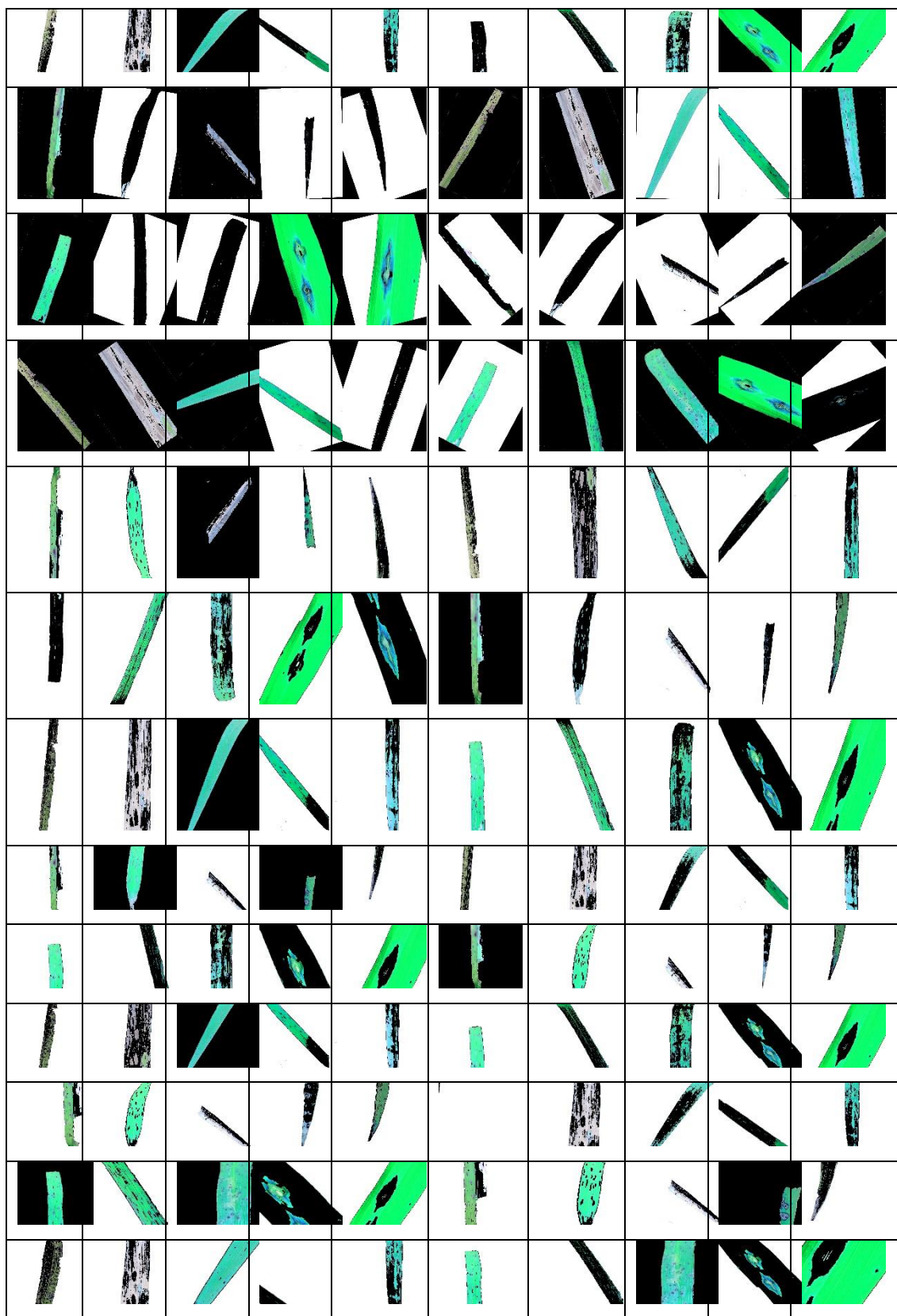


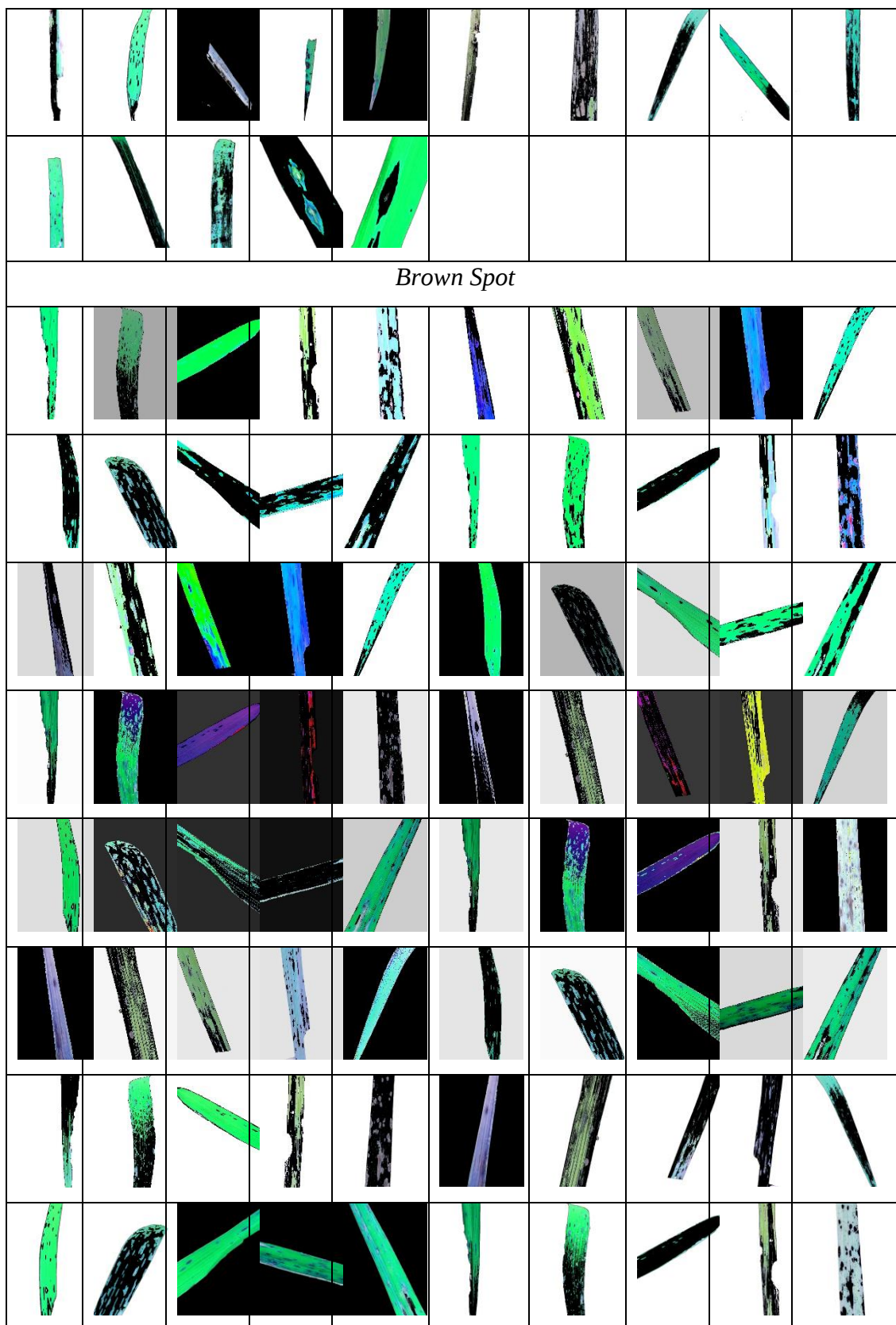


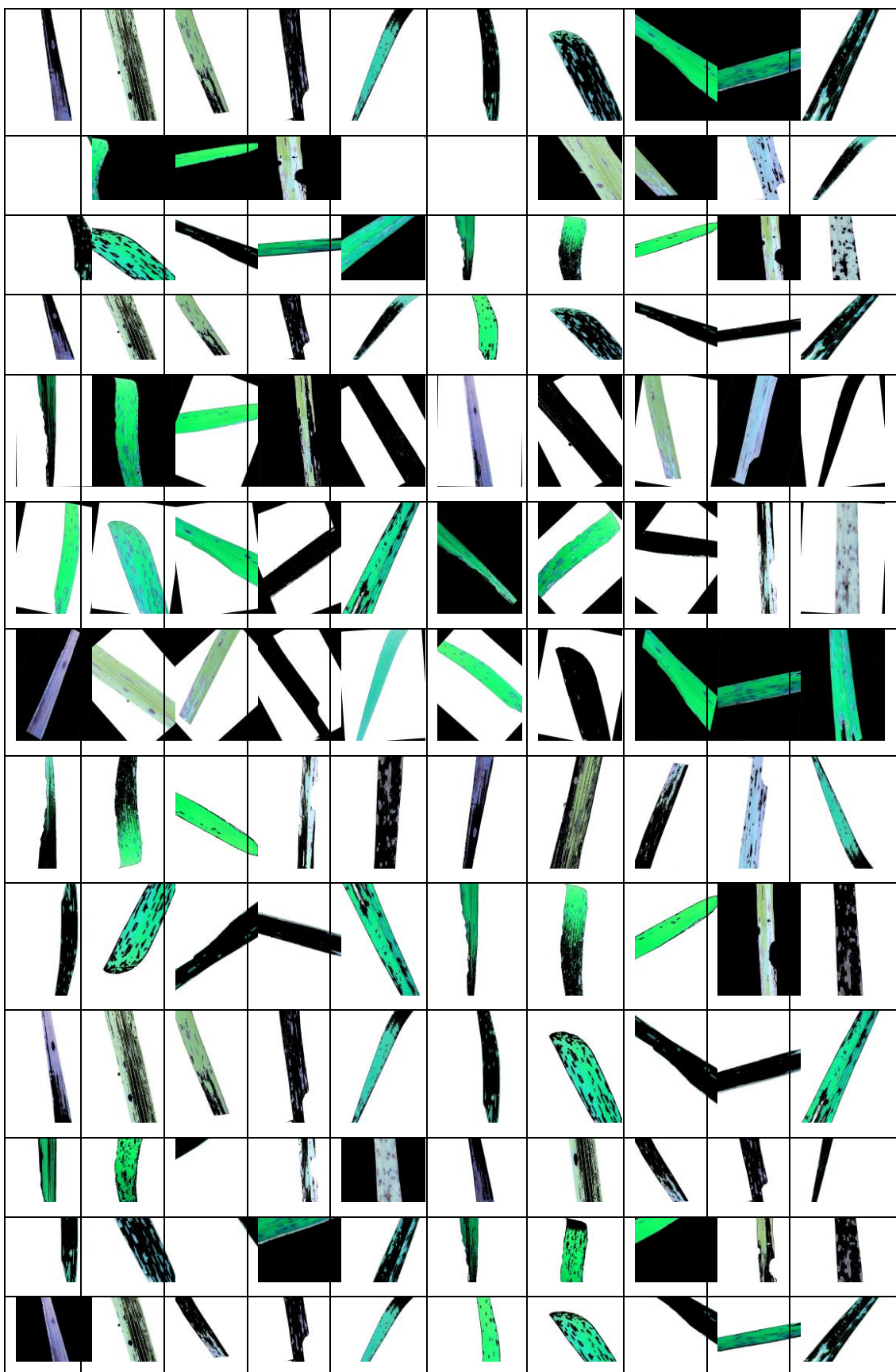
									
Putih Palsu									
									
									

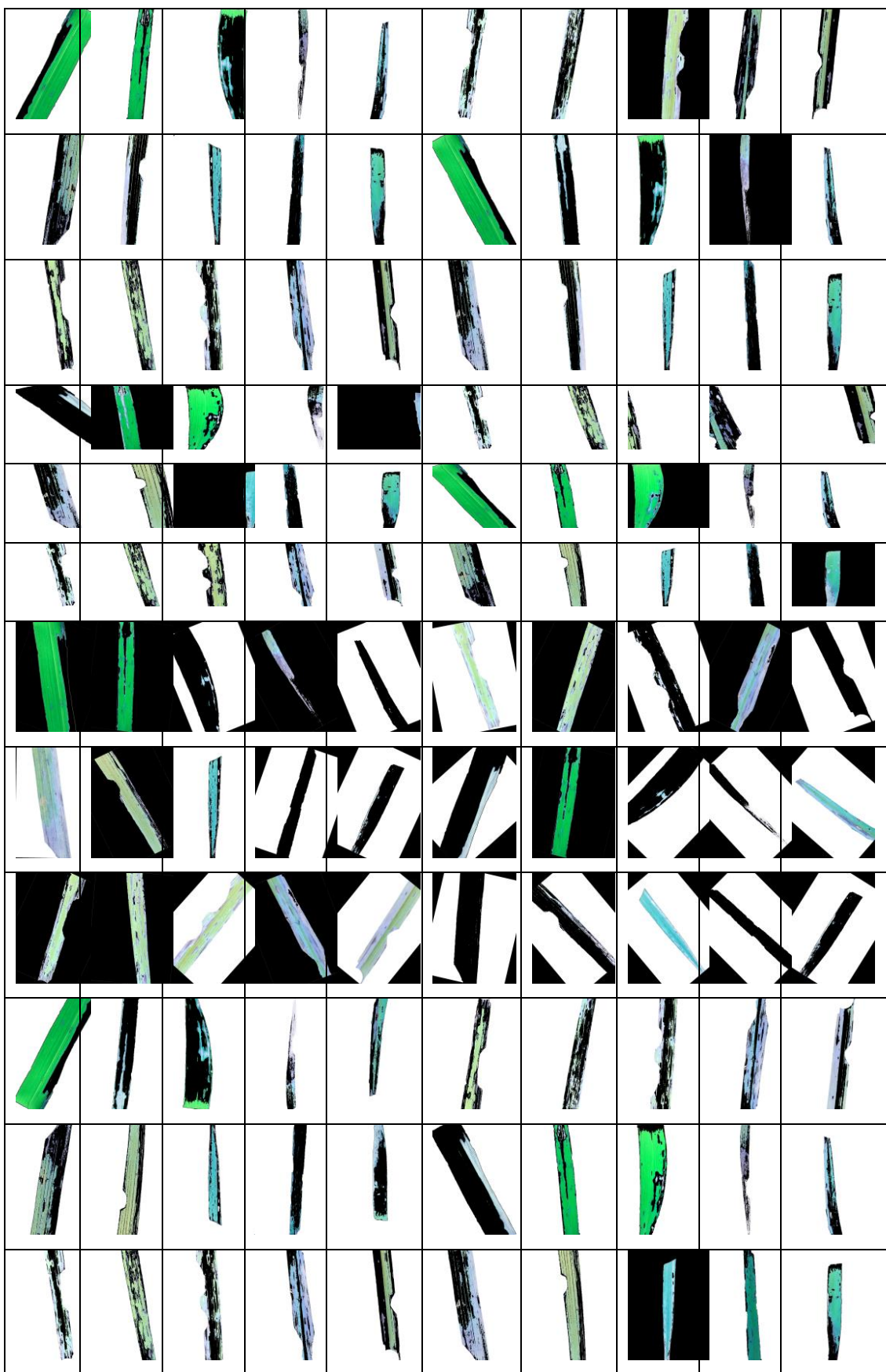
Lampiran 5. Dataset Penyakit Padi Hasil *Augmentation* dan Segmentasi

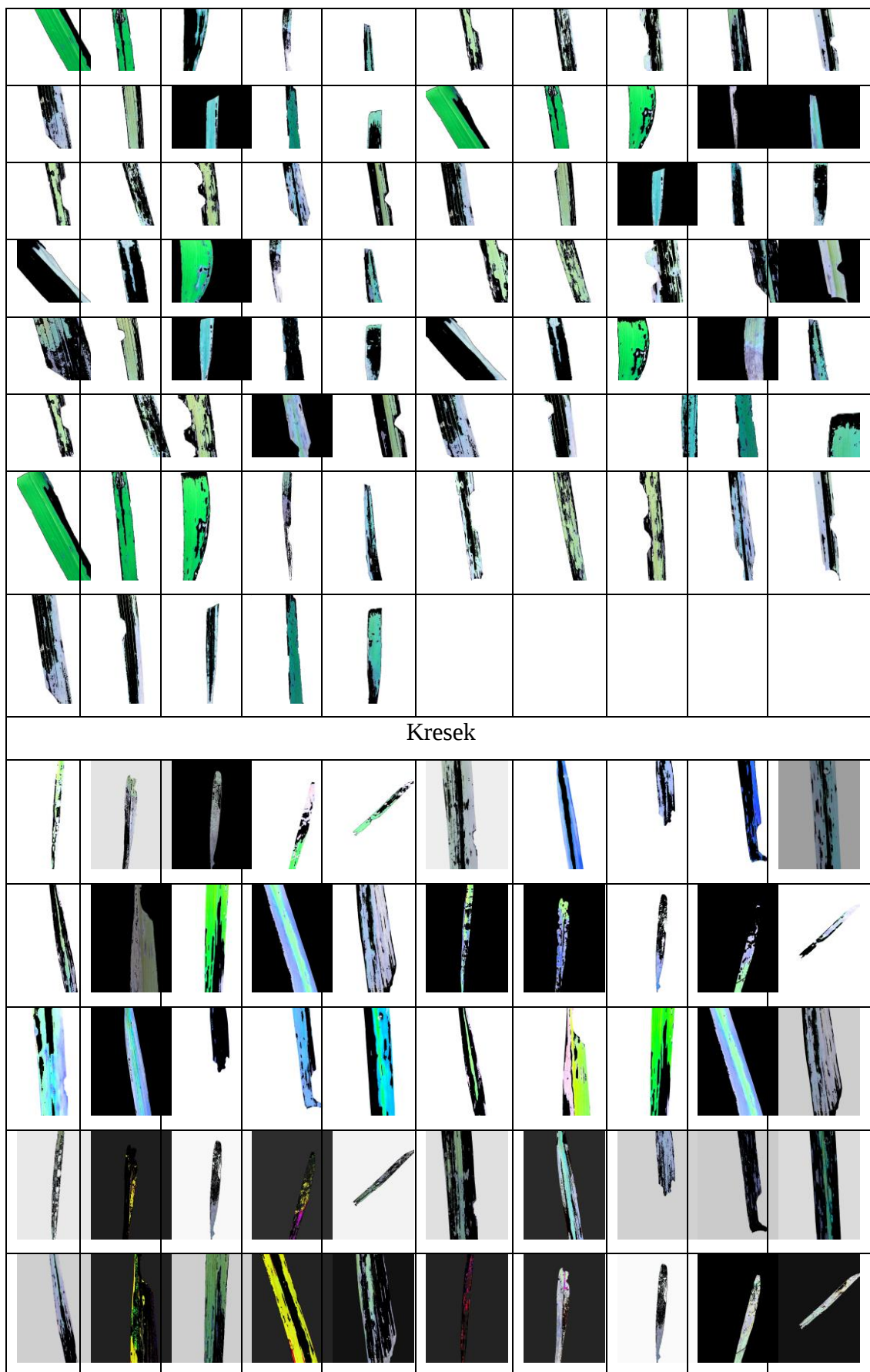


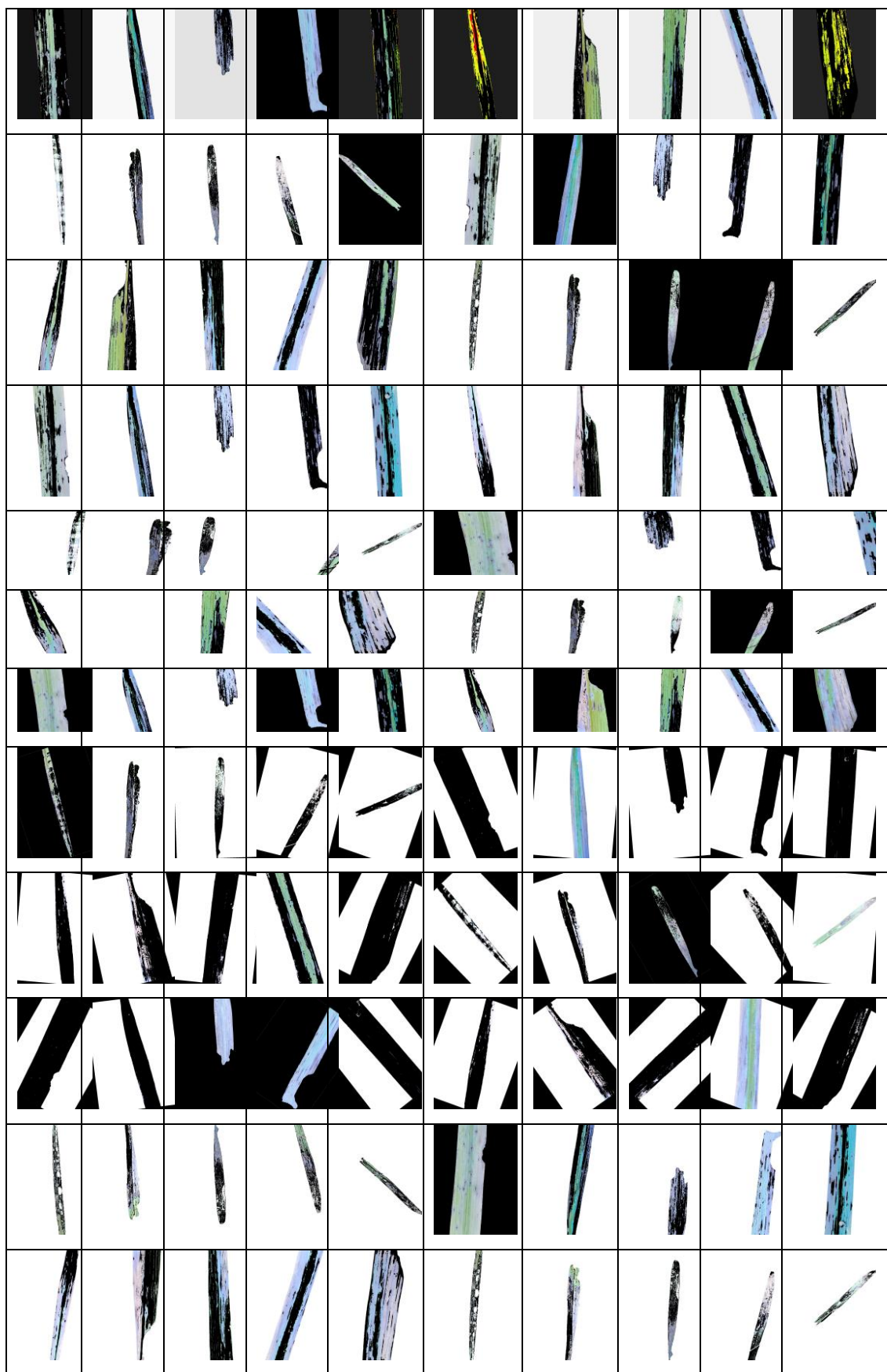


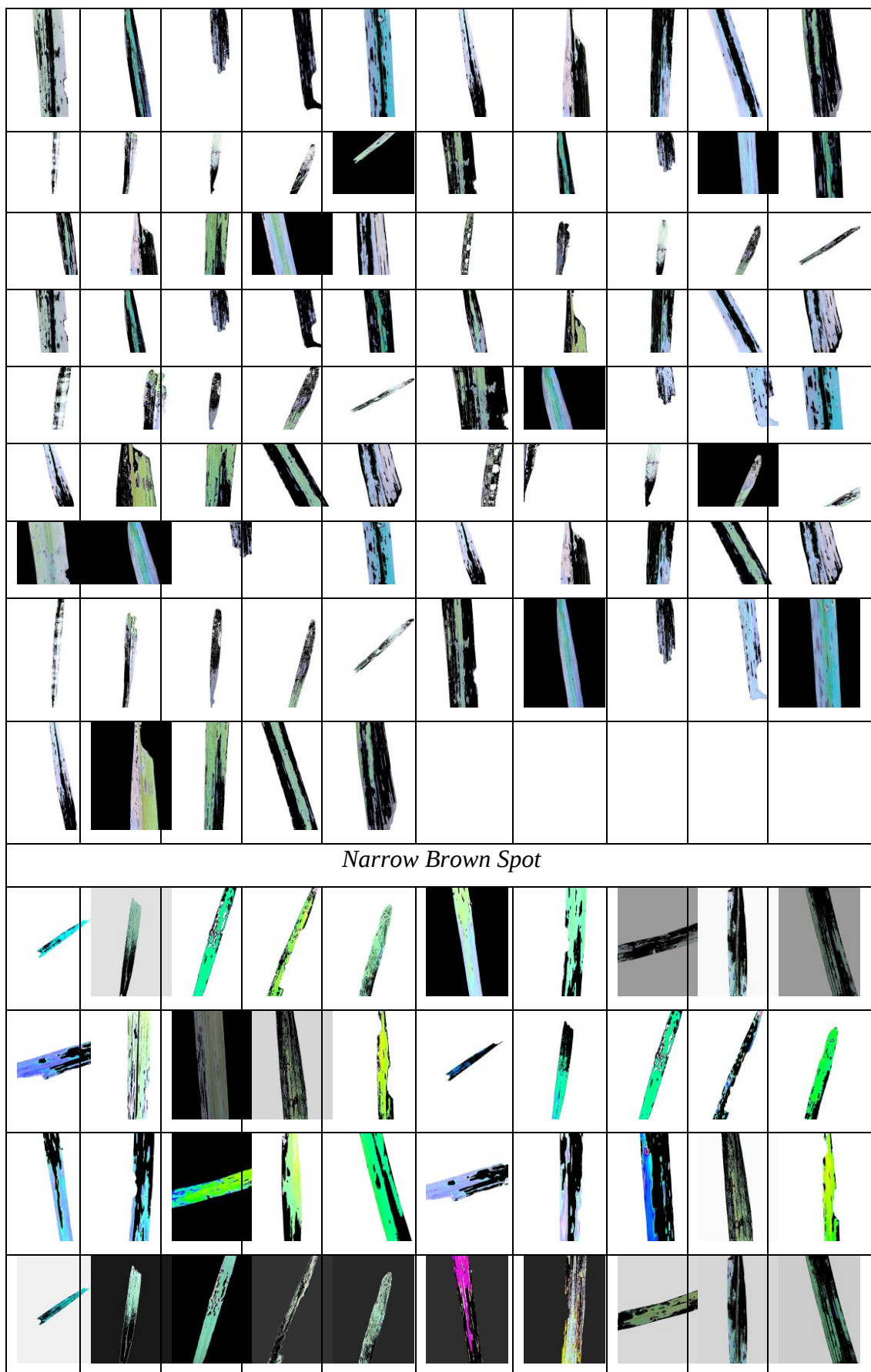


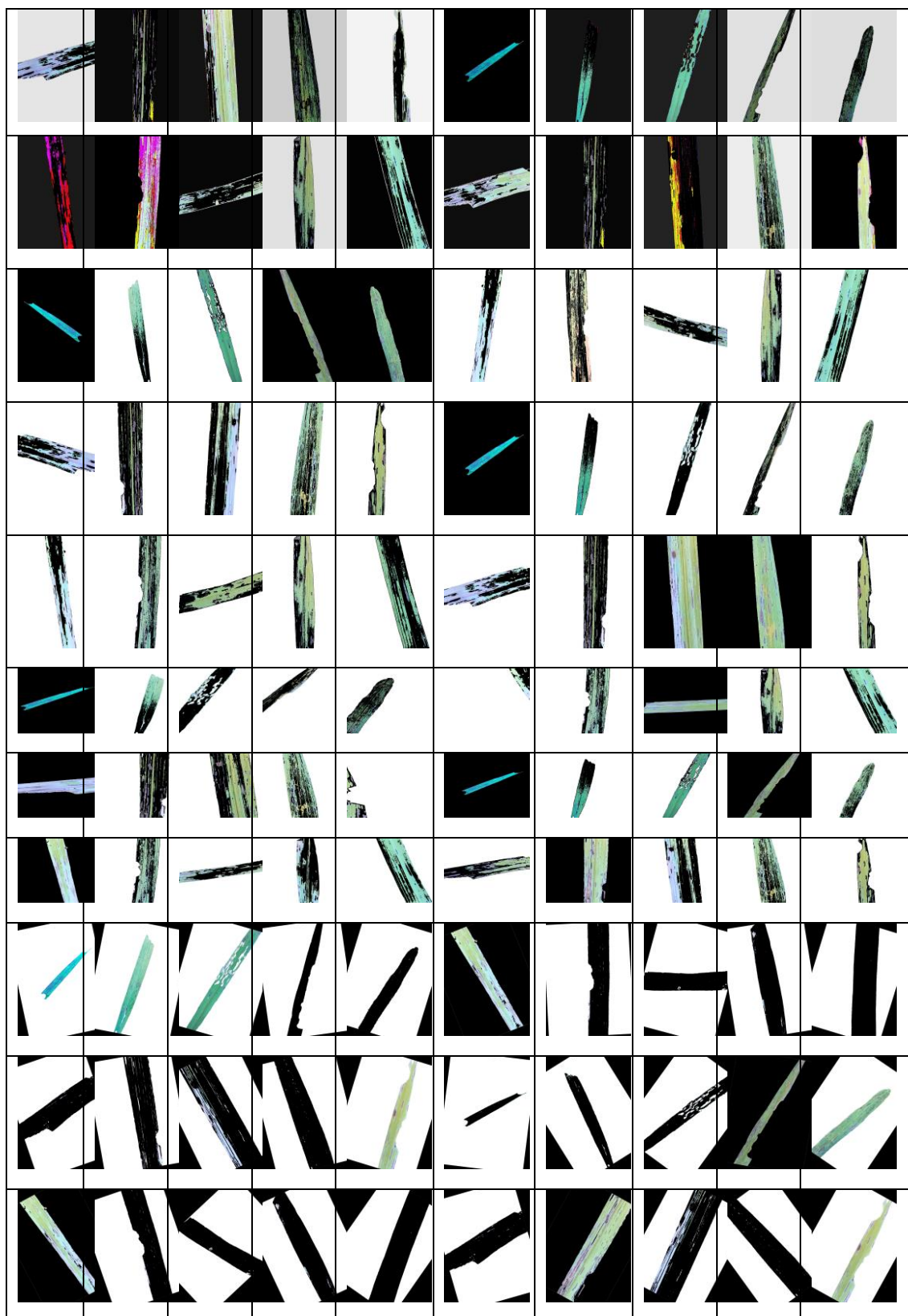


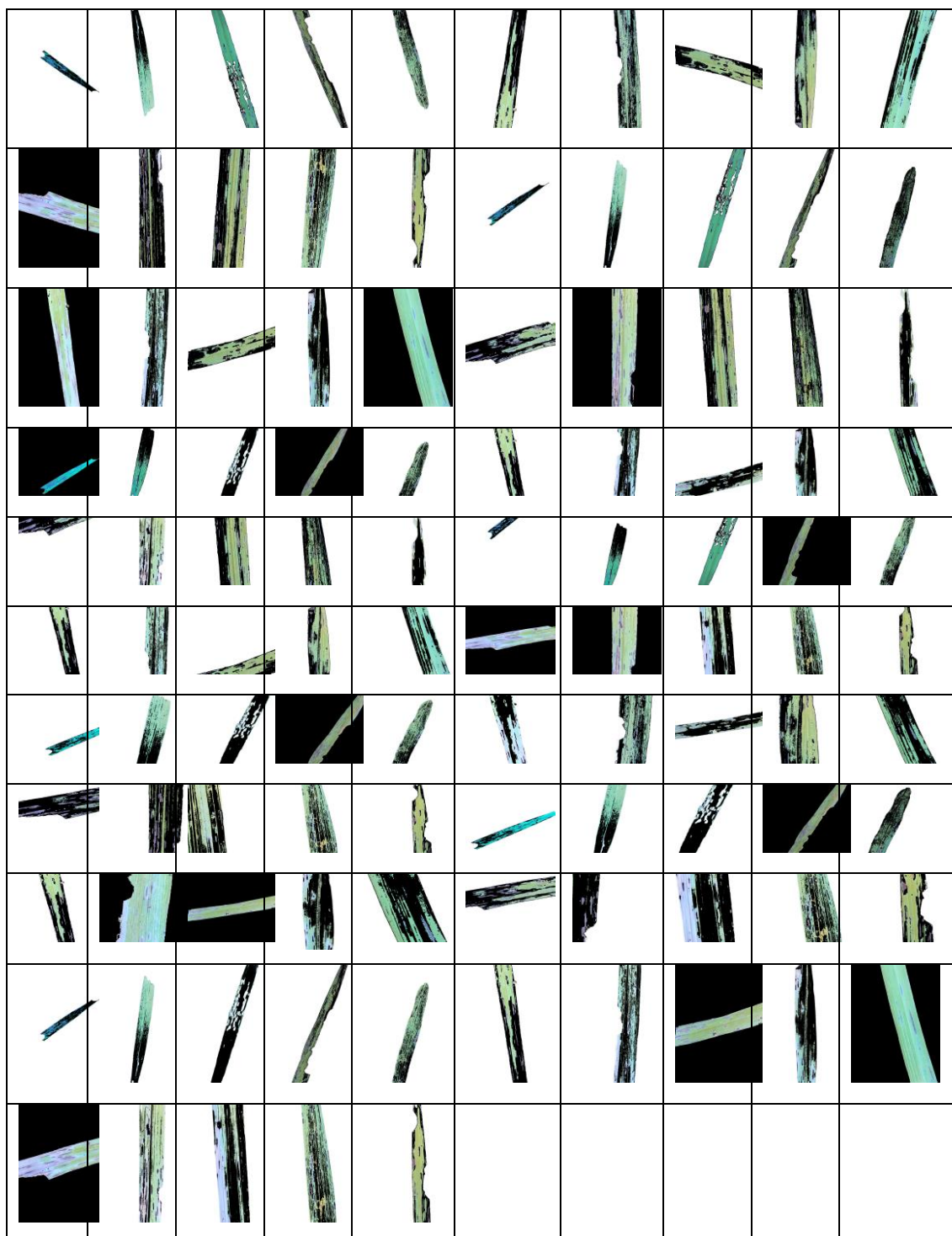












Lampiran 6. Tabel Hasil Pengujian Klasifikasi

Data Testing	Jenis Penyakit	Hasil Klasifikasi	Keterangan
1	Blas	Blas	True
2	Blas	Blas	True
3	Blas	Blas	True
4	Blas	Blas	True
5	Blas	Blas	True
6	Blas	Blas	True
7	Blas	Blas	True
8	Blas	Blas	True
9	Blas	Blas	True
10	Blas	Blas	True
11	Blas	Blas	True
12	Blas	Blas	True
13	Blas	Blas	True
14	Blas	Blas	True
15	Blas	Blas	True
16	Blas	Blas	True
17	Blas	Blas	True
18	Blas	Blas	True
19	Blas	Blas	True
20	Blas	Blas	True
21	Blas	Blas	True
22	Blas	Blas	True
23	Blas	Blas	True
24	Blas	Blas	True
25	Blas	Hawar	False
26	Blas	Kresek	False
27	Blas	Narrow Brown Spot	False
28	Brown Spot	Brown Spot	True
29	Brown Spot	Brown Spot	True
30	Brown Spot	Brown Spot	True
31	Brown Spot	Brown Spot	True
32	Brown Spot	Brown Spot	True
33	Brown Spot	Brown Spot	True
34	Brown Spot	Brown Spot	True
35	Brown Spot	Brown Spot	True
36	Brown Spot	Brown Spot	True
37	Brown Spot	Brown Spot	True
38	Brown Spot	Brown Spot	True
39	Brown Spot	Brown Spot	True
40	Brown Spot	Brown Spot	True
41	Brown Spot	Blas	False
42	Brown Spot	Blas	False
43	Brown Spot	Hawar	False
44	Hawar	Hawar	True
45	Hawar	Hawar	True
46	Hawar	Hawar	True
47	Hawar	Hawar	True
48	Hawar	Hawar	True
49	Hawar	Hawar	True
50	Hawar	Hawar	True
51	Hawar	Hawar	True
52	Hawar	Hawar	True
53	Hawar	Hawar	True
54	Hawar	Hawar	True
55	Hawar	Hawar	True
56	Hawar	Hawar	True
57	Hawar	Hawar	True
58	Hawar	Hawar	True
59	Hawar	Hawar	True
60	Hawar	Hawar	True
61	Hawar	Hawar	True

Data Testing	Jenis Penyakit	Hasil Klasifikasi	Keterangan
62	Hawar	Hawar	True
63	Hawar	Hawar	True
64	Hawar	Hawar	True
65	Hawar	Hawar	True
66	Hawar	Hawar	True
67	Hawar	Hawar	True
68	Hawar	Hawar	True
69	Hawar	Hawar	True
70	Kresek	Kresek	True
71	Kresek	Kresek	True
72	Kresek	Kresek	True
73	Kresek	Kresek	True
74	Kresek	Kresek	True
75	Kresek	Kresek	True
76	Kresek	Kresek	True
77	Kresek	Kresek	True
78	Kresek	Kresek	True
79	Kresek	Kresek	True
80	Kresek	Kresek	True
81	Kresek	Kresek	True
82	Kresek	Kresek	True
83	Kresek	Kresek	True
84	Kresek	Kresek	True
85	Kresek	Kresek	True
86	Kresek	Kresek	True
87	Kresek	Kresek	True
88	Kresek	Kresek	True
89	Kresek	Kresek	True
90	Kresek	Kresek	True
91	Kresek	Kresek	True
92	Kresek	Kresek	True
93	Kresek	Kresek	True
94	Kresek	Kresek	True
95	Kresek	Kresek	True
96	Kresek	Kresek	True
97	Kresek	Kresek	True
98	Kresek	Kresek	True
99	Kresek	Kresek	True
100	Kresek	Narrow Brown Spot	False
101	Narrow Brown Spot	Narrow Brown Spot	True
102	Narrow Brown Spot	Narrow Brown Spot	True
103	Narrow Brown Spot	Narrow Brown Spot	True
104	Narrow Brown Spot	Narrow Brown Spot	True
105	Narrow Brown Spot	Narrow Brown Spot	True
106	Narrow Brown Spot	Narrow Brown Spot	True
107	Narrow Brown Spot	Narrow Brown Spot	True
108	Narrow Brown Spot	Narrow Brown Spot	True
109	Narrow Brown Spot	Narrow Brown Spot	True
110	Narrow Brown Spot	Narrow Brown Spot	True
111	Narrow Brown Spot	Narrow Brown Spot	True

Data Testing	Jenis Penyakit	Hasil Klasifikasi	Keterangan
112	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
113	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
114	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
115	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
116	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
117	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
118	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
119	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
120	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
121	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
122	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
123	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
124	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
125	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
126	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
127	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>
128	<i>Narrow Brown Spot</i>	<i>Narrow Brown Spot</i>	<i>True</i>

Lampiran 7. Source Code Program

```
[ ] # Imports
import numpy as np # linear algebra
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import cv2
import sklearn
import random
import tensorflow as tf
import os
from tqdm import tqdm
from tensorflow.keras.preprocessing.image import ImageDataGenerator

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

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

[ ] base_path = 'drive/MyDrive/Tesis Sukses'
labels = ['Blas', 'Brown Spot', 'Hauwer', 'kresek', 'Narrow Brown Spot']

[ ] # Constants
IMAGE_SIZE = 224
BATCH_SIZE = 32
VERBOSE = 1

[ ] # Dataset from the training folder.
X_train=[]
y_train=[]
for i in labels:
    folderPath = os.path.join(os.path.join(base_path, 'Datasetku'), i)
    for j in tqdm(os.listdir(folderPath)):
        img = cv2.imread(os.path.join(folderPath, j))
        img = cv2.resize(img, (IMAGE_SIZE, IMAGE_SIZE))
        X_train.append(img)
        y_train.append(i)
print('Training dataset loading complete.')

# Dataset from the testing folder..
for i in labels:
    folderPath = os.path.join(os.path.join(base_path, 'segmentasi_train'), i)
    for j in tqdm(os.listdir(folderPath)):
        img = cv2.imread(os.path.join(folderPath, j))
        img = cv2.resize(img, (IMAGE_SIZE, IMAGE_SIZE))
        X_train.append(img)
        y_train.append(i)
X_train = np.array(X_train)
y_train = np.array(y_train)
X_train, y_train = sklearn.utils.shuffle(X_train, y_train, random_state=0)
print('Testing dataset Loading complete.')

[ ] 100%|#####| 240/240 [00:02:00:00, 93.24it/s]
100%|#####| 240/240 [00:02:00:00, 94.62it/s]
100%|#####| 240/240 [00:02:00:00, 97.39it/s]
100%|#####| 240/240 [00:02:00:00, 96.07it/s]
100%|#####| 240/240 [00:02:00:00, 95.85it/s]
100%|#####| 15/15 [00:00:00:00, 96.36it/s]
0%|#####| 0/15 [00:00:00:00, 71it/s] Training dataset loading complete.
100%|#####| 15/15 [00:00:00:00, 97.22it/s]
100%|#####| 15/15 [00:00:00:00, 93.01it/s]
100%|#####| 15/15 [00:00:00:00, 95.91it/s]
100%|#####| 15/15 [00:00:00:00, 98.14it/s]
Testing dataset Loading complete.

[ ] sns.countplot(y_train)
plt.show()

FutureWarning
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arg

[ ]
250
200
150
100
50
0
Hauwer Brown Spot Blas kresek Narrow Brown Spot

[ ]
[ ] j=0
for i in labels:
    j=0
    while True:
        if y_train[j]==i:
            plt.figure(figsize=(5,5))
            plt.imshow(X_train[j])
            plt.title(y_train[j])
            break
        j+=1

[ ]
0
25
50
75
Blas
```

```
[ ] X_train.shape
(1275, 224, 224, 3)

[ ] # splitting of dataset for Testing.
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X_train, y_train, test_size = 0.10, random_state=0)
print("Shapes X : Train :", X_train.shape, " Test :", X_test.shape)
print("Shapes Y : Train :", y_train.shape, " Test :", y_test.shape)
Shapes X : Train : (1147, 224, 224, 3) Test : (128, 224, 224, 3)
Shapes Y : Train : (1147,) Test : (128,)

[ ] y_train_new = []
for i in y_train:
    y_train_new.append(labels.index(i))
y_train = y_train_new
y_train = tf.keras.utils.to_categorical(y_train)

y_test_new = []
for i in y_test:
    y_test_new.append(labels.index(i))
y_test = y_test_new
y_test = tf.keras.utils.to_categorical(y_test)
```

EfficientNetB4

```
[ ] # Importing the model and initializing.
from tensorflow.keras.applications import EfficientNetB4
effnet = EfficientNetB4(weights = 'imagenet', include_top=False, input_shape=(IMAGE_SIZE, IMAGE_SIZE, 3))

[ ] # Adding Top Layers.
model1 = effnet.output
model1 = tf.keras.layers.GlobalAveragePooling2D()(model1)
model1 = tf.keras.layers.Dropout(0.5)(model1)
model1 = tf.keras.layers.Dense(s, activation = 'softmax')(model1)
model1 = tf.keras.models.Model(inputs = effnet.input, outputs = model1)

[ ] # Model summary.
model1.summary()

Model: "model_1"
Layer (type) Output Shape Param # Connected to
-----
input_2 (InputLayer) [(None, 224, 224, 3)] 0
rescaling_1 (Rescaling) (None, 224, 224, 3) 0 input_2[0][0]
normalization_1 (Normalization) (None, 224, 224, 3) 7 rescaling_1[0][0]
stem_conv_pad (ZeroPadding2D) (None, 225, 225, 3) 0 normalization_1[0][0]
stem_conv (Conv2D) (None, 112, 112, 48) 1296 stem_conv_pad[0][0]
stem_bn (BatchNormalization) (None, 112, 112, 48) 192 stem_conv[0][0]
stem_activation (Activation) (None, 112, 112, 48) 0 stem_bn[0][0]
block1a_dwconv (DepthwiseConv2D) (None, 112, 112, 48) 432 stem_activation[0][0]
block1a_bn (BatchNormalization) (None, 112, 112, 48) 192 block1a_dwconv[0][0]
```

```
[ ] # Compiling Model
model1.compile(loss = 'categorical_crossentropy', optimizer = 'Adam', metrics=['accuracy'])

[ ] # Callbacks.
from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau, TensorBoard, ModelCheckpoint
tensorboard = TensorBoard(log_dir = 'logs')
checkpoint = ModelCheckpoint("effnet.hs", monitor="val_accuracy", save_best_only=True, mode="auto", verbose=1)
reduce_lr = ReduceLROnPlateau(monitor = 'val_accuracy', factor = 0.3, patience = 3, min_delta = 0.001, mode='auto', verbose=VERBOSE)

[ ] # Fitting the model
history1 = model1.fit(x_train, y_train, validation_split=0.1, epochs = 25, verbose=VERBOSE, batch_size=BATCH_SIZE, callbacks=[tensorboard, checkpoint, reduce_lr])

/usr/local/lib/python3.7/dist-packages/tensorflow/python/keras/utils/generic_utils.py:497: CustomMaskLearning: Custom mask layers require a config and must override get_config. When loading, the custom mask layer must t
category: customMaskLearning) - 455 893ms/step - loss: 1.1599 - accuracy: 0.5620 - val_loss: 1.3129 - val_accuracy: 0.5304
Epoch 1/25
33/33 [=====] - 455 893ms/step - loss: 1.1599 - accuracy: 0.5620 - val_loss: 1.3129 - val_accuracy: 0.5304
Epoch 00001: val_accuracy improved from -inf to 0.53043, saving model to effnet.hs
Epoch 2/25
33/33 [=====] - 245 737ms/step - loss: 0.5466 - accuracy: 0.8023 - val_loss: 2.4887 - val_accuracy: 0.6870
Epoch 00002: val_accuracy improved from 0.53043 to 0.68696, saving model to effnet.hs
Epoch 3/25
33/33 [=====] - 245 734ms/step - loss: 0.3639 - accuracy: 0.8857 - val_loss: 0.6793 - val_accuracy: 0.8087
Epoch 00003: val_accuracy improved from 0.68696 to 0.80870, saving model to effnet.hs
Epoch 4/25
33/33 [=====] - 245 726ms/step - loss: 0.4039 - accuracy: 0.8769 - val_loss: 0.3086 - val_accuracy: 0.8370
Epoch 00004: val_accuracy improved from 0.80870 to 0.88696, saving model to effnet.hs
Epoch 5/25
33/33 [=====] - 245 737ms/step - loss: 0.2775 - accuracy: 0.9021 - val_loss: 0.7218 - val_accuracy: 0.8174
Epoch 00005: val_accuracy did not improve from 0.88696
Epoch 6/25
33/33 [=====] - 245 729ms/step - loss: 0.2625 - accuracy: 0.9099 - val_loss: 0.4744 - val_accuracy: 0.8522
Epoch 00006: val_accuracy did not improve from 0.88696
```

```
[ ] epochs = [i for i in range(25)]
fig, ax = plt.subplots(1,2)
train_acc = history1.history['accuracy']
train_loss = history1.history['loss']
val_acc = history1.history['val_accuracy']
val_loss = history1.history['val_loss']
fig.set_size_inches(17, 8)

ax[0].plot(epochs, train_acc, 'go-', label = 'Training Accuracy')
ax[0].plot(epochs, val_acc, 'r-o-', label = 'Validation Accuracy')
ax[0].set_title('Training & Validation Accuracy')
ax[0].legend()
ax[0].set_xlabel('Epochs')
ax[0].set_ylabel('Accuracy')

ax[1].plot(epochs, train_loss, 'g-o-', label = 'Training Loss')
ax[1].plot(epochs, val_loss, 'r-o-', label = 'Validation Loss')
ax[1].set_title('Testing Accuracy & Loss')
ax[1].legend()
ax[1].set_xlabel('Epochs')
ax[1].set_ylabel('Training & Validation Loss')
plt.show()
```

```
[ ] # Evaluating Model.
result1=model1.evaluate(x_test, y_test)
print("Testing Loss :", result[0])
print("Testing Accuracy :", result[1]*100, "%")

4/4 [=====] - 35 147ms/step - loss: 0.1919 - accuracy: 0.9453
Testing Loss : 0.1919262111869812
Testing Accuracy : 94.53125 %

[ ] predictions1 = model1.predict(x_test)
predictions1 = np.argmax(predictions, axis = 1)
y_test_edit = np.argmax(y_test , axis = 1)

[ ] # Classification report.
cf_report = sklearn.metrics.classification_report(y_test_edit, predictions1 )
print(cf_report)
```

	precision	recall	f1-score	support
0	0.92	0.89	0.91	27
1	1.00	0.81	0.90	16
2	0.93	1.00	0.96	26
3	0.97	0.97	0.97	31
4	0.93	1.00	0.97	28
accuracy			0.95	128
micro avg	0.95	0.93	0.94	128
weighted avg	0.95	0.95	0.94	128

