

**PENERAPAN ARSITEKTUR *MOBILENETV2* PADA  
KLASIFIKASI PENYAKIT DAUN TEH**



TESIS

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

MONIKKA NUR WINNARTO

14002374

Program Studi Ilmu Komputer (S2)

Fakultas Teknologi Informasi

Universitas Nusa Mandiri

2021

**PENERAPAN ARSITEKTUR *MOBILENETV2* PADA  
KLASIFIKASI PENYAKIT DAUN TEH**



TESIS

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

MONIKKA NUR WINNARTO

14002374

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 : Monikka Nur Winnarto  
NIM : 14002374  
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: "Penerapan Arsitektur *MobileNetV2* Pada Klasifikasi Penyakit Daun Teh" 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.

Demikianlah 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, 01 Agustus 2021

Yang menyatakan,

A handwritten signature in black ink is written over a yellow 10,000 Rupiah Indonesian postage stamp. The stamp features the Garuda Pancasila emblem and the text '10000', 'REPUBLIK INDONESIA', and 'POSTERAI TEMPEL'. The serial number 'CEBF1AJX333750429' is visible at the bottom of the stamp.

Monikka Nur Winnarto

## HALAMAN PERSETUJUAN DAN PENGESAHAN TESIS

Tesis ini diajukan oleh :

Nama : Monikka Nur Winnarto  
NIM : 14002374  
Program Studi : Ilmu Komputer  
Jenjang : Strata Dua (S2)  
Konsentrasi : Image Processing  
Judul Tesis : “Penerapan Arsitektur *MobileNetV2* Pada Klasifikasi Penyakit Daun Teh”

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, 12 Agustus 2021

### PEMBIMBING TESIS

Pembimbing I : Dr. Dwina Kuswandari, M.Kom



Pembimbing II : Sukmawati Anggraeni Putri, M.Kom



### DEWAN PENGUJI

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



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

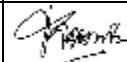
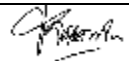
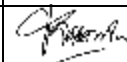
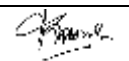


Penguji II/  
Pembimbing : Dr. Dwina Kuswandari, M.Kom



|                                                                                   |                                 |
|-----------------------------------------------------------------------------------|---------------------------------|
|  | <b>LEMBAR BIMBINGAN TESIS</b>   |
|                                                                                   | <b>UNIVERSITAS NUSA MANDIRI</b> |

NIM : 14002374  
 Nama Lengkap : Monikka Nur Winnarto  
 Dosen Pembimbing : Dr. Dra. Dwina Kuswandari, M.Kom  
 Judul Tesis : “Penerapan Arsitektur *MobileNetV2* Pada Klasifikasi Penyakit Daun Teh”

| NO | Tanggal Bimbingan | Pokok Bahasan                     | Paraf Dosen Pembimbing                                                                |
|----|-------------------|-----------------------------------|---------------------------------------------------------------------------------------|
| 1  | 12 April 2021     | Pendahuluan                       |    |
| 2  | 10 Mei 2021       | Pengajuan Bab I                   |    |
| 3  | 14 Juni 2021      | Revisi Bab I dan Pengajuan Bab II |   |
| 4  | 28 Juni 2021      | Pengajuan Bab III                 |  |
| 5  | 03 Juli 2021      | Revisi Bab III                    |  |
| 6  | 12 Juli 2021      | Pengajuan Bab IV dan V            |  |
| 7  | 18 Juli 2021      | Revisi Bab IV                     |  |
| 8  | 31 Juli 2021      | ACC Keseluruhan                   |  |

Catatan untuk dosen pembimbing

Bimbingan Tesis

- Dimulai pada tanggal : 12 April 2021
- Diakhiri pada tanggal : 31 Juli 2021
- Jumlah pertemuan bimbingan : 8 (Kali) Bimbingan

Disetujui oleh,  
**Dosen Pembimbing**



(Dr. Dra. Dwina Kuswandari, M.Kom)

## KATA PENGANTAR

Puji syukur alhamdulillah, penulis panjatkan kehadirat 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 “Penerapan Arsitektur *MobileNetV2* Pada Klasifikasi Penyakit Daun Teh”

Tujuan penulisan laporan tesis ini dibuat sebagai salah satu syarat untuk mendapatkan gelar Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri. Penulis menyadari bahwa tanpa bimbingan dan dukungan dari semua pihak dalam pembuatan laporan tesis ini, maka penulis tidak dapat menyelesaikan laporan tesis ini tepat pada waktunya. Untuk itu izinkanlah penulis pada kesempatan ini untuk mengucapkan terima kasih kepada :

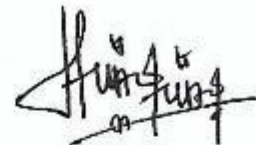
1. Rektor Universitas Nusa Mandiri.
2. Wakil Rektor I Bidang Akademik Universitas Nusa Mandiri.
3. Dekan Fakultas Teknologi Informasi Universitas Nusa Mandiri.
4. Ketua Program Studi Ilmu Komputer.
5. Dr. Dwina Kuswandari, M.Kom selaku pembimbing pertama tesis yang telah menyediakan waktu, pikiran dan tenaga dalam membimbing penulis dalam menyelesaikan tesis ini.
6. Sukmawati Anggraeni Putri, M.Kom selaku asisten pembimbing yang telah mengarahkan, memberikan masukan, menyumbangkan ide, waktu, serta tenaganya dalam membimbing penulis dalam menyelesaikan tesis ini.
7. Yayasan Bina Sarana Informatika yang telah memberi kesempatan kepada penulis untuk mendapatkan beasiswa prestasi hingga menyelesaikan studi sampai dengan S2.
8. Bapak Dedi Ruswandi selaku Administrasi I SDM dan Umum PT Perkebunan Nusantara VIII Kebun Gedeh Mas yang telah mengizinkan penulis melakukan riset untuk mendapatkan data atau informasi yang penulis butuhkan.

9. Orang tua tercinta yang telah memberikan dukungan material dan moral kepada penulis.
10. Imam Nurfi yang telah memberikan dukungan dan semangat kepada penulis.
11. Seluruh Dosen Program Studi Ilmu Komputer (S2) Universitas Nusa Mandiri yang telah memberikan pelajaran yang berarti bagi penulis selama menempuh studi.
12. Teman-teman seperjuangan yang selalu memberikan semangat dan dukungan yang tiada hentinya.

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

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

Jakarta, 01 Agustus 2021



Monikka Nur Winnarto

Penulis

## SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Yang bertanda tangan di bawah ini, saya :

Nama : Monikka Nur Winnarto  
NIM : 14002374  
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 izin 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 : “Penerapan Arsitektur *MobileNetV2* Pada Klasifikasi Penyakit Daun Teh” 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 izin 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, 01 Agustus 2021  
Yang menyatakan,



Monikka Nur Winnarto

## ABSTRAK

Nama : Monikka Nur Winnarto  
NIM : 14002374  
Program Studi : Ilmu Komputer  
Fakultas : Teknologi Informasi  
Jenjang : Strata Dua (S2)  
Konsentrasi : Image Processing  
Judul : “Penerapan Arsitektur *MobilenetV2* Pada Klasifikasi Penyakit Daun Teh”

Perkembangan produksi teh di Indonesia semakin menurun setiap tahunnya. Penurunan produksi teh salah satunya disebabkan adanya serangan hama dan penyakit pada daun teh yang mempengaruhi hasil panen dan kualitas teh. Klasifikasi penyakit daun teh sangat penting untuk mengetahui perawatan yang diperlukan dalam keberlanjutan sistem budidaya. Klasifikasi penyakit daun teh masih mengandalkan tenaga profesional dan pengalaman kerja petani, hal ini akan memakan waktu, melelahkan dan tidak efisien. Beberapa tahun terakhir kemunculan algoritma pembelajaran mesin menyediakan sistem dukungan untuk klasifikasi penyakit daun teh. Penelitian ini bertujuan untuk melakukan klasifikasi jenis penyakit daun teh menggunakan arsitektur CNN *MobileNetV2*. *MobilenetV2* memiliki score akurasi cukup tinggi, jumlah *training parameters* dan *model size* yang kecil namun memiliki performa yang baik. Tahapan penelitian berupa pengumpulan data, augmentasi, data *preprocessing*, klasifikasi, evaluasi dan implementasi. Arsitektur *MobileNetV2* menghasilkan akurasi sebesar 98,33% dan nilai loss 0,0307% yang menunjukkan bahwa arsitektur ini mampu melakukan klasifikasi penyakit daun teh dengan baik.

Kata Kunci : Klasifikasi, Penyakit Daun Teh, CNN, *MobileNetV2*

## **ABSTRACT**

Name : Monikka Nur Winnarto  
NIM : 14002374  
Study of Program : Ilmu Komputer  
Faculty : Information Technology  
Levels : Strata Dua (S2)  
Concentration : Image Processing  
Title : “Application of MobilenetV2 Architecture on Tea Leaf Disease Classification”

*The development of tea production in Indonesia is decreasing every year. One of the reasons for the decline in tea production is the presence of pests and diseases on tea leaves that affect the yield and quality of tea. Classification of tea leaf disease is very important to know the care needed in the sustainability of the cultivation system. Classification of tea leaf diseases still relies on professionals and work experience of farmers, this will be time consuming, tiring and inefficient. In recent years the emergence of machine learning algorithms provides a support system for tea leaf disease classification. This study aims to classify the types of tea leaf disease using the CNN MobileNetV2 architecture. MobilenetV2 has a fairly high accuracy score, a number of training parameters and a small model size but has good performance. The stages of the research are data collection, augmentation, data preprocessing, classification, evaluation and implementation. The MobileNetV2 architecture produces an accuracy of 98.33% and a loss value of 0.0307% which shows that this architecture is able to classify the disease of tea leaf well.*

*Keywords: Classification, Tea Leaf Disease, CNN, MobileNetV2*

## DAFTAR ISI

|                                                                                                |             |
|------------------------------------------------------------------------------------------------|-------------|
| <b>HALAMAN SAMPUL .....</b>                                                                    | <b>i</b>    |
| <b>HALAMAN JUDUL .....</b>                                                                     | <b>ii</b>   |
| <b>SURAT PERNYATAAN ORISINALITAS DAN BEBAS PLAGIARISME..</b>                                   | <b>iii</b>  |
| <b>HALAMAN PERSETUJUAN DAN PENGESAHAN TESIS.....</b>                                           | <b>iv</b>   |
| <b>LEMBAR BIMBINGAN TESIS .....</b>                                                            | <b>v</b>    |
| <b>KATA PENGANTAR.....</b>                                                                     | <b>vi</b>   |
| <b>SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH<br/>UNTUK KEPENTINGAN AKADEMIS.....</b> | <b>viii</b> |
| <b>ABSTRAK .....</b>                                                                           | <b>ix</b>   |
| <b>ABSTRACT .....</b>                                                                          | <b>x</b>    |
| <b>DAFTAR ISI.....</b>                                                                         | <b>xi</b>   |
| <b>DAFTAR TABEL.....</b>                                                                       | <b>xiii</b> |
| <b>DAFTAR GAMBAR.....</b>                                                                      | <b>xiv</b>  |
| <b>DAFTAR LAMPIRAN .....</b>                                                                   | <b>xv</b>   |
| <b>BAB I PENDAHULUAN .....</b>                                                                 | <b>1</b>    |
| 1.1 Latar Belakang Penulisan .....                                                             | 1           |
| 1.2 Identifikasi Masalah .....                                                                 | 4           |
| 1.3 Tujuan Penelitian.....                                                                     | 4           |
| 1.4 Ruang Lingkup Penelitian .....                                                             | 4           |
| 1.5 Hipotesis .....                                                                            | 5           |
| 1.6 Sistematika Penulisan .....                                                                | 5           |
| <b>BAB II LANDASAN/KERANGKA PEMIKIRAN.....</b>                                                 | <b>7</b>    |
| 2.1 Tinjauan Pustaka.....                                                                      | 7           |
| 2.1.1 Teh .....                                                                                | 7           |
| 2.1.2 Klasifikasi .....                                                                        | 9           |
| 2.1.3 Citra Digital .....                                                                      | 9           |
| 2.1.4 Pemrosesan Citra Digital .....                                                           | 10          |
| 2.1.5 Data Augmentasi.....                                                                     | 10          |
| 2.1.6 <i>Deep Learning</i> .....                                                               | 11          |
| 2.1.7 Convolutional Neural Network (CNN).....                                                  | 12          |
| 2.1.8 <i>Hyperparameter</i> .....                                                              | 15          |
| 2.1.9 <i>MobileNetV2</i> .....                                                                 | 17          |
| 2.1.10 <i>ImageNet Large Scale Visual Recognition Challenges<br/>(ILSVRC)</i> .....            | 20          |
| 2.1.11 <i>Transfer Learning</i> .....                                                          | 20          |
| 2.1.12 <i>Google Colaboratory</i> .....                                                        | 21          |
| 2.1.13 <i>Global Average Polling (GAP)</i> .....                                               | 21          |
| 2.1.14 <i>Python</i> .....                                                                     | 21          |
| 2.1.15 <i>Flask</i> .....                                                                      | 22          |
| 2.1.16 Keras dan Tensorflow.....                                                               | 22          |
| 2.2 Tinjauan Studi.....                                                                        | 23          |
| 2.3 Tinjauan Organisasi/Object Penelitian .....                                                | 26          |
| <b>BAB III METODOLOGI PENELITIAN .....</b>                                                     | <b>27</b>   |
| 3.1 Metode Penelitian .....                                                                    | 27          |

|                                                    |           |
|----------------------------------------------------|-----------|
| 3.2 Pengumpulan Data.....                          | 28        |
| 3.3 Augmentasi Data .....                          | 29        |
| 3.4 Data Preprocessing.....                        | 30        |
| 3.5 Klasifikasi.....                               | 30        |
| 3.5.1 Pelatihan.....                               | 32        |
| 3.5.2 Pengujian .....                              | 32        |
| 3.6 Evaluasi .....                                 | 32        |
| 3.7 Implementasi .....                             | 33        |
| 3.8 Tools Pendukung .....                          | 33        |
| <b>BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....</b> | <b>35</b> |
| 4.1 Pengumpulan Data.....                          | 35        |
| 4.2 Augmentasi Data .....                          | 37        |
| 4.3 Data <i>Preprocessing</i> .....                | 41        |
| 4.4 Klasifikasi.....                               | 42        |
| 4.5 Evaluasi .....                                 | 46        |
| 4.6 Implementasi .....                             | 48        |
| <b>BAB V PENUTUP .....</b>                         | <b>56</b> |
| 5.1 Kesimpulan.....                                | 56        |
| 5.2 Saran .....                                    | 56        |
| <b>DAFTAR REFERENSI .....</b>                      | <b>57</b> |
| <b>DAFTAR RIWAYAT HIDUP .....</b>                  | <b>60</b> |
| <b>SURAT KETERANGAN RISET .....</b>                | <b>61</b> |
| <b>LAMPIRAN.....</b>                               | <b>62</b> |

## DAFTAR TABEL

|                                                                                           | Halaman |
|-------------------------------------------------------------------------------------------|---------|
| Tabel 2.1. Tinjauan Studi .....                                                           | 25      |
| Tabel 3.1. Teknik Augmentasi .....                                                        | 29      |
| Tabel 3.2. Tabel Model <i>Summary MobilenetV2</i> .....                                   | 31      |
| Tabel 3.4. Spesifikasi <i>Hardware</i> .....                                              | 33      |
| Tabel 3.5. Spesifikasi <i>Software</i> .....                                              | 34      |
| Tabel 4.1. Jumlah Cita Pada Setiap Kelas .....                                            | 35      |
| Tabel 4.2. Sample Citra Daun Teh .....                                                    | 36      |
| Tabel 4.3. Hasil Augmentasi Citra .....                                                   | 37      |
| Tabel 4.4. Pembagian Data .....                                                           | 41      |
| Tabel 4.5. Penelitian Penentuan Dengan Augmentasi Data dan Tanpa<br>Augmentasi Data ..... | 42      |
| Tabel 4.6. Penelitian Penentuan Nilai <i>Batch Size</i> .....                             | 43      |
| Tabel 4.7. Penelitian Penentuan Jumlah <i>Epoch</i> .....                                 | 43      |
| Tabel 4.8. Perbandingan Nilai Akurasi Dan <i>Loss</i> .....                               | 44      |
| Tabel 4.9. <i>Confusion Matrix</i> .....                                                  | 46      |
| Tabel 4.10. <i>Classification Report</i> .....                                            | 47      |
| Tabel 4.11. Kesimpulan Hasil Klasifikasi .....                                            | 53      |

## DAFTAR GAMBAR

|                                                                                       | Halaman |
|---------------------------------------------------------------------------------------|---------|
| Gambar 2.1. Contoh Augmentasi Data .....                                              | 11      |
| Gambar 2.2. Jaringan Syaraf <i>Deep Learning</i> .....                                | 12      |
| Gambar 2.3. Arsitekure CNN .....                                                      | 13      |
| Gambar 2.4. Proses <i>Convolutional Layer</i> .....                                   | 13      |
| Gambar 2.5. Proses <i>Polling</i> untuk mereduksi dimensi data .....                  | 14      |
| Gambar 2.6. Normal <i>Convolution</i> vs <i>Depthwise Separable Convolution</i> ..... | 17      |
| Gambar 2.7. <i>Bootleneck Residual Block</i> .....                                    | 18      |
| Gambar 2.8. Arsitektur <i>MobileNetV2</i> .....                                       | 20      |
| Gambar 2.9. <i>MobileNetV2</i> .....                                                  | 20      |
| Gambar 3.1. Tahapan Penelitian .....                                                  | 27      |
| Gambar 3.2. Perangkat yang digunakan untuk pengambilan data citra .....               | 28      |
| Gambar 3.3. Model <i>MobileNetV2</i> yang diusulkan .....                             | 31      |
| Gambar 4.1. Grafik akurasi <i>training</i> dan validasi .....                         | 44      |
| Gambar 4.2. Grafik akurasi loss <i>training</i> dan validasi .....                    | 44      |
| Gambar 4.3. Hasil Klasifikasi Penyakit Daun Teh .....                                 | 45      |
| Gambar 4.4. <i>Flowchart Website</i> Klasifikasi Jenis Penyakit Daun Teh .....        | 48      |
| Gambar 4.5. Halaman <i>Index</i> .....                                                | 49      |
| Gambar 4.6. Hasil Klasifikasi Citra Daun Teh Sehat .....                              | 49      |
| Gambar 4.7. Hasil Klasifikasi Citra Daun Teh <i>Blister Blight</i> .....              | 50      |
| Gambar 4.8. Hasil Klasifikasi Citra Daun Teh <i>Mite</i> .....                        | 50      |
| Gambar 4.9. Hasil Klasifikasi Citra Daun Teh <i>Helopeltis</i> .....                  | 51      |
| Gambar 4.10. Hasil Klasifikasi Citra Daun Teh Ulat Jengkal .....                      | 51      |
| Gambar 4.11. Hasil Klasifikasi Citra Daun Teh <i>Empoasca</i> .....                   | 52      |
| Gambar 4.12. Hasil Klasifikasi Citra Daun Teh Keseluruhan .....                       | 52      |

## DAFTAR LAMPIRAN

|                                                                | Halaman |
|----------------------------------------------------------------|---------|
| Lampiran 1. Dataset Daun Teh Kelas Daun Sehat .....            | 62      |
| Lampiran 2. Dataset Daun Teh Kelas <i>Blister Blight</i> ..... | 64      |
| Lampiran 3. Dataset Daun The Kelas <i>Mite</i> .....           | 66      |
| Lampiran 4. Dataset Daun Teh Kelas <i>Helopeltis</i> .....     | 68      |
| Lampiran 5. Dataset Daun Teh Kelas Ulat Jengkal .....          | 70      |
| Lampiran 6. Dataset Daun Teh Kelas <i>Empoasca</i> .....       | 72      |

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang Penulisan

Teh merupakan salah satu hasil perkebunan yang memiliki peran penting untuk kegiatan perekonomian di Indonesia. Teh merupakan penghasil devisa negara selain minyak dan gas. Sebagai bahan minuman, teh memiliki nilai lebih dibandingkan dengan minuman lainnya, teh kaya akan mineral dan vitamin yang diperlukan oleh tubuh. Berbagai manfaat teh untuk kesehatan juga telah diakui oleh para pakar gizi. Produksi teh Indonesia sebagian besar dipasarkan ke mancanegara. Perkembangan produksi daun teh tahun 2017 hingga 2019 cenderung mengalami penurunan. Pada tahun 2017 produksi daun teh kering sebesar 97.590 ton mengalami penurunan menjadi 90.016 ton di tahun 2018 dengan persentase penurunan sebesar 7,76%. Tahun 2019 produksi daun teh kering menurun sebesar 79.449 ton dengan persentase penurunan 11,7%. Penurunan produksi teh mengakibatkan penurunan nilai ekspor teh ke berbagai benua seperti Asia, Afrika, Australia, Amerika, dan Eropa. Pada Tahun 2017 volume ekspor teh sebesar 54.195 ton dengan nilai US\$ 114,2 juta mengalami penurunan di tahun 2018 sebesar 9,51% menjadi 49.038 ton dengan nilai ekspor US\$ 108,5 Juta. Tahun 2019 volume ekspor kembali mengalami penurunan menjadi 42.811 ton atau sebesar 12,70% dengan nilai ekspor US\$ 92,3 Juta [1] . Penurunan produksi teh di Indonesia salah satunya disebabkan adanya serangan hama dan penyakit pada daun teh.

Hama dan penyakit pada daun teh memiliki karakteristik yang berbeda yang dapat dibedakan dari kerusakan pada daunnya [2]. Di Indonesia hama dan penyakit yang tergolong kategori penting yang menyerang tanaman teh adalah penyakit daun teh Blister Blight, Mite, Helopeltis, *Ulat Jengkal* dan Empoasca. Penyakit *Blister Blight* yang disebabkan oleh jamur *E.Vexans* dapat menurunkan produksi pucuk basah sampai 50% karena menyerang daun dengan gejala awal terlihat bintik-bintik kecil tembus cahaya dan pasca panen teh bercak menjadi coklat tua akhirnya daun teh mati dan membentuk lubang. *Mite* penyakit pada daun teh yang disebabkan oleh polusi udara sehingga daun menjadi kusam. Kepik pengisap daun atau *Helopeltis*

menyerang pucuk daun muda, kepik ini menusuk dan menghisap daun teh sehingga menjadi bercak-bercak hitam. Ulat Jengkal menyerang daun, pupus daun dan pentil teh, serangan berat menyebabkan daun berlubang dan pucuk tanaman gundul, sehingga tinggal tulang daunnya. *Empoasca* menyerang pucuk dan daun muda dengan cara mengisap cairan daun. Munculnya penyakit pada daun teh sangat mempengaruhi hasil dan kualitas teh [3]. Klasifikasi penyakit pada daun teh sangat penting untuk mengetahui perawatan yang diperlukan dalam keberlanjutan sistem budidaya sehingga dapat mencegah hilangnya hasil panen dan kualitas tanaman [4]. Klasifikasi penyakit pada daun teh mengandalkan tenaga profesional dan pengalaman kerja para petani teh namun akan memakan waktu, melelahkan dan tidak efisien [5]. Untuk membantu para petani dalam tugas penting terkait penyakit daun teh dalam masa pertumbuhan, praktis untuk memiliki sistem klasifikasi sebagai tindakan pencegahan. Kemunculan banyak algoritma pembelajaran mesin menyediakan sistem dukungan untuk klasifikasi penyakit daun teh [5].

Dalam beberapa tahun terakhir, klasifikasi menggunakan pemrosesan gambar tradisional telah secara bertahap digantikan oleh *Deep Learning* salah satunya adalah *Convolutional Neural Network* [6]. Model *Convolutional Neural Network* (CNN) telah banyak digunakan dalam masalah klasifikasi citra [7]. *Convolutional Neural Network* (CNN) merupakan jaringan saraf *feed-forward multi-layer* yang memiliki kemampuannya untuk mengidentifikasi target dengan kompleksitas tinggi dengan kecepatan perhitungan yang lebih cepat dan merupakan model pembelajaran mesin yang populer [6]. *Convolutional Neural Network* (CNN) telah banyak digunakan dalam masalah klasifikasi citra [7]. *Convolutional Neural Network* (CNN) merupakan metode yang paling optimal dalam kasus klasifikasi citra, dimana salah satu kelebihanannya adalah ekstraksi fitur citra yang dilakukan secara otomatis sehingga dapat menghemat waktu dan tenaga.

Penelitian [3] melakukan klasifikasi penyakit daun teh menggunakan dataset 3810 citra yang berisi 7 jenis penyakit daun teh. Metode klasifikasi menggunakan *Machine learning* dan *Convolutional Neural Network* (CNN) dengan arsitektur *LeafNet*. Arsitektur *LeafNet* menghasilkan akurasi sebesar 90,16% dan mampu mengalahkan algoritma klasifikasi SVM dan MLP dengan akurasi sebesar 60,62% dan 70,77%. Model Tradisional yang diusulkan belum menggunakan proses

ekstraksi fitur secara otomatis dan menghasilkan akurasi yang lebih rendah dari CNN. CNN yang disebut LeafNet diusulkan dalam penelitian ini adalah model sekuensial di mana *Convolutional Neural Network* ditumpuk di atas lapisan sebelumnya dan aliran informasi hanya berjalan satu arah. Menurut [2] model sekuensial memiliki keterbatasan, terutama ketika jaringan menggunakan banyak lapisan. Pertama, jaringan mungkin kehilangan beberapa informasi karena implementasi dari lapisan penyatuan. Kedua, jaringan mungkin membutuhkan lebih banyak data untuk pelatihan karena tingginya jumlah parameter yang harus dioptimalkan.

Penelitian [2] melakukan klasifikasi penyakit daun teh dengan menggabungkan tiga Arsitektur CNN yaitu *GoogleNet*, *Xception* dan *Inception-Resnet-V2* dimana semua arsitektur menggunakan data test dan data validasi yang sama dengan fungsi aktivasi ReLu dan *Batch Size* sejumlah 10. 4272 citra digunakan dengan perbandingan data *training*, *testing* dan *validation* sebesar 80% 10% dan 10% dan menghasilkan akurasi sebesar 89,64%.

Penelitian [8] melakukan klasifikasi penyakit daun tomat menggunakan *Convolutional Neural Network* dengan arsitektur *MobileNetV2*. *MobileNetV2* disesuaikan untuk mendeteksi tiga jenis penyakit daun tomat. Algoritma *MobileNetV2* diuji pada 4.671 citra yang diperoleh dari kumpulan data *PlantVillage*. Performa klasifikasi terbaik diperoleh saat *MobileNetV2* dilatih menggunakan Adagrad dengan ukuran *batch* 16 dengan *learning rate* 0,001 dan pembagian data rasio 4:1 antara pelatihan dan pengujian memberikan kinerja klasifikasi yang paling akurat. Hasil menunjukkan bahwa *MobileNetV2* mampu mendeteksi penyakit dengan hasil akurasi 95,94%.

Penelitian [9] melakukan klasifikasi genus tanaman anggrek menggunakan metode *Convolutional Neural Network* (CNN) berbasis *Website*. Pada penelitian ini akan digunakan 4 tipe arsitektur CNN yaitu arsitektur kustom, arsitektur kustom dengan *dropout*, arsitektur *MobileNetV2* orisinal, dan arsitektur *MobileNetV2* dengan *dropout*. Arsitektur *MobileNetV2* teruji menghasilkan *score* akurasi *testing* yang tinggi yaitu 90.44%. *MobileNetV2* memiliki *score* akurasi cukup tinggi, jumlah *training parameters* yang kecil dibandingkan dengan arsitektur CNN lainnya, sehingga kebutuhan akan komputasinya lebih ringan. Selain itu *model size*

*MobileNetV2* berukuran kecil yaitu 14MB namun memiliki performa yang baik sehingga saat model di *deploy* kedalam sebuah *real app* seperti aplikasi *android* atau aplikasi berbasis *website* akan lebih ringan.

Berdasarkan permasalahan yang terjadi, Pada penelitian ini *Covolutional Neural Network* menggunakan arsitektur *MobileNetV2* diusulkan untuk melakukan klasifikasi penyakit daun teh ke dalam 6 kelas yaitu daun sehat, *Blister Blight*, *Mite*, *Helopeltis*, Ulat Jengkal dan *Empoasca*.

## 1.2 Identifikasi Masalah

Identifikasi masalah dalam penelitian ini adalah :

1. Bagaimana penerapan arsitektur *MobileNetV2* untuk klasifikasi jenis penyakit daun teh?
2. Bagaimana kinerja arsitektur *MobileNetV2* dalam melakukan klasifikasi jenis penyakit daun teh?
3. Bagaimana hasil klasifikasi penyakit daun teh menggunakan arsitektur *MobileNetV2*?

## 1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah :

1. Mengetahui penerapan arsitektur *MobileNetV2* untuk klasifikasi penyakit daun teh.
2. Mengetahui kinerja arsitektur *MobileNetV2* dalam melakukan klasifikasi jenis penyakit daun teh.
3. Mengetahui hasil klasifikasi penyakit daun teh menggunakan arsitektur *MobileNetV2*.

## 1.4 Ruang Lingkup Penelitian

Ruang lingkup dalam penelitian ini adalah :

1. Klasifikasi citra mencakup 6 kelas yang terdiri dari 5 kelas penyakit daun teh yaitu *Blister Blight*, *Mite*, *Helopeltis*, Ulat Jengkal, *Empoasca* dan 1 kelas daun teh sehat.
2. Klasifikasi menggunakan Arsitektur *MobileNetV2*.

### 1.5 Hipotesis

Hipotesis yang diajukan dalam penelitian ini adalah :

1. Diduga arsitektur *MobileNetV2* dapat diterapkan untuk klasifikasi jenis penyakit daun teh.
2. Diduga arsitektur *MobileNetV2* menghasilkan akurasi yang tinggi dalam melakukan klasifikasi jenis penyakit daun teh.
3. Diduga arsitektur *MobileNetV2* mampu melakukan klasifikasi jenis penyakit pada daun teh dengan baik.

### 1.6 Sistematika Penulisan

Sistematika penulisan yang diuraikan dalam penulisan tesis ini dapat diuraikan sebagai berikut:

#### **BAB I PENDAHULUAN**

Pada bab ini akan dibahas tentang latar belakang penulisan, identifikasi masalah, tujuan penelitian, ruang lingkup penelitian, hipotesis dan sistematika penulisan.

#### **BAB II LANDASAN/KERANGKA PEMIKIRAN**

Pada bab ini akan dibahas mengenai teori-teori dan konsep yang berhubungan dengan penelitian yang dilakukan. Selain itu pada bab ini memaparkan penelitian-penelitian terdahulu yang berhubungan dengan permasalahan yang diteliti. Memaparkan objek penelitian dan Memuat teori-teori dalam pelaksanaan pengumpulan dan pengolahan data serta metode yang digunakan.

#### **BAB III METODE PENELITIAN**

Bab ini menjelaskan tentang metode penelitian yang merupakan tahapan-tahapan penelitian yang dilakukan. Tahapan penelitian terdiri dari tahap pengumpulan data, augmentasi data, data *preprocessing*, klasifikasi menggunakan arsitektur *MobileNetV2*, evaluasi dan implementasi dalam bentuk *website*.

#### **BAB IV HASIL PENELITIAN DAN PEMBAHASAN**

Bab ini menjelaskan mengenai hasil dan pembahasan yang dilakukan yaitu mengenai tahap pengumpulan data, augmentasi data, data *preprocessing*, klasifikasi menggunakan arsitektur *MobileNetV2*, evaluasi dan implementasi.

#### **BAB V PENUTUP**

Bab ini membahas mengenai kesimpulan yang diperoleh dari hasil penelitian yang telah dilakukan serta saran-saran yang dapat diterapkan dari hasil penelitian yang dapat menjadi masukan yang berguna kedepannya.

## **BAB II**

### **LANDASAN/KERANGKA PEMIKIRAN**

#### **2.1 Tinjauan Pustaka**

Dalam penulisan tesis ini dilakukan tinjauan studi dengan menggunakan jurnal dan buku yang berhubungan dengan tema yang dipilih. Secara lebih detail tinjauan studi dalam penulisan tesis ini dijelaskan sebagai berikut :

##### **2.1.1 Teh**

Teh termasuk kedalam keluarga jenis *Camellia* yaitu *Camellia assamica* dan *camellia sinensis* yang berasal dari China, Tibet dan India bagian utara. Tanaman teh memiliki 2 (dua) jenis varietas utama. *Camellia sinensis* yang hidup di pegunungan tinggi seperti China dan Jepang merupakan varietas berdaun kecil. *Camellia assamica* yang tumbuh di daerah beriklim tropis dan lembab merupakan varietas berdaun lebar. Di Indonesia tanaman teh yang paling umum ditanam adalah jenis varietas *camellia assamica*, namun beberapa perkebunan juga memiliki varietas *camellia sinensis* yang telah disilang agar dapat tumbuh di daerah dengan iklim tropis. Varietas *camellia assamica* yang banyak ditanam di Indonesai seperti Gabung dan TRI [10].

Teh memiliki enam jenis yang berasal dari tanaman yang sama dan yang membedakan adalah pada pemrosesannya. Jenis-jenis diantaranya ada teh hijau, teh putih, teh hitam, teh oolong dan teh kuning [10]. Walaupun teh mempunyai berbagai varietas, pada dasarnya jenis teh terbanyak di Indonesia hanya terdiri dari 2 kelompok utama yaitu [1]:

##### **a. Teh Hijau**

Pada pengolahannya teh hijau tidak melalui proses oksidasi enzimatis (proses pengeringan). Karena teh hijau tanpa melalui proses oksidasi enzimatis, untuk memperkuat aroma perlu ditambahkan wangian dari pohon selain teh seperti bunga melati.

b. Teh Hitam

Teh hitam berbeda dengan teh hijau, teh hitam melalui proses oksidasi enzimatis (proses pengeringan) dalam pengolahannya. Tidak adanya unsur-unsur lain pada teh hitam diluar pucuk teh.

Perkembangan produksi daun teh kering tahun 2017 sampai dengan 2019 cenderung menurun. Penurunan ini disebabkan oleh adanya hama dan tanaman pada daun teh. Menurut [11] Berbagai jenis penyakit dan hama yang tergolong penting pada daun teh adalah sebagai berikut:

a. Kepik Penghisap Daun Teh (*Helopeltis sp.*)

Kepik penghisap daun atau yang biasa disebut *Helopeltis*, hama ini menyerang bagian pucuk daun teh yang masih muda. *Helopeltis* akan menusuk lalu mengisap, serangan *helopeltis* menyebabkan bercak-bercak hitam pada daun teh.

b. Ulat Jengkal

Ulat jengkal menyerang bagian daun, pucuk dan pupus daun. Daun teh yang mengalami serangan dari ulat jengkal akan berlubang dan pucuk daun teh akan menjadi gundul dan yang tersisa hanya tulang daunnya.

c. *Empoasca sp*

*Empoasca* menyerang daun muda dan pucuk daun muda dengan cara mengisap cairan pada daun. Daun yang terkena hama *Empoasca* akan mengerut dan mengalami perubahan warna menjadi sedikit kuning.

d. Cacar daun atau *Blister Blight (Exobasidium vexans* Massee)

Penyakit cacar daun teh dapat menurunkan produksi pucuk daun yang masih basah hingga 50% karena penyakit ini menyerang daun dan ranting yang muda. Cacar daun teh terjadi pada daun pertama, daun kedua, daun ketiga dan pucuk piko. Awalnya daun terlihat bintik-bintik kecil tembus cahaya, kemudian bercak melebar dan pusat bercak menjadi coklat tua akhirnya daun mati dan berlubang. Penyakit ini terus mengalami perkembangan yang disebabkan karena udara yang lembab, ketinggian lokasi perkebunan, angin dan sifat pada tanaman.

e. *Mite*

*Mite* penyakit pada daun teh yang disebabkan oleh polusi udara sehingga daun menjadi kusam.

### 2.1.2 Klasifikasi

Klasifikasi dapat diartikan sebagai usaha untuk membuat suatu fungsi atau model yang dapat digunakan dalam menjelaskan dan membedakan konsep dari kelas data. Klasifikasi dapat mengenali suatu objek dan mengelompokkan sesuai dengan kelasnya yaitu memprediksi hasil terhadap model yang dibentuk. Proses dalam klasifikasi dibagi menjadi 2 tahap yaitu tahap pelatihan yang merupakan tahap pembelajaran suatu data yang kelasnya sudah diketahui dan pengujian untuk melakukan evaluasi kinerja model yang dihasilkan pada tahap pelatihan menggunakan data baru atau data uji. Pada tahap ini menghasilkan tingkat keakuratan dari model dalam memprediksi data yang kelasnya belum diketahui yaitu menggunakan data test atau data uji. Proses klasifikasi berhubungan dengan algoritma atau teknik yang mampu belajar dan melakukan pengelompokan data ke dalam kelas-kelasnya [12].

Sebuah algoritma yang menerapkan klasifikasi dikenal sebagai “*Classifier*”. *Classifier* juga mengarah pada fungsi matematika dan diimplementasikan oleh algoritma klasifikasi yang memetakan data masukan ke suatu kategori. Dalam Pembelajaran mesin, klasifikasi sering dikenal sebagai *instance*, variabel penjelas disebut fitur (dikelompokkan ke dalam vektor fitur) dan kategori yang mungkin diprediksi adalah kelas [13].

### 2.1.3 Citra Digital

Citra merupakan kemiripan atau gambaran suatu objek [14]. Citra digital sebagai representasi atas citra dua dimensi (2D) menggunakan sejumlah berhingga dari titik-titik yang biasanya dikenal dengan elemen citra, piksel atau pel. Piksel direpresentasikan oleh nilai numeris. Citra Abu atau monokrom sebuah nilai merepresentasikan intensitas dari piksel dalam rentang [0, 255], untuk citra warna tiga nilai (merepresentasikan intensitas komponen RGB (*Red*, *Green*, *Blue*)) diperlukan [15].

Ada beberapa tipe citra yang digunakan diantaranya [14] :

1. Citra Biner

Pada citra biner 1 bit memori yang dibutuhkan setiap pixel. Setiap pixel memiliki kemungkinan 2 buah nilai intensitas yaitu 0 atau 1.

2. Citra Abu (*Grayscale*)

Citra *grayscale* merupakan matriks yang nilainya mewakili intensitas setiap piksel antara 0 - 255. Setiap piksel membutuhkan delapan bit memory.

3. Citra RGB (Warna)

Citra warna adalah citra yang setiap pixelnya mempunyai 3 (tiga) komponen warna yang spesifik yaitu komponen *red*, *green* dan *blue*. Warna setiap piksel ditentukan berdasarkan kombinasi dari intensitas warna *red*, *green* dan *blue* yang tersimpan pada bidang warna di lokasi pixel sebagai citra 24 bit, citra ini berasal dari komponen *red*, *green* dan *blue* dengan masing-masing delapan bit.

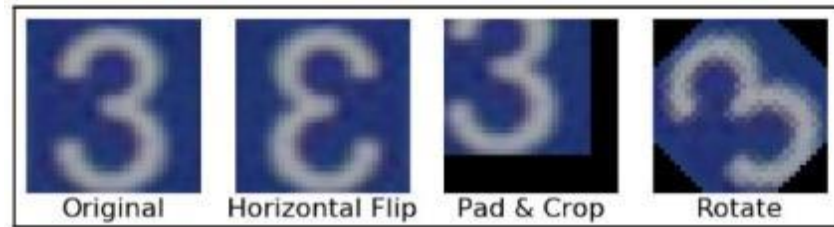
#### 2.1.4 Pemrosesan Citra Digital

Pemrosesan citra atau *image preprocessing* adalah bentuk pengolahan sinyal berupa citra dan keluarannya berupa citra lain dengan menggunakan teknik tertentu. Tujuan dari *Image preprocessing* adalah memperbaiki data sinyal gambar yang disebabkan akibat transmisi dan selama akuisi sinyal juga untuk meningkatkan kualitas gambar yang disajikan agar lebih mudah diinterpretasi oleh sistem penglihatan baik melakukan manipulasi atau analisa terhadap citra. Dalam ilmu komputer pemrosesan citra digital adalah penggunaan komputer digital untuk memproses gambar digital melalui suatu algoritma [13].

#### 2.1.5 Data Augmentasi

Augmentasi merupakan suatu teknik yang berfungsi memperbesar gambar, memutar, memberikan cahaya dan teknik augmentasi dapat menaikkan nilai akurasi model [16]. Augmentasi dapat digunakan sebagai strategi dalam meningkatkan keragaman data sebagai model pelatihan, tanpa benar-benar mengumpulkan data baru. Teknik augmentasi seperti *padding*, pemotongan citra (*cropping*), dan *flip* horizontal adalah Teknik yang umum digunakan untuk melatih jaringan neural

besar. Pendekatan yang digunakan dalam pelatihan jaringan neural Sebagian besar hanya menggunakan tipe augmentasi dasar. Sementara arsitektur jaringan neural telah diselidiki secara mendalam [17]. Gambar 2.4 menampilkan contoh data yang diaugmentasi.



Gambar 2.1. Contoh Data Augmentasi

Sumber : [17]

a. *Flip*

Pembalikan Citra (*Flip*) bisa dilakukan secara horizontal atau vertikal. Misalnya citra telapak tangan yang mengarah ke depan dapat dibalik menjadi mengarah ke belakang.

b. *Random Cropping*

*Random cropping* digunakan untuk membuat berbagai macam data dari berbagai macam sudut pandang. *Random cropping* menguci sebuah data masukan dengan banyak model klasifikasi.

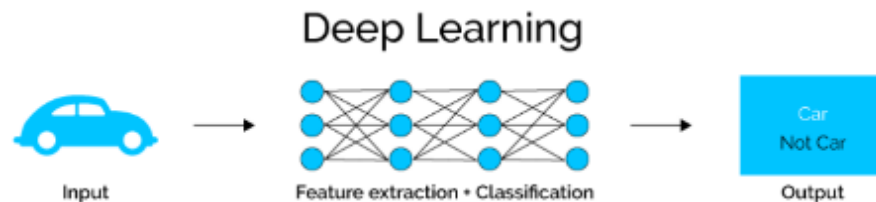
c. *Random Combination*

Digunakan untuk teknik pemrosesan citra seperti pembesaran/pengecilan, *rotation* atau memutar citra dengan derajat tertentu, peregangan citra atau pengerutan citra, *shearing* dan *lens distortions*.

### 2.1.6 Deep Learning

*Deep learning* adalah teknik pembelajaran yang memiliki kemampuan feature engineering yang dapat merekayasa fitur secara otomatis sehingga tidak perlu berusaha dengan susah payah membangun model ekstraksi fitur yang seringkasli sangat rumit. *Deep learning* adalah algoritma pembelajaran mesin yang terinspirasi oleh struktur otak manusia [18]. Algoritma *Deep Larning* dilatih untuk mengidentifikasi pola dan mengklasifikasikan berbagai jenis informasi untuk memberikan *output* yang diinginkan ketika menerima *input* baru. Perbedaan

*machine learning* dan *deep learning* adalah pada *machine learning* *feature extractor* perlu dirancang secara manual dimana ini akan memakan banyak waktu dan tenaga. Sedangkan *deep learning* secara otomatis mengekstrak *features* untuk klasifikasi. *Deep learning* membutuhkan sejumlah data besar untuk melatih algoritmanya. Aplikasi *Deep learning* salah satunya adalah klasifikasi citra [9]. Gambar 2.2 menampilkan jaringan syaraf *deep learning*.



Gambar 2.2. Jaringan Syaraf *Deep Learning*

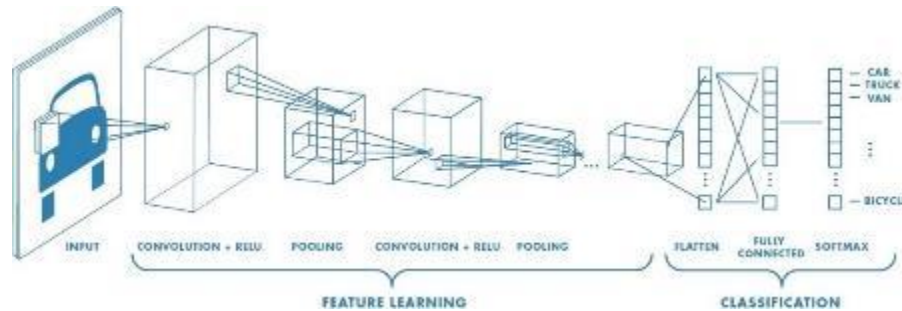
Sumber : <https://talks.navixy.com/>

Informasi akan di *transfer* dari satu lapisan ke lapisan lainnya melalui saluran penghubung. Semua *neuron* memiliki angka unik yang disebut bias. Bias ditambahkan ke jumlah tertimbang input yang mencapai neuron kemudian diterapkan pada fungsi aktivasi. Setiap *neuron* diaktifkan kemudian meneruskan informasi ke lapisan berikutnya dan berlanjut hingga lapisan terakhir kedua. Lapisan keluaran adalah lapisan terakhir yang menghasilkan keluaran untuk program. Melatih jaringan seperti ini, membutuhkan data pelatihan yang besar dikarenakan parameter yang besar harus dipertimbangkan agar solusi menjadi akurat [19].

### 2.1.7 Convolutional Neural Network (CNN)

*Convolutional Neural Network* merupakan MLP (*Multi Layer Perceptron*) yang didesain untuk melakukan identifikasi *image/citra* 2 dimensi dimana CNN menirukan dari cara kerja otak manusia dalam melakukan pengenalan objek yang dilihatnya [20]. CNN adalah metode *Machine Learning* (ML) yang dikembangkan dari *Multi-Layer Perceptron* (MLP) yang diciptakan untuk mengerjakan data 2D. *Convolutional Neural Network* (CNN) termasuk kedalam golongan *Deep Neural Network* (NN).

Arsitektur CNN terdiri dari satu lapisan masukan, suatu lapisan keluaran dan sejumlah lapisan tersembunyi. *Hidden Layer* umumnya berisi *convolutional layers*, *polling layers*, *normalization layers*, *ReLU layer*, *fully connected layers* dan *loss layer* (Atom et al dalam [18]). Gambar 2.3 menunjukkan arsitektur dari CNN.

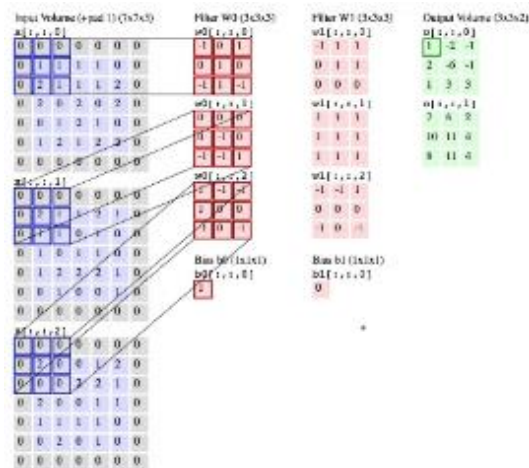


Gambar 2.3. Arsitektur CNN

Sumber : <https://towardsdatascience.com/>

### 1. Convolutional Layer

*Convolutional layer* merupakan block bangunan inti CNN, dimana komputasi akan dilakukan pada lapisan ini. *Concolutional layer* menggunakan lebih dari satu *filter*. Sebagai contoh jika empat *filter* yang digunakan maka *convolutional layer* akan berisi empat *neuron* yang tersusun dalam kisi berukuran  $28 \times 28 \times 4$  dengan demikian ada empat *neuron* yang melihat area yang sama pada citra masukan tersebut sehingga *overviting* dapat dikontrol secara mudah. Tujuan dari *convolutional layer* ialah untuk mengekstraksi fitur dari citra *input* (misalnya tepi, sudut, tekstur, dll). Keluaran dari *convolutional layer* biasa disebut *feature map*. Gambar 2.4 menunjukkan proses dari *convolutional layer*.

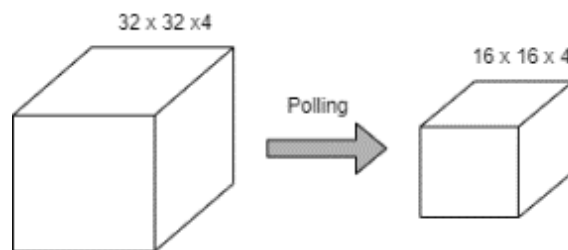


Gambar 2.4. Proses Convolutional Layer

Sumber : [18]

## 2. *Pooling Layer*

*Pooling layer* menjadikan ukuran data kecil pada saat *convolutional* dilakukan yaitu dengan cara reduksi sample (*downsampling*). Dengan pooling data dapat direpresentasikan menjadi lebih kecil, mudah menangani *overfitting* dan mudah mengelolanya. Tujuan *pooling layer* ialah mengurangi dimensi/mereduksi (*downsampling*) *feature map*, sehingga dapat mempercepat komputasi karena akan semakin sedikit bobot yang harus diupdate. Gambar 2.5 menunjukkan proses *pooling* untuk mereduksi dimensi data.



Gambar 2.5. Proses *Pooling* untuk mereduksi dimensi data

Sumber : [18]

Proses *pooling* biasanya dilakukan dengan menggunakan *max pooling* atau *average pooling*. Yaitu dengan memilih nilai maksimum dan atau nilai rata-rata dalam suatu area tertentu.

## 3. *Normalization Layer*

*Normalization layer* berguna untuk mengatasi perbedaan rentang nilai yang signifikan pada citra masukan. Para ahli telah mengusulkan banyak jenis lapisan normalisasi, namun saat ini lapisan normalisasi tidak banyak digunakan secara praktis karena dampaknya yang relatif kecil atau tidak ada dampaknya sama sekali.

## 4. *ReLU Layer*

Fungsi Aktivasi ReLU (*Rectified Linear Unit*) digunakan setelah melakukan proses konvolusi/sebelum melakukan *pooling* (di tahap *feature extractor*) dan dipakai pula di setiap *node* di *hidden layer* (di tahap *fully connected*). ReLU merupakan lapisan aktivasi dengan fungsi  $f(x) = \max(0, x)$  yang membuat seluruh nilai pixel yang bernilai kurang dari nol pada suatu citra akan dijadikan 0. Tujuan ReLU ialah mengurangi linearitas yang terjadi dari proses konvolusi sehingga CNN lebih mudah mencapai nilai optimum.

### 5. *Fully Connected Layer*

*Neuron* yang ada pada *convolutional layer* harus ditransformasi menjadi data 2D sebelum masuk ke *fully connected layer*. *Feature map* dari proses *feature extraction* masih berbentuk multidimensional *array*, sehingga harus dilakukan *flatten* atau *reshape feature map* menjadi sebuah *vector* agar dapat digunakan sebagai inputan dari *fully connected layer*. *Vector* hasil proses *flatten* diumpankan dan akan diproses dengan *feed forward neural network* dan *backpropagation* dimana setiap proses pelatihannya menggunakan jumlah *epoch* [12] .

*Fully Connected Layer* terdiri dari *input*, *hidden* dan *output layer*, jika diperhatikan *layer-layer* ini sama seperti di ANN. Oleh karena itu beberapa literatur menyebut *fully connected layer* ini sebagai ANN. Disetiap *hidden layer* terdapat fungsi aktivasi, yang umum digunakan yaitu ReLU, dan begitu pula di *output layer*, juga terdapat fungsi aktivasi, dalam kasus klasifikasi (yang lebih dari 2 label/kelas/kategori) yang umum digunakan yaitu *Softmax*. Tujuan utama dari *fully connected layer* ialah mengolah data sehingga bisa dilakukan klasifikasi. *Output* dari *fully connected layer* yaitu probabilitas terhadap kategori (jika menggunakan *Softmax*).

### 6. *Loss Layer*

*Loss layer* merupakan lapisan terakhir dalam CNN yang menentukan bagaimana pelatihan memberikan penalti atas penyimpanan antara hasil prediksi dan label. Terdapat sejumlah variasi *loss function* diantaranya *softmax* digunakan untuk memprediksi satu dari kelas-kelas yang saling eksklusif, *sigmoid cross-entropy loss* untuk memprediksi sejumlah nilai probabilitas dengan interval [0,1] dan *Eulidean loss* yang digunakan untuk regresi nilai kontinu. Fungsi Aktivasi *Softmax* dipakai dalam kasus klasifikasi lebih dari 2 kelas (*multi-class*).

#### 2.1.8 *Hyperparameter*

Berikut *hyperparameter* yang bisa disesuaikan oleh perancang model CNN:

##### a. *Input Shape*

Bisa saja model CNN tidak mempelajari *features* di citra dengan baik akibat ukuran citra yang terlalu kecil, sehingga ukuran citra sebagai *input* perlu

diperhatikan. Beberapa arsitektur CNN yang terkenal menyarankan ukuran citra (*shape*) yaitu 224x224x3 piksel (3 adalah *channel* RGB).

b. *Batch Size*

Pelatihan data (*training*) secara keseluruhan sekaligus itu berat secara komputasi, terlebih jika *dataset training* yang digunakan banyak misalnya hingga ribuan dan akan semakin berat lagi jika data berupa data citra. Terdapat beberapa nilai yang umum digunakan sebagai *batch size*, yaitu 16/32/64/128/256.

c. *Epoch*

Satu *epoch* berarti satu putaran penuh pelatihan (*training*) terhadap seluruh *dataset*. Sebagai acuan, hentikan *epoch* saat nilai *loss* sudah *converge* (tidak turun lagi) atau nilai akurasi sudah tidak mengalami kenaikan yang signifikan.

d. *Jumlah Filter*

Jumlah filter di *Convolutional Layer* bisa disesuaikan dan tidak ada keterangan yang pasti terkait berapa jumlah *filter* yang optimal, namun sebagai acuan secara umum banyak para peneliti yang menggunakan nilai 2n (misalnya: 8/16/32/64/128/256).

e. *Filter Size*

Ukuran *filter* di *Convolutional Layer* dan *Pooling Layer* bisa disesuaikan, namun secara umum ukuran *filter Convolutional* yaitu 3x3 atau 5x5 dengan *stride*=1 atau *stride*=2, sedangkan untuk *Pooling* yaitu 2x2 dengan *stride*=2 (ini akan mereduksi/mengecilkan citra sebesar 2x dari ukuran citra aslinya).

f. *Stride*

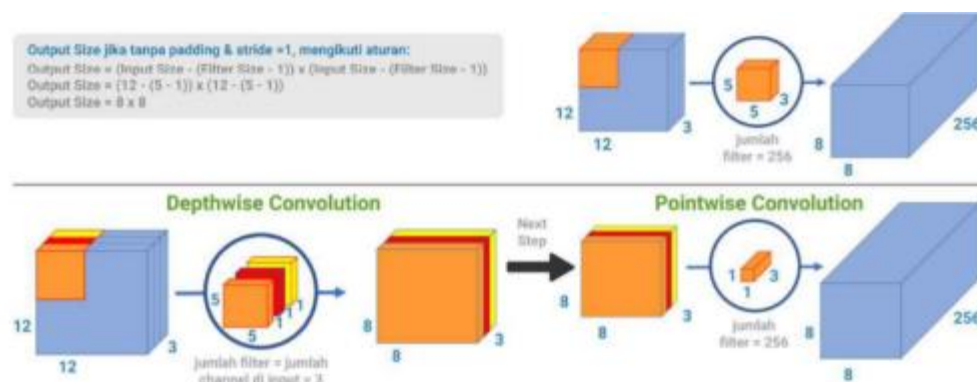
Digunakan untuk menentukan berapa jumlah pergeseran *filter*. Apabila diberikan nilai 1, maka *filter* akan bergeser 1 piksel secara horizontal kemudian vertikal hingga seluruh area terkena efek *filter*.

g. *Optimizer*

Sebuah algoritma atau metode yang digunakan untuk meng-*update* bobot (*weight*) dengan tujuan untuk menurunkan nilai *loss* sehingga menghasilkan *score* akurasi yang baik. Salah satu *optimizer* terbaik dan paling sering digunakan kalangan para peneliti yaitu *Adam Optimizer*, dianggap sebagai *optimizer* yang cepat dalam mencapai *loss* minimum (*converge*).

### 2.1.9 MobileNetV2

*MobileNet* dibuat oleh Google, mempunyai *layer* khusus yang disebut dengan *depthwise separable convolution* terdiri dari dua blok yaitu *depthwise convolution* dan *pointwise convolution*. Tujuan adalah untuk mereduksi komputasi dalam artian agar parameternya lebih sedikit sehingga dapat menghasilkan ukuran model yang lebih kecil. Untuk memahami *depthwise separable convolution*, berikut ilustrasi perbandingannya dengan normal *convolution*. Terdapat input *image* berukuran 12x12x3 dan 256 kernel berukuran 5x5x3. Gambar 2.6 menunjukkan perbandingan dari normal *convolution* dan *depthwise separable convolution*.



Gambar 2.6. Normal Convolution vs Depthwise Separable Convolution

Sumber : <https://hackmd.io/>

Dalam normal *convolution*, *filter* melakukan konvolusi terhadap input *image* dan menghasilkan *feature map* (*output image*). Sedangkan dalam *depthwise convolution*, jumlah filter akan sama dengan jumlah channel di *input image*, setiap filter akan melakukan konvolusi terhadap masing-masing *channel* di *input image* (atau dengan kata lain, tidak langsung melakukan konvolusi terhadap seluruh *channel* di *input image* seperti yang dilakukan pada normal *convolution*), lalu menghasilkan *feature map* yang memiliki jumlah channel yang sama dengan jumlah *filter*. Tahap selanjutnya, agar output *image* dengan *depthwise convolution* sama dengan *output image* yang dihasilkan dengan normal *convolution*, maka akan dilakukan *pointwise convolution*, dimana hasil dari tahap *depthwise convolution* akan dikonvolusi lagi dengan *filter* 1x1, *filter* ini memiliki kedalaman sama dengan jumlah *channel* di *output image* sebelumnya. Berikut perbandingan jumlah komputasi antara normal *convolution* dengan *depthwise separable convolution*:

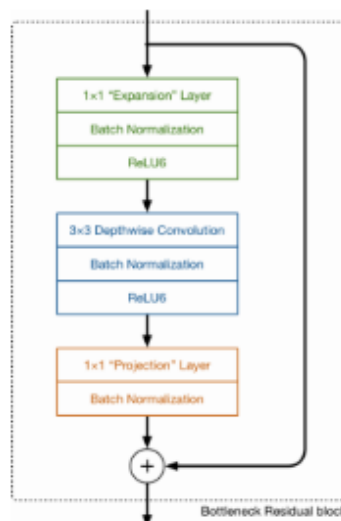
a. *Normal Convolution*

Terdapat 256 filter  $5 \times 5 \times 3$  yang bergeser sebanyak  $8 \times 8$  kali ( $8 \times 8$  yaitu jumlah pergeseran *filter*  $5 \times 5$  dari kiri atas ke kanan bawah terhadap input *image*  $12 \times 12$ ). Ini berarti  $256 \times 5 \times 5 \times 3 \times 8 \times 8 = 1.228.800$  total perkalian yang dilakukan saat proses konvolusi.

b. *Depthwise Separable Convolution*

Pertama, pada tahap *depthwise convolution* terdapat 3 filter  $5 \times 5 \times 1$  yang bergeser sebanyak  $8 \times 8$  kali. Ini berarti  $3 \times 5 \times 5 \times 1 \times 8 \times 8 = 4.800$ . Selanjutnya, pada tahap *pointwise convolution* terdapat 256 filter  $1 \times 1 \times 3$  yang bergeser sebanyak  $8 \times 8$  kali. Ini berarti  $256 \times 1 \times 1 \times 3 \times 8 \times 8 = 49.152$ . Maka total perkalian yang dilakukan saat proses konvolusi di *depthwise separable convolution* ini yaitu  $4.800 + 49.152 = 52.952$ .

Perkembangan lebih lanjut dari *MobileNet* yaitu *MobileNetV2*, dimana ditambahkan fitur baru yaitu *bottleneck residual block* yang merupakan sebuah blok dalam arsitektur yang didalamnya menggunakan *depthwise separable convolution*. Blok bangunan utamanya sekarang ditunjukkan oleh gambar 2.7.



Gambar 2.7. *Bottleneck Residual Block*

Sumber : [21]

*MobileNetV1* konvolusi pointwise menjaga jumlah saluran tetap sama atau menggandakannya. Dalam *MobileNetV2* melakukan yang sebaliknya yaitu membuat jumlah saluran lebih kecil. Inilah sebabnya mengapa lapisan ini sekarang dikenal sebagai *projection layer*, yang memproyeksikan data dengan jumlah

dimensi (saluran) yang tinggi ke dalam tensor dengan jumlah dimensi yang jauh lebih rendah.

Lapisan pertama adalah anak baru di blok tersebut yang merupakan konvolusi  $1 \times 1$ . Tujuannya adalah untuk memperluas jumlah saluran dalam data sebelum masuk ke *depthwise convolution*, oleh karena itu, di *expansion layer* selalu memiliki lebih banyak saluran keluaran daripada saluran masukan dan melakukan kebalikan dari lapisan proyeksi. Persisnya seberapa banyak data yang diperluas diberikan oleh *expansion factor*. Ini adalah salah satu *hyperparameter* untuk bereksperimen dengan pengorbanan arsitektur yang berbeda. Faktor ekspansi defaultnya adalah 6. Masing-masing lapisan memiliki *batch normalization* dan fungsi aktivasi adalah ReLU6. Namun, output dari lapisan proyeksi tidak memiliki fungsi aktivasi karena lapisan ini menghasilkan data berdimensi rendah, menggunakan nonlinearitas setelah lapisan ini benar-benar menghancurkan informasi yang berguna.

Hal baru kedua di blok bangunan *MobileNetV2* adalah *residual connection*. Blok ini berfungsi seperti di *ResNet* yang membantu aliran gradien melalui jaringan. (Sambungan residual hanya digunakan ketika jumlah saluran yang masuk ke blok sama dengan jumlah saluran yang keluar darinya, yang tidak selalu terjadi karena setiap beberapa blok saluran keluaran bertambah.) Blok residual asli berisi input diikuti oleh beberapa bottlenecks kemudian diikuti oleh ekspansi dan pintasan ada di antara lapisan tebal (lapisan dengan banyak saluran). Namun, terinspirasi oleh intuisi bahwa *bottlenecks* sebenarnya berisi semua informasi yang diperlukan dan lapisan ekspansi bertindak hanya sebagai transformasi non-linear, *MobileNetV2* menggunakan pintasan langsung di antara *bottlenecks* (lapisan tipis). Lapisan menetas menggunakan aktivasi linier. ReLU6 digunakan sebagai aktivasi non-liner karena kekokohnya saat digunakan dengan komputasi presisi rendah. Arsitektur umum *MobileNet V2* ditunjukkan oleh gambar 2.8.

| Input                    | Operator    | $t$ | $c$  | $n$ | $s$ |
|--------------------------|-------------|-----|------|-----|-----|
| $224^2 \times 3$         | conv2d      | -   | 32   | 1   | 2   |
| $112^2 \times 32$        | bottleneck  | 1   | 16   | 1   | 1   |
| $112^2 \times 16$        | bottleneck  | 6   | 24   | 2   | 2   |
| $56^2 \times 24$         | bottleneck  | 6   | 32   | 3   | 2   |
| $28^2 \times 32$         | bottleneck  | 6   | 64   | 4   | 2   |
| $14^2 \times 64$         | bottleneck  | 6   | 96   | 3   | 1   |
| $14^2 \times 96$         | bottleneck  | 6   | 160  | 3   | 2   |
| $7^2 \times 160$         | bottleneck  | 6   | 320  | 1   | 1   |
| $7^2 \times 320$         | conv2d 1x1  | -   | 1280 | 1   | 1   |
| $7^2 \times 1280$        | avgpool 7x7 | -   | -    | 1   | -   |
| $1 \times 1 \times 1280$ | conv2d 1x1  | -   | k    | -   | -   |

Gambar 2.8. Arsitektur *MobileNetV2*

Sumber : [21]

Berikut spesifikasi *MobileNetV2* yang ditunjukkan oleh gambar 2.9.

| Model       | Size  | Top-1 Accuracy | Top-5 Accuracy | Parameters | Depth |
|-------------|-------|----------------|----------------|------------|-------|
| MobileNetV2 | 14 MB | 0.713          | 0.901          | 3,538,984  | 88    |

Gambar 2.9. *MobileNetV2*

Sumber : <https://keras.io/api/applications/>

#### 2.1.10 *ImageNet Large Scale Visual Recognition Challenges (ILSVRC)*

ILSVRS adalah kompetisi global yang mengevaluasi algoritma untuk mendeteksi obyek dan melakukan klasifikasi gambar pada skala yang menjadi standar acuan dalam pengenalan obyek skala besar [22]. *ImageNet* merupakan dataset yang memiliki 1 juta gambar dengan 1000 kelas, masing-masing memiliki 50 ribu dataset validasi dan 150 ribu dataset uji [23].

#### 2.1.11 *Transfer Learning*

*Transfer learning* merupakan model arsitektur CNN yang sudah dilatih oleh suatu dataset sebelumnya yang bisa digunakan untuk klasifikasi pada dataset lain. *Transfer learning* memiliki arsitektur lapisan *convolution* dan *pooling* yang lebih dalam dibandingkan arsitektur CNN sederhana, sehingga dapat melakukan ekstraksi tekstur citra lebih banyak dan menghasilkan informasi dari citra yang lebih baik. Kedalaman lapisan arsitektur pada *Transfer Learning* membuat suatu metode bisa mengatasi *overfit* pada data yang sedikit [24].

### 2.1.12 *Google Colaboratory*

*Google Colaboratory* adalah proyek yang memiliki tujuan menyebarluaskan pendidikan pembelajaran mesin. *Notebook* kolaboratif didasarkan pada *Jupyter* dan berfungsi sebagai objek *Google Documents* yang dapat dibagikan dan antar pengguna dapat berkolaborasi pada *notebook* yang sama. Kolaborasi telah menyediakan runtime Python 2 dan 3 dan telah dikonfigurasi sebelumnya dengan essential pembelajaran mesin dan kecerdasan buatan, seperti TensorFlow, Matplotlib, dan Keras. Mesin virtual di bawah runtime (VM) dinonaktifkan setelah jangka waktu tertentu dan semua data dan konfigurasi pengguna hilang. Namun, *notebook* diawetkan, dan juga memungkinkan untuk mentransfer file dari *hard disk* VM ke akun *Google Drive* pengguna [25].

### 2.1.13 *Global Average Polling (GAP)*

*Fully Connected layer* selalu menjadi konfigurasi standar jaringan CNN. Namun terlalu banyak parameter di *fully connecter layer* memperlambat kecepatan pelatihan jaringan untuk training dan membuat mudah *overviting*. *Global Average Polling (GAP)* adalah rata-rata seluruh piksel dari setiap peta fitur dan mendapatkan output untuk setiap peta fitur. Vektor yang terdiri dari fitur *output* akan dikirim ke *softmax* untuk melakukan klasifikasi [26]. Gagasan *Global Average Polling* adalah untuk menghasilkan satu peta fitur untuk setiap kategori klasifikasi yang sesuai di lapisan konvolusi 1x1x1 terakhir. Alih-alih menambahkan lapisan *fully connected layer* di atas peta fitur, dibutuhkan rata-rata setiap peta fitur dan memasukkan vektor yang dihasilkan langsung ke lapisan *softmax* [27].

### 2.1.14 *Python*

Python adalah salah satu bahasa pemrograman tingkat tinggi yang telah menjadi standar dunia komputasi ilmiah. *Python* adalah bahasa pemrograman *open source multi-platform* yang bisa digunakan pada banyak sistem informasi seperti Windows, Linux, dan MacOS. *Python* juga termasuk bahasa pemrograman yang mudah dipelajari. Bahasa pemograman python lebih mudah dibaca dan lebih ringkas bahasa pemograman lain seperti C dan Fortran. Terdapat modul standar dalam *Python* yang menyediakan fungsi dan algoritma, untuk menyelesaikan

pekerjaan seperti mengurai data teks, memanipulasi dan mencari file di dalam disk, menuliskan dan membaca file terkompresi dan mengunduh data dari server web. Keunggulan utama dari suatu bahasa pemrograman terinterpretasi (*Python*) adalah tidak membutuhkan deklarasi variabel.[28].

#### 2.1.15 *Flask*

*Flask* adalah *Framework Aplikasi Web* yang ditulis dalam bahasa pemrograman *Python*. *Flask* dapat mempermudah pengembangan *Aplikasi Web*. *Flask* menggunakan dua *libraries eksternal* yaitu *Jinja2 Template Engine* dan *WSGI Toolkit. Extensions* digunakan untuk menambahkan berbagai fitur yang tidak disediakan oleh *Flask*, dapat ditambahkan seperti *Upload Handling* atau *Form Validation*. *Flask* tersedia dalam bentuk *library* dan dapat di-install melalui *Package Manager Python* [29].

#### 2.1.16 *Keras dan Tensorflow*

Keras menyediakan API tingkat tinggi yang sederhana, bersih, dan mudah digunakan untuk membangun model pembelajaran mendalam yang efisien dengan kode minimal. Kelebihannya adalah dapat dikonfigurasi untuk bekerja di atas beberapa kerangka kerja *deep learning* tingkat rendah (disebut *backend*), termasuk *theano* dan *TensorFlow*. Dokumentasi Keras dapat diakses di <https://keras.io/>

*Tensorflow* merupakan kerangka kerja pembelajaran mendalam yang paling populer. Dibuat oleh *Google Brain* dan *open source* pada tahun 2015, ia dengan cepat menarik perhatian ML, peneliti pembelajaran mendalam, insinyur, dan ilmuwan data. *TensorFlow* memiliki dukungan untuk eksekusi berbasis multi-CPU dan GPU, serta mendukung beberapa bahasa, termasuk C++, Java, R, dan *Python*. *TensorFlow* secara umum adalah *library* tingkat rendah, mirip dengan *Theano*, tetapi juga memiliki kemampuan untuk memanfaatkan API tingkat tinggi untuk pembuatan prototipe dan pengembangan yang cepat. Bagian penting dari *TensorFlow* juga mencakup modul *tf.contrib*, yang berisi berbagai fitur eksperimental, termasuk Keras API itu sendiri [30].

## 2.2 Tinjauan Studi

Berikut adalah beberapa penelitian sebelumnya yang pernah dilakukan dan menjadi rujukan untuk menyusun penelitian ini. Secara garis besar tinjauan pustaka dalam penelitian ini meliputi :

- a. Penelitian [2] melakukan klasifikasi penyakit daun teh menggunakan gabungan tiga Arsitektur CNN yaitu *GoogleNet*, *Xception* dan *Inception-Resnet-V2* dimana semua arsitektur menggunakan data *test* dan data validasi yang sama dengan fungsi aktivasi *ReLU* dan *Batch Size* sejumlah 10. 4272 citra digunakan dengan perbandingan data *training*, *testing* dan *validation* sebesar 80% 10% dan 10% dan menghasilkan akurasi sebesar 89,64%.
- b. Penelitian [3] melakukan klasifikasi penyakit daun teh dengan jumlah digunakan sebanyak 3810 citra yang berisi 7 jenis penyakit. Klasifikasi arsitektur *LeafNet*. Arsitektur ini mampu mengenali tujuh jenis penyakit daun teh yang disajikan dengan hasil akurasi sebesar 90,16% dan mampu mengalahkan algoritma klasifikasi SVM dan MLP dengan akurasi sebesar 60,62 dan 70,77%.
- c. Penelitian [8] melakukan klasifikasi penyakit daun tomat menggunakan arsitektur *MobileNetV2*. Algoritma *MobileNet V2* diuji pada 4.671 citra yang diperoleh dari kumpulan data *PlantVillage*. Performa klasifikasi terbaik diperoleh saat *MobileNetV2* yang dilatih menggunakan *Adagrad* dengan ukuran batch 16 dengan learning rate 0,001 dan pembagian data rasio 4:1 antara pelatihan dan pengujian memberikan kinerja klasifikasi yang paling akurat. Hasil menunjukkan bahwa *MobileNet V2* mampu mendeteksi penyakit dengan hasil akurasi 95,94%. *MobileNet* memiliki jaringan yang efisien dan merupakan kumpulan dari jaringan yang terdiri dari dua *hyperparameter* untuk membangun model dengan latensi minimal dan rendah. *MobileNet* dibuat dengan metode semi-konvolusi mendalam untuk mengurangi perhitungan beberapa lapisan pertama [31].
- d. Penelitian [32] melakukan deteksi dan klasifikasi penyakit pada daun teh menggunakan *deep learning*. Algoritma *Deep Learning* yang digunakan adalah arsitektur *LeNet*. Penelitian ini melakukan klasifikasi empat jenis penyakit daun

- teh. *Tea Red Scab* adalah penyakit yang paling akurat diklasifikasikan dengan hasil *sensitivity* 94%. Arsitektur *LeNet* menghasilkan akurasi sebesar 90,23%.
- e. Penelitian [33] melakukan klasifikasi penyakit daun pada tanaman menggunakan *machine learning* dan *image processing*. Penelitian ini menggunakan 20.000 citra dengan 19 kelas. Algoritma yang digunakan adalah *Logistic Regression*, KNN, SVM dan CNN dengan alur metode penelitian berupa Pengumpulan data, Pemrosesan gambar, segmentasi dan pemilihan klasifikasi. *Logistic Regression* menghasilkan akurasi sebesar 66,4 %, KNN sebesar 54,4 %, SVM sebesar 53,4 % dan CNN sebesar 98%.
  - f. Penelitian [34] melakukan klasifikasi level *non-proliferatif* Retinopati diabetik menggunakan tiga arsitektur salah satunya adalah *MobileNetV2*, model akan diuji menggunakan optimasi Adam dan SGD. Dataset dari penelitian ini sejumlah 1460 citra fundus, dataset dipilih secara acak dari data APTOS competition 209. Setiap kelas memiliki 365 data. Model *MobileNetV2* dapat mengklasifikasi level retinopati diabetik dengan akurasi pada data uji sebesar 89%.
  - g. Penelitian [35] melakukan klasifikasi daun menggunakan *MobileNet*. *MobileNet* dilatih menggunakan Dataset *ImageNet* dan menerima gambar RGB dengan ukuran 224 x 224 piksel dan diklasifikasikan ke dalam kelas yang sesuai. Arsitektur ini terdiri dari empat belas lapisan konvolusi dan tiga belas lapisan *convolutional Depthwise* yang merupakan salah satu di antara satu sama lain, lapisan yang terhubung penuh dan pengklasifikasi *softmax*. Arsitektur ini mencakup 4,2 juta parameter. Hasil menunjukkan akurasi sebesar 99,6%.
  - h. Penelitian [9] melakukan klasifikasi genus tanaman anggrek dengan metode *Convolutional Neural Network* berbasis Website. Arsitektur *MobileNetV2* teruji menghasilkan *score* akurasi *testing* yang tinggi yaitu 90.44%.
  - i. Penelitian [36] melakukan klasifikasi kanker jinak dan ganas berbasis *Mobile*. Dataset yang digunakan merupakan citra lesi kulit dengan jumlah 48.373 citra. Klasifikasi dilakukan menggunakan arsitektur *MobileNetV2* menggunakan transfer learning dengan *hyperparameter* berupa *optimizer* adam dan beberapa

percobaan *batch size* yaitu 64, 128 dan 32. Konfigurasi dengan ukuran *batch* 32 bekerja paling baik menghasilkan akurasi 91,33%.

- j. Penelitian [37] melakukan klasifikasi menggunakan arsitektur *Deep Learning* untuk mendeteksi dan mengklasifikasikan gejala penyakit tanaman. Berbagai arsitektur CNN digunakan pada penelitian ini salah satunya adalah *MobileNet* menggunakan dataset *PlantVillage* dengan 24 Jenis tanaman. Arsitektur *MobileNet* menghasilkan akurasi sebesar 98,34% dan memiliki jumlah parameter yang lebih sedikit daripada VGG sehingga dapat menghemat waktu dalam melatih model. Tabel 2.1 menjelaskan secara ringkas beberapa tinjauan studi.

Tabel 2.1. Tinjauan Studi

| Penulis | Judul                                                                                                  | Metode                                                              | Persamaan                                         | Perbedaan                                                    |
|---------|--------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|---------------------------------------------------|--------------------------------------------------------------|
| [2]     | <i>Diseases Classification For Tea Plant Using Concatenated Convolution Neural Network</i>             | <i>GoogLeNet</i> , <i>Xception</i> , dan <i>Inception-ResNet-v2</i> | Penelitian mengenai klasifikasi penyakit daun teh | Metode yang digunakan dan Jumlah kelas yang diklasifikasikan |
| [3]     | <i>Visual Tea Leaf Disease Recognition Using A Convolutional Neural Network Model</i>                  | MLP, SVM dan arsitektur <i>LeafNet</i>                              | Penelitian mengenai klasifikasi penyakit daun teh | Metode yang digunakan dan jumlah kelas yang diklasifikasikan |
| [8]     | <i>Classification Of Tomato Leaf Diseases Using Mobilenet V2</i>                                       | <i>MobileNet V2</i>                                                 | Arsitektur CNN yang digunakan                     | Objek yang diklasifikasikan                                  |
| [32]    | <i>Image Analysis And Detection Of Tea Leaf Disease Using Deep Learning</i>                            | CNN arsitektur <i>LeNet</i>                                         | Penelitian mengenai klasifikasi penyakit daun teh | Metode yang digunakan dan jumlah kelas yang diklasifikasikan |
| [33]    | <i>Classification Of Plant Leaf Diseases Using Machine Learning And Image Preprocessing Techniques</i> | <i>Regresi Logistik</i> , KNN, SVM dan CNN                          | Penelitian mengenai klasifikasi penyakit daun     | Metode yang digunakan                                        |

| Penulis | Judul                                                                                                      | Metode             | Persamaan                     | Perbedaan                   |
|---------|------------------------------------------------------------------------------------------------------------|--------------------|-------------------------------|-----------------------------|
| [34]    | Klasifikasi Level Non-Proliferatif Retinopati Diabetik Dengan <i>Ensemble Convolutional Neural Network</i> | <i>MobileNetV2</i> | Arsitektur CNN yang digunakan | Objek yang diklasifikasikan |
| [35]    | <i>Leaf Classification For Plant Recognition With Deep Transfer Learning</i>                               | <i>MobileNet</i>   | Arsitektur CNN yang digunakan | Objek yang diklasifikasikan |
| [9]     | Klasifikasi Genus Tanaman Anggrek Menggunakan Metode <i>Convolutional Neural Network (Cnn)</i>             | <i>MobileNetV2</i> | Arsitektur CNN yang digunakan | Objek yang diklasifikasikan |
| [36]    | <i>Deep Neural Network Based Mobile Dermoscopy Application For Triaging Skin Cancer Detection</i>          | <i>MobileNetV2</i> | Arsitektur CNN yang digunakan | Objek yang diklasifikasikan |
| [37]    | <i>Plant Disease Detection And Classification By Deep Learning</i>                                         | <i>MobileNet</i>   | Arsitektur CNN yang digunakan | Objek yang diklasifikasikan |

Sumber : Hasil Penelitian

### 2.3 Tinjauan Organisasi/Object Penelitian

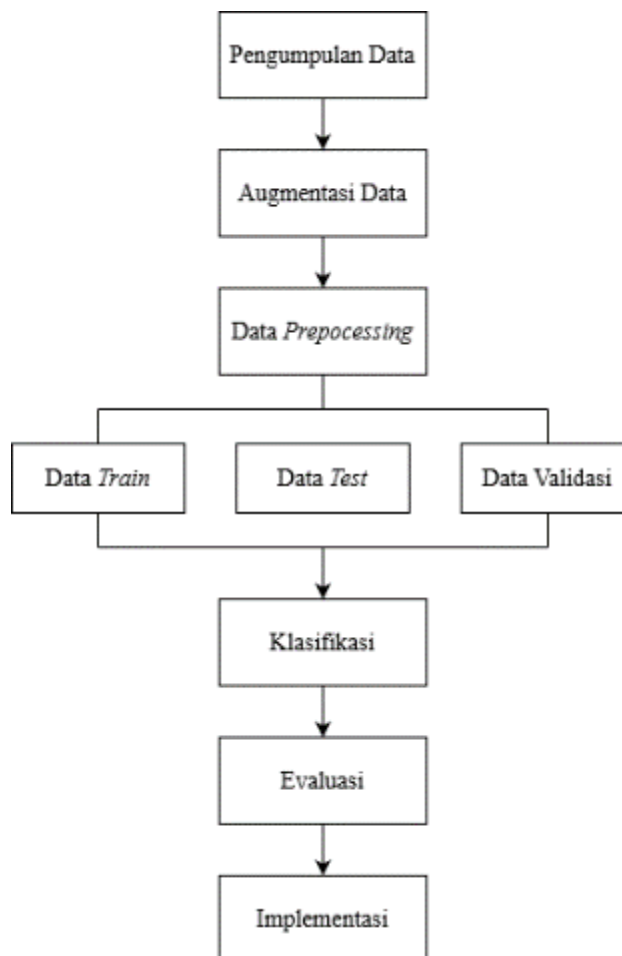
Objek penelitian pada tesis ini adalah mengenai penyakit daun teh. Dataset penyakit daun teh merupakan dataset privat yang diambil dari Perkebunan Nusantara VIII Kebun Gedeh Mas yang terletak di Cisarua, Bogor, Jawa Barat. Dataset penyakit daun berjumlah 600 citra yang terdiri dari 6 kelas yaitu Daun Sehat, *Blister Blight*, *Mite*, *Helopeltis*, Ulat Jengkal dan *Empoasca*.

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Metode Penelitian

Metode penelitian yang dilakukan mengacu pada penelitian [2] yang terdiri dari pengumpulan dataset, data *preprocessing*, pembagian dataset, perancangan model klasifikasi dan hasil klasifikasi. Penelitian ini menggunakan penelitian eksperimen. Penelitian eksperimen melibatkan penyelidikan perlakuan pada parameter atau variabel dan menggunakan tes yang dikendalikan oleh peneliti itu sendiri. Gambar 3.1 merupakan tahapan penelitian yang dilakukan dalam penelitian klasifikasi penyakit daun teh.



Gambar 3.1. Tahapan Penelitian

Sumber : Hasil Penelitian

### 3.2 Pengumpulan Data

Tahap pertama yaitu pengumpulan dataset berupa citra daun teh yang akan digunakan. Data Citra Daun teh diambil dari Perkebunan Nusantara VIII Kebun Gedeh Mas menggunakan kamera utama (kamera belakang) Smartphone Xiaomi Redmi Note 4 dengan resolusi 13 Megapixel. Gambar 3.2 menampilkan perangkat yang digunakan untuk pengambilan data citra daun teh.



Gambar 3.2. Perangkat yang digunakan untuk pengambilan data citra

Sumber : Hasil Penelitian

Sebelum melakukan pengambilan citra, daun teh diletakkan di atas kertas putih untuk mendapatkan citra dengan *background* putih. Hal ini dilakukan untuk menyamakan keadaan pada setiap daun teh dan fokus mendapatkan objek daun teh. Citra daun teh terdiri dari 6 kelas yaitu daun sehat, daun karena penyakit *Blister*, *Blight*, *Mite*, *Helopeltis*, Ulat Jengkal dan *Empoasca*. Citra daun teh yang diambil rata-rata memiliki ukuran dimensi 4160 x 3120 dengan karakteristik warna RGB. Setiap kelas memiliki 10 sample daun teh yang diambil dari beberapa sisi untuk mendapatkan berbagai citra dengan bentuk yang berbeda sehingga dapat digunakan untuk pengujian suatu model. Setiap daun diambil dari 10 sisi yang berbeda sehingga setiap kelas memiliki total 100 citra daun teh dan total daun teh yang didapatkan dari semua kelas adalah 600 citra.

### 3.3 Augmentasi Data

*Deep learning* membutuhkan sejumlah data besar untuk melatih algoritmanya, sehingga diperlukan metode untuk memperbanyak citra yang dapat menghasilkan data baru tanpa harus melakukan sample tambahan yaitu dengan augmentasi data. Augmentasi merupakan suatu teknik yang berfungsi memperbesar gambar, memutar, memberikan cahaya dan teknik augmentasi dapat menaikkan nilai akurasi model [16]. Augmentasi data pada penelitian ini bertujuan untuk memperbanyak citra daun teh guna menaikkan akurasi model dan mencegah *overfitting*.

Tabel 3.1 menunjukkan teknik augmentasi yang dilakukan pada penelitian ini.

Tabel. 3.1. Teknik Augmentasi

| Variabel               | Teknik Augmentasi |
|------------------------|-------------------|
| <i>Rotation range</i>  | 30                |
| <i>Shear range</i>     | 0.2               |
| <i>Zoom range</i>      | 0.2               |
| <i>Horizontal flip</i> | True              |

Sumber : Hasil Penelitian

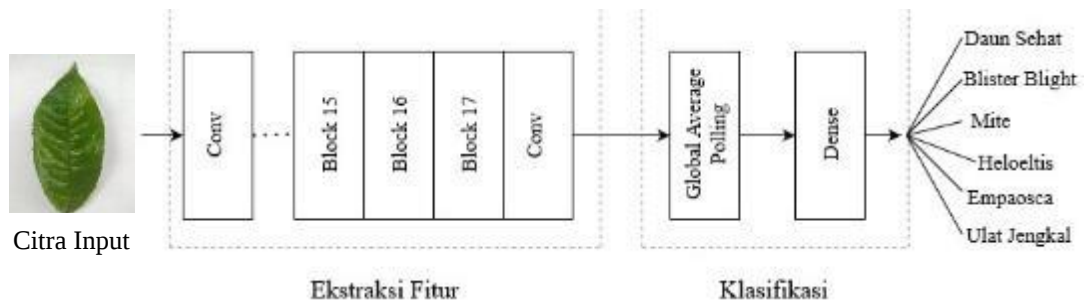
Augmentasi citra menggunakan teknik *Rotation* yaitu memutar citra secara acak dengan sudut 30 derajat, *Shear range* yaitu menentukan sudut kemiringan derajat, *Zoom range* yang digunakan untuk memperbesar citra dan *Horizontal Flip* yang digunakan untuk membalik citra secara horizontal. Setiap kelas dilakukan augmentasi sebanyak 100 citra dengan ukuran citra yang tersimpan adalah 224x224, format “jpg” dan warna RGB. Jumlah keseluruhan citra augmentasi adalah 600 citra. Hasil citra asli dan citra augmentasi menghasilkan jumlah citra sebesar 1200 citra dan merupakan keseluruhan jumlah citra yang akan diproses dalam penelitian ini.

### 3.4 Data Preprocessing

Tahap persiapan atau data *preprocessing* adalah mengolah data asli agar data siap digunakan pada model klasifikasi. Data preprocessing pada penelitian ini adalah merubah ukuran citra menjadi ukuran 224x224 piksel sehingga semua citra memiliki ukuran yang sama dengan warna RGB. Dataset kemudian dibagi menjadi 3 bagian untuk dimasukkan kedalam model klasifikasi dengan perincian pembagian data adalah 80% data *train* yaitu data yang digunakan untuk melatih model klasifikasi, 10% data validasi yaitu data yang digunakan untuk memvalidasi model klasifikasi untuk menghindari terjadinya *overfitting* dan 10% data *test* yaitu data yang digunakan untuk menguji ketepatan model klasifikasi.

### 3.5 Klasifikasi

Setelah proses data *preprocessing* selanjutnya adalah proses klasifikasi citra menggunakan arsitektur *MobileNetV2* dengan teknik *transfer learning*. Menurut [35] teknik transfer learning dapat memecahkan masalah keterbatasan data dengan menggunakan *Pre-trained* untuk melatih model pada dataset yang sedikit. Pada tahap ini terdapat dua langkah, langkah pertama adalah ekstraksi fitur dan langkah kedua adalah klasifikasi citra. Pada Proses ekstraksi fitur menggunakan *MobileNetV2* yang telah dilatih sebelumnya oleh *ImageNet* sebagai *ekstraktor* fitur. *MobileNetV2* yang telah dilatih sebelumnya pada kumpulan data *ImageNet* yang merupakan kumpulan data besar dari 1,4 juta gambar dan 1000 kelas gambar *web*, digunakan untuk mengekstrak fitur citra daun teh dalam kumpulan dataset penyakit daun teh. Selanjutnya untuk proses klasifikasi menggunakan lapisan konvolusi 1x1 *Global Average Polling* dan lapisan Dense dengan aktivasi *Softmax*. Menurut [26] parameter di *fully connecter layer* memperlambat kecepatan pelatihan jaringan untuk training dan membuat mudah *overviting*. Oleh sebab itu digunakan *Global Average Polling* untuk menghasilkan satu peta fitur pada setiap kategori klasifikasi yang langsung memasukkan ke lapisan softmax [27]. Aktiviasi *Softmax* digunakan untuk melakukan klasifikasi terhadap 6 kelas penyakit daun teh yaitu kelas daun teh sehat, *blister blight*, *mite*, *helopeltis*, ulat jengkal dan *empoasca*. Gambar 3.3 merupakan gambaran arsitektur *MobileNetV2* yang diusulkan pada penelitian ini.



Gambar 3.3. Arsitektur *MobileNetV2* yang diusulkan

Sumber : Hasil Penelitian

Tabel 3.2 menampilkan *model summary* arsitektur *MobileNetV2*.

Tabel 3.2. Tabel Model Summary *MobileNetV2*

| Input      | Operator      | t | c    | n | s |
|------------|---------------|---|------|---|---|
| 224x224x3  | conv2d        | - | 32   | 1 | 2 |
| 112x112x32 | bottleneck    | 1 | 16   | 1 | 1 |
| 112x112x16 | bottleneck    | 6 | 24   | 2 | 2 |
| 56x56x24   | bottleneck    | 6 | 32   | 3 | 2 |
| 28x28x32   | bottleneck    | 6 | 64   | 4 | 2 |
| 14x14x64   | bottleneck    | 6 | 96   | 3 | 1 |
| 14x14x96   | bottleneck    | 6 | 160  | 3 | 2 |
| 7x7x160    | bottleneck    | 6 | 320  | 1 | 1 |
| 7x7x320    | conv2d 1x1    | - | 1280 | 1 | 1 |
| 7x7x1280   | globalavgpoll | - | 1280 | 1 | - |
| 1x1x1280   | dense         | - | 6    | - | - |

Sumber : Hasil Penelitian

1. Memasukkan citra input dengan resolusi 224x224 pada arsitektur *MobileNetV2*. Resolusi ini merupakan resolusi standar dari arsitektur *MobileNetV2*.
2. Pada *Linear Residual Block (Bottleneck)* terdiri dari 3 lapisan Konvolusi yaitu *Expansion Layer*, *Depthwise Convolution* dan *Projection Layer*. *Ekspansion layer* merupakan konvolusi 1x1 yang bertujuan untuk memperluas jumlah saluran data sebelum masuk ke *depthwise convolution* sehingga *ekspansion layer* memiliki lebih banyak saluran keluaran daripada

saluran masukan, seberapa banyak data yang diperluas diberikan oleh *factor ekspansi* dan *factor ekspansi* memiliki default 6. *Depthwise convolution* melakukan penyaringan apapun yang penting pada tahap jaringan ini. Lapisan *Projection* mengembalikan data menjadi semula atau lebih kecil.

3. Resolusi *Output* dari jaringan dasar berupa citra 7x7x1280. Ukuran citra ini akan masuk ke dalam Pengklasifikasian pertama menggunakan *Global Average Polling* untuk mengurangi ukuran dari 7x7 menjadi 1x1.
4. Pada lapisan dense dengan *activation softmax* akan melakukan klasifikasi enam kelas penyakit daun teh yaitu daun teh sehat, *Blister Blight*, *Mite*, *Helopeltis*, Ulat Jengkal dan *Empoasca*.

### 3.5.1 Pelatihan

Pada proses pelatihan, nilai parameter yang diinisialisasi adalah menggunakan *categorical crossentropy* dikarenakan melakukan klasifikasi multiclass dengan *optimizer adam*, *batch size* dengan nilai 32 dan jumlah *epoch* adalah 30. Nilai batch size dan jumlah epoch yang memberikan akurasi optimum ditentukan melalui eksperimen. Oleh karena itu, pada penelitian ini dilakukan pelatihan menggunakan beberapa nilai batch size dan jumlah epoch sehingga diperoleh hasil akurasi tertinggi.

### 3.5.2 Pengujian

Setelah proses pelatihan selesai dilakukan, selanjutnya adalah menguji apakah model mampu melakukan klasifikasi citra sesuai dengan kelasnya. Proses pengujian dilakukan menggunakan 180 citra. Hasil dari pengujian akan menampilkan beberapa sample citra terhadap hasil klasifikasi yang telah dilakukan.

## 3.6 Evaluasi

Penentuan baik atau tidaknya performa suatu model klasifikasi dapat dilihat dari parameter pengukuran performanya, yaitu tingkat akurasi, sensitivitas, dan presisi. Untuk menghitung faktor-faktor tersebut diperlukan sebuah matriks yang biasa disebut *confusion matrix*. *Confusion matrix* mengandung nilai: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Seluruh

kemungkinan kejadian sebenarnya positif (P) dan seluruh kemungkinan kejadian sebenarnya negative (N). Evaluasi juga dapat dilihat pada *classification report* yang secara langsung menampilkan nilai akurasi, sensitivitas dan presisi.

### 3.7 Implementasi

Hasil penelitian diimplementasikan dalam bentuk *website*. Pembuatan *website* menggunakan *framework flask* yang merupakan *web framework* dari Bahasa pemograman *python*. Model yang telah didapatkan dari proses pelatihan akan dimasukkan ke dalam fungsi dari *framework flask* dimana model ini akan digunakan dalam proses klasifikasi pada *website*. Proses klasifikasi dilakukan dengan memasukkan satu atau lebih citra baru penyakit daun teh dan sistem akan melakukan proses klasifikasi dan menampilkan hasil klasifikasi.

### 3.8 Tools Pendukung

Penelitian ini membutuhkan *tools* pendukung baik *hardware* maupun *software* dalam melakukan penelitian dari proses awal hingga hasil. Berikut adalah spesifikasi *hardware* dan *software* yang akan digunakan dalam penelitian ini :

#### A. Hardware

Spesifikasi *Hardware* yang digunakan pada penelitian ini ditunjukkan oleh tabel 3.4.

Tabel 3.4. Spesifikasi *Hardware*

| Nama Perangkat | Spesifikasi                                            |
|----------------|--------------------------------------------------------|
| Laptop         | Laptop-K7PVPUEK                                        |
| Processor      | Intel® Core™ i3-1005G1 CPU @ 1.20GHz (4 CPUs), ~1.2Ghz |
| Harddisk       | 500 GB                                                 |
| RAM            | 4096MB                                                 |
| Handphone      | Xiomi Redmi Note 4                                     |

Sumber : Hasil penelitian

## B. *Software*

Spesifikasi *Software* yang digunakan pada penelitian ini ditunjukkan oleh tabel 3.5.

Tabel 3.5. Spesifikasi *Software*

| <b>Nama Software</b>          | <b>Spesifikasi</b>                              |
|-------------------------------|-------------------------------------------------|
| Sistem Operasi                | <i>Windows 10</i>                               |
| <i>Office</i>                 | <i>Microsoft Word 2019</i>                      |
| <i>Handphone</i>              | <i>Camera 13 Mega Pixel</i>                     |
| <i>Text Editor for python</i> | <i>Google Collaboratory</i><br><i>Notepad++</i> |
| <i>Text Editor for web</i>    | <i>Notepad++</i>                                |
| <i>Web browser</i>            | <i>Mozilla Firefox</i>                          |

Sumber : Hasil penelitian

## BAB IV

### HASIL PENELITIAN DAN PEMBAHASAN

#### 4.1 Pengumpulan Data

Citra penyakit daun teh yang diambil dari Perkebunan Nusantara VIII Kebun Gedeh Mas menggunakan kamera utama Smartphone Xiaomi Redmi Note 4 dengan resolusi 13 Megapixel menghasilkan citra dengan 6 kelas yaitu daun Sehat, *Blister Blight*, *Mite*, *Helopeltis*, Ulat Jengkal dan *Empoasca* dimana setiap kelas memiliki 100 citra daun teh sehingga total daun teh yang dihasilkan adalah sebanyak 600 citra.

Detail jumlah citra setiap kelas ditunjukkan pada Tabel 4.1.

Tabel 4.1. Jumlah Citra pada Setiap Kelas

| No | Nama Kelas            | Jumlah    |
|----|-----------------------|-----------|
| 1  | Daun Sehat            | 100 Citra |
| 2  | <i>Blister Blight</i> | 100 Citra |
| 3  | <i>Mite</i>           | 100 Citra |
| 4  | <i>Helopeltis</i>     | 100 Citra |
| 5  | Ulat Jengkal          | 100 Citra |
| 6  | <i>Empoasca</i>       | 100 Citra |

Sumber : Hasil Penelitian

Sample daun teh yang dihasilkan pada setiap kelas ditunjukkan pada Tabel 4.2.

Tabel 4.2. Sample Daun Teh

| No | Nama Kelas            | Citra                                                                                |
|----|-----------------------|--------------------------------------------------------------------------------------|
| 1  | Daun Sehat            |    |
| 2  | <i>Blister Blight</i> |    |
| 3  | <i>Mite</i>           |   |
| 4  | <i>Helopeltis</i>     |  |
| 5  | Ulat Jengkal          |  |
| 6  | <i>Empaosca</i>       |  |

Sumber : Hasil Penelitian

#### 4.2 Augmentasi Data

Augmentasi citra menggunakan teknik *Rotation* dengan sudut 30 derajat, *Shear range*, *Zoom range* dan *Horizintal Flip*. Setiap kelas dilakukan augmentasi sebanyak 100 citra dengan ukuran citra yang tersimpan adalah 224x224, format “jpg” dan warna RGB. Jumlah keseluruhan citra augmentasi adalah 600 citra.

Tabel 4.3 menunjukkan hasil augmentasi citra.

Tabel 4.3. Hasil Augmentasi Citra

| No | Nama Kelas | Citra Asli                                                                          | Hasil Augmentasi                                                                      |
|----|------------|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| 1  | Daun Sehat |    |    |
| 2  | Daun Sehat |   |   |
| 3  | Daun Sehat |  |  |
| 4  | Daun Sehat |  |  |
| 5  | Daun Sehat |  |  |

| No | Nama Kelas            | Citra Asli                                                                          | Hasil Augmentasi                                                                      |
|----|-----------------------|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| 6  | <i>Blister Blight</i> |    |    |
| 7  | <i>Blister Blight</i> |    |    |
| 8  | <i>Blister Blight</i> |    |    |
| 9  | <i>Blister Blight</i> |   |   |
| 10 | <i>Blister Blight</i> |  |  |
| 11 | <i>Mite</i>           |  |  |
| 12 | <i>Mite</i>           |  |  |

| No | Nama Kelas        | Citra Asli                                                                          | Hasil Augmentasi                                                                      |
|----|-------------------|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| 13 | <i>Mite</i>       |   |    |
| 14 | <i>Mite</i>       |    |    |
| 15 | <i>Mite</i>       |   |    |
| 16 | <i>Helopeltis</i> |   |   |
| 17 | <i>Helopeltis</i> |  |  |
| 18 | <i>Helopeltis</i> |  |  |
| 19 | <i>Helopeltis</i> |  |  |

| No | Nama Kelas        | Citra Asli                                                                           | Hasil Augmentasi                                                                      |
|----|-------------------|--------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| 20 | <i>Helopeltis</i> |     |    |
| 21 | Ulat Jengkal      |     |    |
| 22 | Ulat Jengkal      |     |    |
| 23 | Ulat Jengkal      |    |   |
| 24 | Ulat Jengkal      |   |  |
| 25 | Ulat Jengkal      |   |  |
| 26 | <i>Empoasca</i>   |  |  |

| No | Nama Kelas      | Citra Asli                                                                          | Hasil Augmentasi                                                                     |
|----|-----------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| 27 | <i>Empoasca</i> |    |   |
| 28 | <i>Empoasca</i> |    |   |
| 29 | <i>Empoasca</i> |    |   |
| 30 | <i>Empoasca</i> |  |  |

Sumber : Hasil Penelitian

### 4.3 Data Preprocessing

Data *preprocessing* pada penelitian ini adalah merubah semua ukuran citra menjadi ukuran 224x224 piksel. Dataset kemudian dibagi menjadi 3 bagian untuk dimasukkan kedalam tahap klasifikasi dengan perincian pembagian data adalah 80% data *train*, 10% data validasi dan 10% data *test*. Tabel 4.4 menunjukkan pembagian data yang digunakan pada penelitian ini.

Tabel 4.4 Pembagian Data

| No            | Jenis Data   | Presentase | Jumlah      |
|---------------|--------------|------------|-------------|
| 1             | <i>Train</i> | 80%        | 960         |
| 2             | Validasi     | 10%        | 120         |
| 3             | <i>Test</i>  | 10%        | 120         |
| <b>Jumlah</b> |              |            | <b>1200</b> |

Sumber : Hasil Penelitian

#### 4.4 Klasifikasi

Penelitian ini menggunakan arsitektur *MobileNetV2* sebagai klasifikasi penyakit pada daun teh. Pembagian data yang digunakan adalah 80% sebagai data train, 10% sebagai data validasi dan 10% sebagai data test. Optimizer yang digunakan adalah “adam”. Untuk menghasilkan akurasi yang optimal pada penelitian ini dilakukan beberapa percobaan terhadap implementasi data dan *hyperparameter* yang digunakan.

Pada tahap awal penelitian melakukan uji coba terhadap implementasi data. Uji coba yang dilakukan berupa pengujian metode menggunakan augmentasi data dan tanpa augmentasi data. Tabel 4.5 menunjukkan hasil percobaan yang dilakukan dengan menggunakan augmentasi data dan tanpa augmentasi data.

Tabel 4.5 Penelitian penentuan dengan augmentasi data dan tanpa augmentasi data

| Data              | Split Data |            |            | Optimizer   | Batch Size | Epoch     | Akurasi       |
|-------------------|------------|------------|------------|-------------|------------|-----------|---------------|
|                   | Train      | Validasi   | Test       |             |            |           |               |
| Tanpa Augmentasi  | 80%        | 10%        | 10%        | Adam        | 32         | 10        | 0.9500        |
| <b>Augmentasi</b> | <b>80%</b> | <b>10%</b> | <b>10%</b> | <b>Adam</b> | <b>32</b>  | <b>10</b> | <b>0.9667</b> |

Sumber : Hasil Penelitian

Berdasarkan hasil uji coba pada implementasi data akan dipilih berdasarkan nilai akurasi terbesar yang dihasilkan, nilai akurasi terbesar adalah dengan menggunakan augmentasi data. Kemudian nilai tersebut akan digunakan untuk percobaan selanjutnya yaitu menentukan *batch size*. Nilai *batch size* ditentukan dengan melakukan uji coba dengan memasukkan nilai *batch size* sebesar 32, 64 dan 128. Nilai ini dipilih karena merupakan ukuran populer nilai *batch size*.

Tabel 4.6 menampilkan hasil percobaan yang telah dilakukan dalam penentuan nilai *batch size*.

Tabel 4.6. Penelitian Penentuan Nilai Batch Size

| Data              | Split Data |            |            | Optimizer   | Batch Size | Epoch     | Akurasi       |
|-------------------|------------|------------|------------|-------------|------------|-----------|---------------|
|                   | Train      | Validasi   | Test       |             |            |           |               |
| <b>Augmentasi</b> | <b>80%</b> | <b>10%</b> | <b>10%</b> | <b>Adam</b> | <b>32</b>  | <b>10</b> | <b>0.9667</b> |
| Augmentasi        | 80%        | 10%        | 10%        | Adam        | 64         | 10        | 0.9583        |
| Augmentasi        | 80%        | 10%        | 10%        | Adam        | 128        | 10        | 0.9250        |

Sumber : Hasil Penelitian

Berdasarkan uji coba pada nilai batch size dipilih berdasarkan nilai akurasi terbesar yang dihasilkan, nilai akurasi terbesar yaitu dengan nilai *batch size* 32. Kemudian nilai tersebut akan digunakan untuk percobaan selanjutnya dalam menentukan jumlah *epoch* optimum.

Tabel 4.7 menunjukkan hasil percobaan yang telah dilakukan untuk penentuan jumlah epoch.

Tabel 4.7. Penelitian Penentuan Jumlah Epoch

| Data              | Split Data |            |            | Optimizer   | Batch Size | Epoch     | Akurasi       |
|-------------------|------------|------------|------------|-------------|------------|-----------|---------------|
|                   | Train      | Validasi   | Test       |             |            |           |               |
| Augmentasi        | 80%        | 10%        | 10%        | Adam        | 32         | 10        | 0.9667        |
| <b>Augmentasi</b> | <b>80%</b> | <b>10%</b> | <b>10%</b> | <b>Adam</b> | <b>32</b>  | <b>30</b> | <b>0.9833</b> |

Sumber : Hasil Penelitian

Berdasarkan uji coba pada jumlah *epoch* dipilih berdasarkan nilai akurasi terbesar yang dihasilkan yaitu dengan jumlah epoch 30. Berdasarkan percobaan, hasil terbaik dari penelitian ini adalah menggunakan arsitektur *MobileNetV2* dengan Augmentasi Data, pembagian data meliputi 80% *train*, 10% validasi dan 10% *test*, menggunakan *optimizer* “adam” dengan nilai *batch size* adalah 32 dan jumlah *epoch* adalah 30 yang dapat menghasilkan akurasi tertinggi sebesar 98,33%. Perbandingan hasil akurasi dan loss untuk data *train*, validasi dan *test* ditunjukkan oleh tabel 4.8.

Tabel 4.8. Perbandingan Nilai Akurasi dan *Loss*

| Data              | Akurasi | Nilai <i>Loss</i> |
|-------------------|---------|-------------------|
| Data <i>Train</i> | 1.0000  | 0.0179            |
| Data Validasi     | 0.9917  | 0.0307            |
| Data <i>Test</i>  | 0.9833  | 0.0380            |

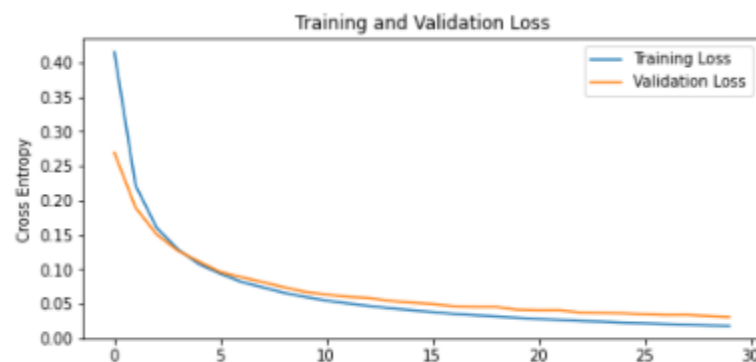
Sumber : Hasil Penelitian

Dari hasil akurasi dan nilai *loss* data *train*, data *test* dan data validasi dapat diketahui bahwa model tersebut memiliki nilai yang seimbang. Selanjutnya akan ditampilkan grafik akurasi dan nilai *loss* antara data *train* dan validasi untuk mengetahui model mengalami *overfitting*, *underfitting* atau *good fit*.

Berikut adalah grafik akurasi dan nilai *loss* pada data *train* dan data validasi:

Gambar 4.1. Grafik akurasi *training* dan validasi

Sumber : Hasil Penelitian

Gambar 4.2. Grafik akurasi loss *training* dan validasi

Sumber : Hasil Penelitian

Gambar 4.1 menunjukkan grafik akurasi dari data train dan validasi dan gambar 4.2 menunjukkan grafik akurasi *loss* dari data train dan validasi. Garis berwarna biru menunjukkan data train dan garis berwarna orange menunjukkan data validasi. Jika dilihat dari grafik akurasi dan nilai *loss* pada data *train* dan data validasi menunjukkan performa yang baik, dimana nilai akurasi mengalami kenaikan yang stabil dan nilai *loss* mengalami penurunan yang stabil yang menunjukkan bahwa model adalah *good fit* dan tidak mengalami *overviting* atau underfitting. Selanjutnya akan ditampilkan hasil akurasi melalui pengujian menggunakan data *test*.

Gambar 4.3 menunjukkan hasil klasifikasi penyakit daun teh menggunakan *MobileNetV2*.



Gambar 4.3. Hasil Klasifikasi Penyakit Daun Teh

Sumber : Hasil Penelitian

Dari 120 jumlah citra data test ditampilkan 24 sample hasil klasifikasi. Hasil klasifikasi dari 24 sample citra daun teh menunjukkan bahwa semua sample data yang diujikan terklasifikasi dengan benar. Warna biru menunjukkan bahwa data terklasifikasi dengan benar sesuai dengan kelasnya sedangkan apabila klasifikasi mengalami kesalahan akan ditunjukkan dengan warna merah.

#### 4.5 Evaluasi

Proses evaluasi dengan menggunakan Confusion Matriks dan *Classification Report*. Beberapa nilai di dalam *confusion matrix* yaitu: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Seluruh kemungkinan kejadian sebenarnya positif (P) dan seluruh kemungkinan kejadian sebenarnya *negative* (N). *Classification Report* dapat menunjukkan akurasi, *recall* dan *precision*. Tabel 4.9 menampilkan hasil *confusion matrix*.

Tabel 4.9. *Confusion Matrix*

|            |                       |                 |                       |             |                   |              |                 |
|------------|-----------------------|-----------------|-----------------------|-------------|-------------------|--------------|-----------------|
| True Label | Daun Sehat            | 28              | 0                     | 0           | 0                 | 0            | 0               |
|            | <i>Blister Blight</i> | 0               | 24                    | 0           | 0                 | 0            | 0               |
|            | <i>Mite</i>           | 0               | 0                     | 18          | 0                 | 0            | 0               |
|            | <i>Helopeltis</i>     | 0               | 0                     | 0           | 13                | 0            | 1               |
|            | Ulat Jangkal          | 1               | 0                     | 0           | 0                 | 18           | 0               |
|            | <i>Empoasca</i>       | 0               | 0                     | 0           | 0                 | 0            | 17              |
|            |                       | Daun Sehat      | <i>Blister Blight</i> | <i>Mite</i> | <i>Helopeltis</i> | Ulat Jangkal | <i>Empoasca</i> |
|            |                       | Predicted Label |                       |             |                   |              |                 |

Sumber : Hasil Penelitian

Berdasarkan tabel 4.9 hasil klasifikasi dari model terhadap data *test* menunjukkan hasil yang baik. Pada kasus ini terdapat 120 citra pengujian diantaranya :

1. 28 citra daun sehat terklasifikasi sebagai daun sehat. Total keseluruhan data *test* untuk citra daun sehat adalah 28.
2. 24 citra daun *mite* terklasifikasi sebagai daun *mite*. Total keseluruhan data *test* untuk citra daun *mite* adalah 24.
3. 18 citra daun *helopeltis* terklasifikasi sebagai daun *helopeltis*. Total keseluruhan data *test* untuk citra daun *helopeltis* adalah 18.
4. 13 citra daun *helopeltis* terklasifikasi sebagai daun *helopeltis* dan 1 citra terklasifikasi sebagai daun *empoasca*. Total keseluruhan data *test* untuk citra daun *helopeltis* adalah 14.

5. 18 citra daun ulat jengkal terklasifikasi sebagai daun ulat jengkal dan 1 citra terklasifikasi sebagai daun sehat. Total keseluruhan data *test* untuk citra daun ulat jengkal adalah 19.
6. 17 citra daun *empoasca* terklasifikasi sebagai daun *empoasca*. Total keseluruhan data *test* untuk citra *empoasca* adalah 17.

Tabel 4.10 menampilkan *classification report* yang menunjukkan hasil akurasi, *precision* dan *recall*.

Tabel 4.10. Classification Report

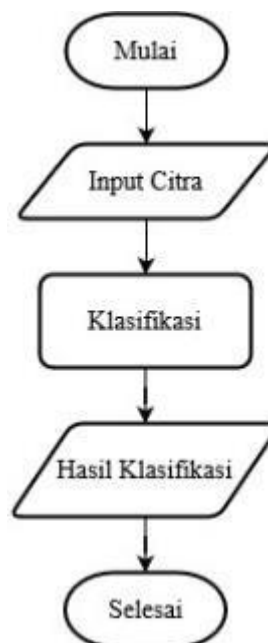
|                       | <i>Precision</i> | <i>Recall</i> | <i>F1-score</i> | <i>Support</i> |
|-----------------------|------------------|---------------|-----------------|----------------|
| Daun Sehat            | 0.97             | 1.00          | 0.98            | 28             |
| <i>Blister Blight</i> | 1.00             | 1.00          | 1.00            | 24             |
| <i>Mite</i>           | 1.00             | 1.00          | 1.00            | 18             |
| <i>Helopeltis</i>     | 1.00             | 0.93          | 0.96            | 14             |
| Ulat Jengkal          | 1.00             | 0.95          | 0.97            | 19             |
| <i>Empaosca</i>       | 0.94             | 1.00          | 0.97            | 17             |
| <i>Accuracy</i>       |                  |               | 0.98            | 120            |
| <i>Macro avg</i>      | 0.98             | 0.98          | 0.98            | 120            |
| <i>Weighted avg</i>   | 0.98             | 0.98          | 0.98            | 120            |

Sumber : Hasil Penelitian

Dari tabel *Confusion Matrix* dan *Classification Report* dapat dilihat bahwa arsitektur *MobileNetV2* dapat mengklasifikasi jenis penyakit daun teh dengan baik. Hasil akurasi yang diperoleh adalah 98%, dengan nilai *precision* 98% dan nilai *recall* 98%.

#### 4.6 Implementasi

Pada proses implementasi dihasilkan sebuah *website* menggunakan *framework flask* yang merupakan framework Bahasa pemograman *python*, *website* ini digunakan untuk klasifikasi jenis penyakit pada daun teh. Klasifikasi jenis penyakit pada daun teh menggunakan CNN dengan arsitektur MobileNetV2 menghasilkan sebuah model dengan format hdf5. Model yang dihasilkan akan digunakan dalam proses klasifikasi jenis penyakit daun teh pada *website* yang dibangun. Flowchart *website* klasifikasi jenis penyakit pada daun teh adalah seperti gambar 4.4.



Gambar 4.4. *Flowchart Website Klasifikasi Jenis Penyakit Daun Teh*

Sumber : Hasil Penelitian

Proses klasifikasi dapat dilakukan menggunakan satu atau lebih citra. Sample citra diinput kedalam sistem, sistem akan melakukan proses klasifikasi dan menunjukkan hasil klasifikasi citra sesuai dengan labelnya. Hasil klasifikasi citra akan dibandingkan dengan label aslinya, jika hasil klasifikasi sesuai maka *website* layak untuk digunakan.

*Website* klasifikasi jenis penyakit daun teh memiliki 2 halaman yaitu home atau index atau halaman awal dan halaman hasil klasifikasi.

### 1. Halaman *Index*

Halaman *index* adalah halaman awal *website*. Pada halaman ini terdapat menu untuk input citra yang akan digunakan dalam proses klasifikasi. Citra input dalam format gambar seperti “jpg”, “png”, “JPEG” dan format gambar lainnya. Tampilan halaman *index* ditunjukkan oleh gambar 4.5.



Gambar 4.5. Halaman *Index*

Sumber : Hasil Penelitian

### 2. Halaman Hasil Klasifikasi

Halaman ini menampilkan hasil klasifikasi citra yang telah diinput. Hasil klasifikasi akan menampilkan citra input dan prediksi label citra. Berikut adalah tampilan hasil klasifikasi citra :

#### a. Klasifikasi Citra Daun Teh Sehat



Gambar 4.6. Hasil Klasifikasi Citra Daun Teh Sehat

Sumber : Hasil Penelitian

b. Klasifikasi Citra Daun Teh *Blister Blight*



Gambar 4.7. Hasil Klasifikasi Citra Daun Teh *Blister Blight*

Sumber : Hasil Penelitian

c. Klasifikasi Citra Daun Teh *Mite*



Gambar 4.8. Hasil Klasifikasi Citra Daun Teh *Mite*

Sumber : Hasil Penelitian

d. Klasifikasi Citra Daun Teh *Helopeltis*



Gambar 4.9. Hasil Klasifikasi Citra Daun Teh *Helopeltis*

Sumber : Hasil Penelitian

e. Klasifikasi Citra Daun Teh Ulat Jengkal



Gambar 4.10. Hasil Klasifikasi Citra Daun Teh Ulat Jengkal

Sumber : Hasil Penelitian

f. Klasifikasi Citra Daun Teh *Empoasca*



Gambar 4.11. Hasil Klasifikasi Citra Daun Teh *Empoasca*

Sumber : Hasil Penelitian

g. Klasifikasi Citra Daun Teh Keseluruhan



Gambar 4.12 Hasil Klasifikasi Citra Daun Teh Keseluruhan

Sumber : Hasil Penelitian

Kesimpulan hasil klasifikasi ditunjukkan pada tabel 4.11.

Tabel 4.11 Kesimpulan Hasil Klasifikasi

| No | Citra                                                                               | Jenis Penyakit        | Prediksi              | Kesimpulan |
|----|-------------------------------------------------------------------------------------|-----------------------|-----------------------|------------|
| 1  |    | Daun Sehat            | Daun Sehat            | Sesuai     |
| 2  |    | Daun Sehat            | Daun Sehat            | Sesuai     |
| 3  |   | <i>Blister Blight</i> | <i>Blister Blight</i> | Sesuai     |
| 4  |  | <i>Blister Blight</i> | <i>Blister Blight</i> | Sesuai     |
| 5  |  | <i>Mite</i>           | <i>Mite</i>           | Sesuai     |
| 6  |  | <i>Mite</i>           | <i>Mite</i>           | Sesuai     |

| No | Citra                                                                               | Jenis Penyakit    | Prediksi          | Kesimpulan   |
|----|-------------------------------------------------------------------------------------|-------------------|-------------------|--------------|
| 7  |    | <i>Helopeltis</i> | <i>Helopeltis</i> | Sesuai       |
| 8  |    | <i>Helopeltis</i> | <i>Helopeltis</i> | Sesuai       |
| 9  |    | Ulat Jengkal      | Ulat Jengkal      | Sesuai       |
| 10 |   | Ulat Jengkal      | Ulat Jengkal      | Sesuai       |
| 11 |  | <i>Empoasca</i>   | <i>Empoasca</i>   | Sesuai       |
| 12 |  | <i>Empoasca</i>   | <i>Mite</i>       | Tidak Sesuai |

Sumber : Hasil Penelitian

Dari hasil percobaan klasifikasi menggunakan 12 citra penyakit daun teh baru menunjukkan bahwa 11 citra terdeteksi sesuai dan 1 citra terklasifikasi tidak sesuai, hal ini disebabkan karena karakteristik maupun bentuk citra kelas *Empoasca* yang diujikan hampir memiliki kemiripan dengan citra daun teh kelas *Mite*. Hasil ini membuktikan bahwa metode secara keseluruhan mampu melakukan klasifikasi citra daun teh dengan baik.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Dari hasil penelitian yang dilakukan diperoleh kesimpulan sebagai berikut :

1. Implementasi arsitektur *MobileNetV2* menggunakan teknik *transfer learning*. Pada Proses ekstraksi fitur menggunakan *MobileNetV2* yang telah dilatih sebelumnya oleh *ImageNet* sebagai fitur *ekstraktor*. Pada proses klasifikasi menggunakan lapisan konvolusi 1x1 *Global Average Polling* dan lapisan Dense dengan aktivasi *Softmax* untuk klasifikasi jenis penyakit daun teh.
2. Klasifikasi penyakit daun teh menghasilkan nilai akurasi sebesar 98,33%, nilai *precision* 98% dan nilai *recall* 98%. Nilai yang dihasilkan membuktikan bahwa arsitektur ini terbukti mampu digunakan untuk klasifikasi penyakit daun teh.
3. Hasil pengujian pada aplikasi berbasis *web* menggunakan data baru sebanyak 12 citra menunjukkan bahwa model dapat melakukan klasifikasi dengan baik. Model yang terbentuk diimplementasikan ke dalam sebuah aplikasi berbasis *web* agar dapat membantu dan memudahkan bagi petani teh dalam melakukan klasifikasi penyakit daun teh.

#### **5.2 Saran**

Meskipun arsitektur *MobileNetV2* yang digunakan sudah memberikan hasil yang lebih baik, namun ada beberapa hal yang dapat ditambahkan untuk penelitian selanjutnya, adapun saran-saran yang diusulkan:

1. Adanya penelitian lebih lanjut dengan data yang lebih banyak.
2. Tahapan *preprocessing* dapat menambahkan proses segmentasi dan pencerahan pada citra dengan bantuan komputer untuk mendapatkan kualitas citra yang lebih baik.

## DAFTAR REFERENSI

- [1] S. D. T. P. Statistik, *Statistik Teh Indonesia 2019*. 2019.
- [2] D. Krisnandi *et al.*, “Diseases Classification for Tea Plant Using Concatenated Convolution Neural Network,” *CommIT (Communication Inf. Technol. J.*, vol. 13, no. 2, pp. 67–77, 2019, doi: 10.21512/commit.v13i2.5886.
- [3] J. Chen, Q. Liu, and L. Gao, “Visual tea leaf disease recognition using a convolutional neural network model,” *Symmetry (Basel)*, vol. 11, no. 3, 2019, doi: 10.3390/sym11030343.
- [4] S. Mukhopadhyay, M. Paul, R. Pal, and D. De, “Tea leaf disease detection using multi-objective image segmentation,” *Multimed. Tools Appl.*, vol. 80, no. 1, pp. 753–771, 2021, doi: 10.1007/s11042-020-09567-1.
- [5] X. Sun, S. Mu, Y. Xu, Z. Cao, and T. Su, “Image Recognition of Tea Leaf Diseases Based on Convolutional Neural Network,” *2018 Int. Conf. Secur. Pattern Anal. Cybern. SPAC 2018*, pp. 304–309, 2018, doi: 10.1109/SPAC46244.2018.8965555.
- [6] S. Lee, C. Wu, and S. Chen, “Development of Image Recognition and Classification Network Written for presentation at the 2018 ASABE Annual International Meeting Sponsored by ASABE,” pp. 2–9, 2018.
- [7] S. H. Lee, C. S. Chan, S. J. Mayo, and P. Remagnino, “How deep learning extracts and learns leaf features for plant classification,” *Pattern Recognit.*, vol. 71, pp. 1–13, 2017, doi: 10.1016/j.patcog.2017.05.015.
- [8] S. Z. M. Zaki, M. A. Zulkifley, M. Mohd Stofa, N. A. M. Kamari, and N. A. Mohamed, “Classification of tomato leaf diseases using mobilenet v2,” *IAES Int. J. Artif. Intell.*, vol. 9, no. 2, pp. 290–296, 2020, doi: 10.11591/ijai.v9.i2.pp290-296.
- [9] M. R. R. Allaam and A. T. Wibowo, “KLASIFIKASI GENUS TANAMAN ANGGREK MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK ( CNN ) Program Studi Sarjana Informatika Fakultas Informatika Universitas Telkom Bandung,” *e-Proceeding Eng.*, vol. 8, no. 2, pp. 1153–1189, 2021.
- [10] R. Somantri and T. K, *Kisah dan Khasiat Teh*. 2013.
- [11] MGIP and PPTK, *Buku Persyaratan Indikasi Geografis*. 2014.
- [12] M. R. A. Yudianto, Kusriani, and H. Al Fatta, “Analisis Pengaruh Tingkat Akurasi Klasifikasi Citra Wayang dengan Algoritma Convolutional Neural Network,” *J. Teknol. Inf.*, vol. 4, no. 2, pp. 182–190, 2020.
- [13] N. Arkhamia, R. M. Awangga, and S. F. Pane, *Perbandingan Faster R-CNN dengan SSD Mobilenet untuk Mendeteksi Plat Nomor*. 2020.
- [14] P. N. Andono, T. Sutojo, and Muljono, *Pengolahan Citra Digital*. 2017.
- [15] V. Siahaan and R. H. Sianipar, *Penduan Praktis dan Komplit Pemrosesan Citra Digital*. 2020.
- [16] Luqman Hakim, Z. Sari, and H. Handhajani, “Klasifikasi Citra Pigmen Kanker Kulit Menggunakan Convolutional Neural Network,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 2, pp. 379–385, 2021, doi: 10.29207/resti.v5i2.3001.

- [17] J. Sanjaya and M. Ayub, "Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup," *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 2, pp. 311–323, 2020, doi: 10.28932/jutisi.v6i2.2688.
- [18] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut*. 2018.
- [19] I. Tahyudin, F. I. Jaya, A. Susli, N. Syafi'ah, A. M. Alfathani, and M. Purwati, *Pengenalan Machine Learning Menggunakan Jupyter Notebook*. 2020.
- [20] R. Primartha, *Belajar Machine Learning Teori dan Praktek*. 2018.
- [21] A. Soud, N. Sakli, and H. Sakli, "Classification and predictions of lung diseases from chest x- rays using mobilenet v2," *Appl. Sci.*, vol. 11, no. 6, 2021, doi: 10.3390/app11062751.
- [22] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, 2015, doi: 10.1007/s11263-015-0816-y.
- [23] Dwi Fitriana Sari and D. Swanjaya, "Implementasi Convolutional Neural Network Untuk Identifikasi Penyakit Daun Gambas," *Semin. Nas. Inov. Teknol.*, pp. 137–142, 2020.
- [24] A. Lumini and L. Nanni, "Deep learning and transfer learning features for plankton classification," *Ecol. Inform.*, vol. 51, pp. 33–43, 2019, doi: 10.1016/j.ecoinf.2019.02.007.
- [25] T. CARNEIRO, R. V. M. DA NÓBREGA, T. NEPOMUCENO, G.-B. BIAN, V. H. C. DE ALBUQUERQUE, and P. P. R. FILHO, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. 4, pp. 1–9, 2016, doi: 10.1109/ACCESS.2018.2874767.
- [26] J. Hang, D. Zhang, P. Chen, J. Zhang, and B. Wang, "Classification of plant leaf diseases based on improved convolutional neural network," *Sensors (Switzerland)*, vol. 19, no. 19, pp. 1–14, 2019, doi: 10.3390/s19194161.
- [27] G. Kang, K. Liu, B. Hou, and N. Zhang, "3D multi-view convolutional neural networks for lung nodule classification," *PLoS One*, vol. 12, no. 11, pp. 1–21, 2017, doi: 10.1371/journal.pone.0188290.
- [28] S. H. S. Herho, *Tutorial Pemrograman Python 2 Untuk Pemula*. 2017.
- [29] M. A. Pangestu and H. Bunyamin, "Analisis Performa dan Pengembangan Sistem Deteksi Ras Anjing pada Gambar dengan Menggunakan Pre-Trained CNN Model," *J. Tek. Inform. dan Sist. Inf.*, vol. 4, pp. 337–344, 2018.
- [30] D. Sarkar, R. Bali, and T. Ghosh, *Hands-on Transfer LEarning With Python*. 2018.
- [31] B. Syamsuri and G. P. Kusuma, "Plant Disease Classification using Lite Pretrained Deep Convolutional Neural Network on Android Mobile Device," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 2, pp. 2796–2804, 2019, doi: 10.35940/ijitee.b6647.129219.
- [32] S. Gayathri, D. C. J. W. Wise, P. B. Shamini, and N. Muthukumaran, "Image Analysis and Detection of Tea Leaf Disease using Deep Learning," *Proc. Int. Conf. Electron. Sustain. Commun. Syst. ICESc 2020*, no. Icesc, pp. 398–403, 2020, doi: 10.1109/ICESc48915.2020.9155850.

- [33] P. Sharma, P. Hans, and S. C. Gupta, "Classification of plant leaf diseases using machine learning and image preprocessing techniques," *Proc. Conflu. 2020 - 10th Int. Conf. Cloud Comput. Data Sci. Eng.*, pp. 480–484, 2020, doi: 10.1109/Confluence47617.2020.9057889.
- [34] R. Faurina, E. P. Purwandari, M. T. Pratama, and I. Agustian, "Klasifikasi Level Non-Proliferatif Retinopati Diabetik Dengan Ensemble Convolutional Neural Network," *Pseudocode*, vol. 8, no. 1, pp. 1–10, 2021, doi: 10.33369/pseudocode.8.1.1-10.
- [35] A. Beikmohammadi and K. Faez, "Leaf Classification for Plant Recognition with Deep Transfer Learning," *Proc. - 2018 4th Iran. Conf. Signal Process. Intell. Syst. ICSPIS 2018*, pp. 21–26, 2018, doi: 10.1109/ICSPIS.2018.8700547.
- [36] A. Ech-Cherif, M. Misbhauddin, and M. Ech-Cherif, "Deep Neural Network Based Mobile Dermoscopy Application for Triaging Skin Cancer Detection," *2nd Int. Conf. Comput. Appl. Inf. Secur. ICCAIS 2019*, pp. 1–6, 2019, doi: 10.1109/CAIS.2019.8769517.
- [37] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant Disease Detection and Classification by Deep Learning Muhammad," *MDPI*, vol. 8, no. 468, pp. 2–22, 2019, doi: 10.1109/ICBAIE49996.2020.00012.

## DAFTAR RIWAYAT HIDUP

### A. Biodata Mahasiswa

NIM : 14002374  
Nama Lengkap : Monikka Nur Winnarto  
Tempat dan Tanggal Lahir : Klaten, 26 Agustus 1997  
Jenis Kelamin : Perempuan  
Alamat : Banyuripan RT.01 RW.02 Banyuripan,  
Bayat, Klaten, Jawa Tengah

### B. Riwayat Pendidikan Formal dan Non Formal

1. SD N 1 Banyuripan, lulus tahun 2009
2. SMP N1 Bayat, lulus tahun 2012
3. SMK N 1 Klaten, lulus tahun 2015
4. D3 Universitas BSI Yogyakarta lulus tahun 2018
5. S1 Universitas BSI Bandung, lulus tahun 2019

### C. Riwayat Pengalaman Organisasi/Pekerjaan

1. Sekretaris SENAT AMIK BSI Yogyakarta Tahun Akademik 2017/2018
2. Magang Staf Riset UBSI Kampus Yogyakarta Tahun 2017 s/d 2019
3. Magang Staf Perpustakaan UBSI Kampus Cengkareng Tahun 2019
4. Staf Bagian Pengembangan Dosen UBSI Tahun 2019 s/d 2020
5. Staf Bagian Staf Rektor UBSI Tahun 2020 s/d sekarang



Jakarta, 01 Agustus 2021

(Monikka Nur Winnarto)

## SURAT KETERANGAN RISET



**PT. PERKEBUNAN NUSANTARA VIII**  
**KEBUN GEDEH MAS**  
Kantor Pusat : Jalan Sindangsari No. 04 Bandung 40133 Jawa Barat  
Telp. (022) 2626906, Fax. (022) 2621495, e-mail: ptnp@ptn.co.id, website: http://www.ptn.co.id  
Desa Sukemulya, Kecamatan Cigugur-Cangkur (43552) email : gdl-gegeh@gmail.com  
Kantor Induk : J. Raya Puncak Km 87 Cisruas - Bogor 16790 Jawa Barat  
Telp./Fax. (0251) 8252802, e-mail: gdl\_gms@gmail.com



---

Nomor : S.Ket/GUM/450/VI/2021  
Lampiran : -  
Perihal : Surat Keterangan Riset/Praktek Kerja Lapangan (PKL)

**SURAT KETERANGAN**

Yang bertanda tangan di bawah ini :

Nama : Dedi Ruswandi  
NIK : 01110675040613  
Jabatan : Administrasi I SDM & Umum

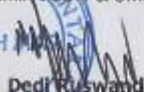
Dengan ini menerangkan bahwa, yang tersebut dibawah ini :

Nama : Monikka Nur Winnarto  
NIM : 14002374  
Perguruan Tinggi : Universitas Nusa Mandiri  
Program Studi : Ilmu Komputer

Adalah benar telah melaksanakan Riset/Praktek Kerja Lapangan dan Penelitian (Research) di PTPN VIII Kebun Teh Gedeh-Mas unit kebun Gunung Mas yang beralamat di Jalan Raya Puncak KM.87 Tugu Selatan Cisruas-Bogor, terhitung sejak tanggal 7 Juni s/d 26 Juni 2021, dan yang bersangkutan telah melaksanakan tugasnya dengan baik dan penuh tanggung jawab.

Demikian surat keterangan ini dibuat dengan benar, agar dapat dipergunakan sebagaimana mestinya.



Admin I SDM & Umum  
  
**Dedi Ruswandi**  
NIK. 01110675040613

---

*Amanah - Kompeten - Harmonis - Loyal - Adaptif - Kolaboratif*

## LAMPIRAN

### Lampiran 1

#### Dataset Daun Teh Kelas Daun Sehat



## Lampiran 2

### Dataset Daun Teh Kelas Blister Blight



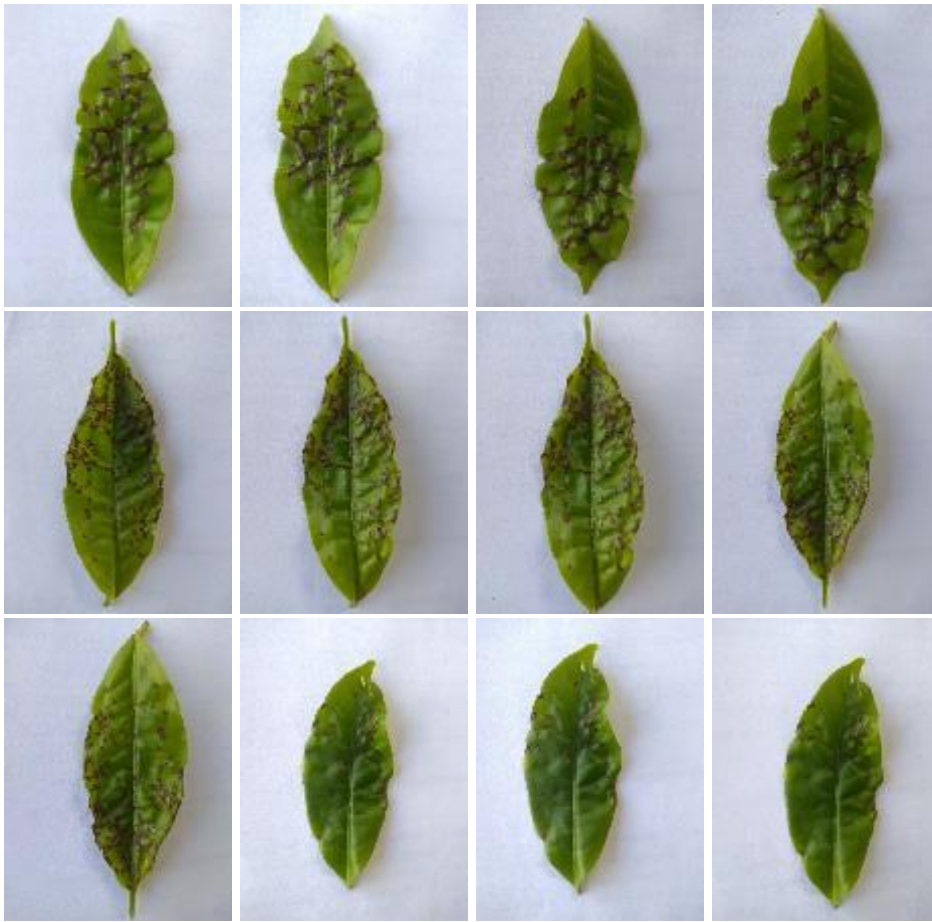
### Lampiran 3

#### Dataset Daun Teh Kelas Mite



#### Lampiran 4

#### Dataset Daun Teh Kelas Helopeltis



## Lampiran 5

### Dataset Daun Teh Kelas Ulat Jengkal



## Lampiran 6

### Dataset Daun Teh Kelas Empoasca

