

**IMPLEMENTASI *TRANSFER LEARNING* DALAM
MENGKLASIFIKASIKAN 16 JENIS BUAH DAN
SAYURAN MENGGUNAKAN ARSITEKTUR
*DENSENET 201***

SKRIPSI



NURFADLIA

H071171525

**PROGRAM STUDI SISTEM INFORMASI
DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR
2023**

**IMPLEMENTASI *TRANSFER LEARNING* DALAM
MENGKLASIFIKASIKAN 16 JENIS BUAH DAN
SAYURAN MENGGUNAKAN ARSITEKTUR
*DENSENET 201***

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana
Komputer pada Program Studi Sistem Informasi Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin**

NURFADLIA

H071171525

**PRODI SISTEM INFORMASI DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN
ALAM UNIVERSITAS HASANUDDIN**

MAKASSAR

2023

LEMBAR PERNYATAAN KEOTENTIKAN

Yang bertanda tangan dibawah ini :

Nama : Nurfadlia
NIM : H071171525
Program Studi : Sistem Informasi
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya sendiri :

Implementasi *Transfer Learning* dalam Mengklasifikasikan 16 Jenis Buah dan Sayuran Menggunakan *Arsitektur DenseNet 201*

Adalah karya tulisan saya sendiri, bukan merupakan pengambil alihan tulisan orang lain dan bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 18 Agustus 2023



Nurfadlia

NIM. H071171525

LEMBAR PENGESAHAN TUGAS AKHIR
**IMPLEMENTASI TRANSFER LEARNING DALAM
MENGKLASIFIKASIKAN 16 JENIS BUAH DAN SAYURAN
MENGUNAKAN ARSITEKTUR DENSENET 201**

Disusun dan diajukan oleh

NURFADLIA

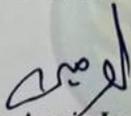
H071171525

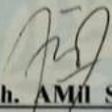
Telah dipertahankan dihadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui

Pembimbing Utama

Pembimbing Pendamping


Dr. Eng. Armin Lawi, S.Si., Eng.
NIP. 197204231995121001


A. Muh. AMil Siddik, S.Si., M.Si
NIP. 199110032019031015

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh:

Nama : Nurfadlia

NIM : H071171525

Program Studi : Sistem Informasi

Judul Skripsi : Implementasi *Transfer Learning* dalam Mengklasifikasikan 16 Jenis Buah dan Sayuran menggunakan Arsitektur *DenseNet 201*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

DEWAN PENGUJI

		Tanda Tangan
Ketua	: Dr. Eng. Armin Lawi, S.Si., M.Eng .	(.....)
Sekretaris	: A. Muh. Amil Siddik, S.Si., M.Si	(.....)
Anggota	: Dr. Andi Muhammad Anwar, S.Si, M.Si	(.....)
Anggota	: Rozalina Amran, S.T., M.Eng.	(.....)

Ditetapkan di : Makassar

Tanggal : 18 Agustus 2023



KATA PENGANTAR

Segala puji bagi Allah Subhanahu Wa ta'ala, Tuhan alam semesta yang telah memberikan nikmat kesempatan, kesehatan dan kemampuan sehingga penulisan skripsi ini bisa selesai. Shalawat serta salam senantiasa tercurah kepada Rasulullah Muhammad Shallallahu 'Alaihi Wasallam, yang merupakan teladan dalam menjalankan kehidupan di dunia.

Alhamdulillah, skripsi dengan judul :Implementasi *Transfer Learning* dalam Mengklasifikasikan 16 jenis Buah dan Sayuran Menggunakan Arsitektur *DenseNet 201*” yang disusun sebagai salah satu syarat akademik untuk meraih gelar sarjana Komputer pada Program Studi Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin ini dapat diselesaikan. Tentunya, dalam penulisan skripsi ini, penulis mampu menyelesaikan berkat bantuan dan dukungan dari berbagai pihak. Oleh karena itu, ucapan terima kasih dan apresiasi yang tak terhingga kepada kedua orang tua penulis, Ayahanda **Sunre** dan Ibunda **Sappe** yang tak kenal lelah dalam memanjatkan doa serta memberikan nasihat dan motivasi kepada penulis. Tugas akhir ini hanya setitik kebahagiaan kecil yang bisa penulis persembahkan. Tidak lupa pula terimakasih kepada saudara – saudara saya, **Sarinah, Jupri, Nasrullah, Misnawati, Andika,** dan **Muh. Asri**, yang senantiasa memberikan dukungan dan semangat.

Disamping itu, penulis juga ingin mengucapkan terima kasih dan penghargaan yang setinggi – tingginya kepada semua pihak yang telah membantu dalam proses penulisan skripsi ini diantaranya adalah:

1. Rektor Universitas Hasanuddin, **Bapak Prof.Dr.Ir. Jamaluddin Jompa, M.Sc.** beserta jajarannya.
2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA), **Dr.Eng. Amiruddin** beserta jajarannya.
3. Ketua Departemen Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam, **Prof.Dr.Nurdin, S.Si., M.Si,** dan juga **Drs. Muhammad Hasbi, M.Sc** sebagai ketua Program Studi Sistem Informasi Universitas Hasanuddin.

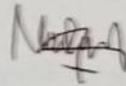
4. Bapak **Dr.Eng. Armin Lawi, S.Si, M.Eng.** sebagai pembimbing utama yang telah memberikan banyak arahan kepada penulis dalam banyak hal.
5. Bapak **A. Muh. Amil Siddik, S.Si., M.Si.** sebagai pembimbing pertama yang telah memberikan banyak arahan dan masukan kepada penulis.
6. Bapak **Andi Muhammad Anwar, S.Si., M.Si.** dan Ibu **Rozalina Amran, S.T, M.Eng.** sebagai tim penguji atas saran dan masukan pada penelitian yang telah dilakukan oleh penulis.
7. Bapak/Ibu **Dosen Departemen Matematika Universitas Hasanuddin** yang telah memberikan ilmu kepada penulis selama menjadi mahasiswa di Departemen Matematika, dan seluruh Staff departemen Matematika dan Sistem Informasi Universitas Hasanuddin yang telah membantu penulis dalam urusan berkas administrasi.
8. Sahabat **Man Jadda Wa Jadda Cica, Icha, Farda** yang telah memberikan dukungan, semangat, ide, dan motivasi kepada penulis selama menyelesaikan skripsi ini.
9. Teman – teman Program Studi **Sistem Informmasi 2017** yang telah berjuang bersama dalam suka dan duka selama ini.
10. Keluarga Besar **Ikatan Mahasiswa Muhammadiyah** atas segala cerita, pengalaman, dan pelajaran selama mengemban amanah.
11. Keluarga Besar **FLP Cabang Makassar**, atas segala cerita, kerja sama,, persahabatan, dan telah menjadi tempat untuk pengembangan kepenulisan.
12. Keluarga Besar **FLP Ranting Unhas** atas cinta, ilmu dan cerita – cerita lain yang kita lalui bersama dalam suka maupun duka.
13. Keluarga **LDM Al – Aqsho Unhas** atas kerja sama, persahabatan dan kebersamaannya selama mengemban amanah.
14. Keluarga **UKM KPI Unhas** atas segala cerita, pengalaman dan pelajaran selama mengemban amanah.

15. Teman teman KKN Pinrang 01 COVID 19 yang telah bersama menjalani pengabdian untuk mengurangi dampaknya penyebaran covid 19 yang ada di pinrang kecamatan duampunua.

16. Semua pihak yang tidak dapat di sebutkan satu persatu, atas segala bentuk bantuan, partisipasi, ide, serta motivasi yang diberikan kepada penulis selama ini. Semoga kebaikan selalu menyertai kita semua.

Penulis menyadari bahwa masih banyak kekurangan dalam tugas akhir ini, untuk itu, dengan segala kerendahan hati penulis mohon maaf sebesar-besarnya. Semoga tulisan ini memberikan manfaat untuk siapapun yang telah membacanya.

Makassar, 18 Agustus 2023



Nurfadlia

**PERNYATAAN PUBLIKASI TUGAS AKHIR UNTUK
KEPENTINGAN AKADEMIS**

Sebagai civitas akademik Universitas Hasanuddin, saya bertanda tangan dibawah ini:

Nama : Nurfadlia

NIM : H071171525

Program Studi : Sistem Informasi

Departemen : Matematika

Fakultas : Matematika dan Ilmu Pengetahuan Alam

Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Hasanuddin Hak Bebas Royalti Non Eksklusif (*Non-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul”

“Implementasi *Transfer Learning* dalam Mengklasifikasikan 16 jenis buah dan sayuran menggunakan Arsitektur *DenseNet 201*”

Beserta perangkat yang ada (jika diperlukan). Terkait dengan hal di atas, maka pihak universitas berhak menyimpan, mengalih-media/format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Makassar pada tanggal 18 Agustus 2023

Yang menyatakan


Nurfadlia

ABSTRAK

Perkembangan teknologi yang begitu pesat dalam berbagai macam hal. Salah satunya adalah membantu dalam bidang pertanian, dimana pada bidang pertanian ada beragam macam sayuran dan buah yang hampir serupa, baik dari segi ukurannya, warna, dan jenisnya. Sehingga kadangkala kesulitan untuk membedakan buah dan sayuran jika hanya berdasarkan penampakannya. Seperti pada buah apel dengan apel lainnya. Namun, teknologi semakin canggih ini tentunya dapat dikembangkan untuk dilakukannya klasifikasi jenis buah dan sayuran berdasarkan karakteristiknya. Agar memudahkan pada bidang pertanian dalam membedakan buah dan sayuran yang hampir serupa, *Convolutional Neural Network* (CNN) dapat dimanfaatkan untuk mengklasifikasikan buah dan sayuran. Pada penelitian ini arsitektur CNN yang digunakan adalah *DenseNet* 201. Penelitian ini bertujuan mengimplementasikan *transfer learning* dan menggunakan arsitektur *DenseNet* 201 dalam mengklasifikasikan 16 jenis buah dan sayuran. Performa model diukur berdasarkan nilai akurasi, F-1 *score*, luas area di bawah kurva ROC, dan kecepatan komputasi. Hasil penelitian ini menunjukkan pada arsitektur *DenseNet* memiliki performa yang sangat baik pada *learning rate* 10^{-3} dan 10^{-4} dengan akurasi mencapai 100% begitupun dengan f-1 *score*, luas area kurva ROC yang berada di angka 1. Namun, model *DenseNet* 201 dengan *learning rate* 10^{-5} memiliki performa yang buruk. Akurasi yang tidak semua berada di atas 70%. Secara keseluruhan, performa semua model sudah sangat baik, kecuali *DenseNet* 201 dengan *learning rate* 10.

Kata Kunci : Buah, Sayur, *DenseNet*-201, *Transfer Learning*, Implementasi, Performa model.

ABSTRAC

The rapid development of technology in various ways. One of them is helping in the agricultural sector, where in the agricultural sector there are various kinds of vegetables and fruit that are almost the same, both in terms of size, color and type. So that sometimes it is difficult to distinguish fruits and vegetables if only based on their appearance. As in apples with other apples. However, this increasingly sophisticated technology can certainly be developed to classify types of fruits and vegetables based on their characteristics. In order to make it easier for the agricultural sector to distinguish fruits and vegetables that are almost similar, a Convolutional Neural Network (CNN) can be used to classify fruits and vegetables. In this study, the CNN architecture used is DenseNet 201. This study aims to implement transfer learning and use the DenseNet 201 architecture to classify 16 types of fruits and vegetables. Model performance is measured based on accuracy, F-1 score, area under the ROC curve, and computational speed. The results of this study show that the DenseNet architecture has very good performance at learning rates 10^{-3} and 10^{-4} with accuracy reaching 100% as well as with f-1 scores, the area of the ROC curve is at number 1. However, the DenseNet 201 model with learning rate 10^{-5} has poor performance. Not all accuracy is above 70%. Overall, the performance of all models is very good, except DenseNet 201 with a learning rate of 10.

Keywords : Fruits, Vegetables, DenseNet-201, Transfer Learning, Implementation, Model performance.

DAFTAR ISI

HALAMAN JUDUL	ii
LEMBAR PERNYATAAN KEOTENTIKAN	iii
LEMBAR PENGESAHAN.....	iv
HALAMAN PENGESAHAN	v
KATA PENGANTAR.....	vi
PERNYATAAN PUBLIKASI TUGAS AKHIR.....	ix
ABSTRAK	x
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Terkait	5
2.2 Buah dan Sayuran.....	6
2.3 Pengolahan Citra Digital.....	6
2.4 <i>Deep Learning</i>	7
2.5 <i>Convolutional Neural Network</i>	7
2.5.1 <i>Covolutional Layer</i>	8
2.5.2 Fungsi Aktivasi <i>ReLU</i>	8
2.5.3 <i>Pooling Layer</i>	9
2.5.4 <i>Fully Connected Layer</i>	10

2.5.5 <i>Softmax Function</i>	11
2.5.6 <i>Cross Entropy Loss</i>	11
2.6 <i>DenseNet</i>	12
2.7 <i>Transfer Learning</i>	13
2.8 <i>Tensorflow</i>	13
2.9 Evaluasi Kinerja Model	14
2.9.1 <i>Confusion matrix</i>	14
2.9.2 Kurva <i>ROC</i>	15
2.10 <i>Streamlit</i>	16
2.11 <i>Google Colab</i>	16
BAB III METODE PENELITIAN	17
3.1 Tahapan Penelitian	17
3.1.1 Input Data.....	18
3.1.2 <i>Preprocessing</i>	19
3.1.3 Pelatihan Model.....	19
3.1.4 Evaluasi Model.....	19
3.1.5 Analisis Hasil	19
3.2 Waktu dan Tempat.....	19
3.3 Instrumen Penelitian	20
BAB VI HASIL DAN PEMBAHASAN.....	21
4.1 <i>Preprocessing</i>	21
4.1.1 <i>Rescale</i>	21
4.1.2 <i>Resize</i>	21
4.1.2 Pembagian Data.....	22

4.2 Arsitektur <i>DenseNet</i>	23
4.2.1 Dense Block	23
4.2.2 Transition Layer	24
4.3 Pelatihan Data.....	25
4.4 Pengujian Data	27
A. <i>Confusion Matrix</i>	28
B. Presisi.....	30
C. <i>Recall</i>	34
D. Spesifisitas	38
E. <i>F1-Score</i>	42
F. Kurva <i>ROC</i>	46
4.5 GUI Hasil Klasifikasi Citra.....	47
BAB V KESIMPULAN DAN SARAN.....	49
5.1 Kesimpulan	49
5.2 Saran	49
DAFTAR PUSTAKA.....	50
LAMPIRAN.....	53

DAFTAR GAMBAR

Gambar 2.1 Ilustrasi Arsitektur CNN pada citra mobil	7
Gambar 2.2 Grafik Fungsi <i>ReLU</i>	9
Gambar 2.3 Ilustrasi <i>Pooling layer</i>	10
Gambar 2.4 <i>Fully connected layer</i>	11
Gambar 2.5 Arsitektur <i>DenseNet</i>	13
Gambar 2.6 Kurva ROC	14
Gambar 3.1 Alur penelitian	18
Gambar 4.1 Ilustrasi gambar dengan masing-masing piksel	22
Gambar 4.2 Diagram alir <i>preprocessing</i>	23
Gambar 4.3 Diagram alir Arsitektur <i>DenseNet</i>	24
Gambar 4.4 Diagram alir <i>Dense block</i>	25
Gambar 4.5 Diagram alir <i>Transition layer</i>	26
Gambar 4.6 Grafik perubahan akurasi terhadap jumlah <i>epoch</i> pada data latih dan data validasi model dengan $lr = 1e^{-3}$	27
Gambar 4.7 Grafik perubahan akurasi terhadap jumlah <i>epoch</i> pada data latih dan data validasi model dengan $lr = 1e^{-4}$	28
Gambar 4.8 Grafik perubahan akurasi terhadap jumlah <i>epoch</i> pada data latih dan data validasi model dengan $lr = 1e^{-4}$	28
Gambar 4.9 <i>Confusion matrix learning rate</i> 10^{-3}	30
Gambar 4.10 <i>Confusion matrix learning rate</i> 10^{-4}	31
Gambar 4.11 <i>Confusion matrix learning rate</i> 10^{-5}	32
Gambar 4.12 Kurva ROC dengan <i>learning rate</i> 10^{-3}	48
Gambar 4.13 Kurva ROC dengan <i>learning rate</i> 10^{-4}	49
Gambar 4.14 Kurva ROC dengan <i>learning rate</i> 10^{-5}	49
Gambar 4.15 Hasil klasifikasi menggunakan <i>streamlit</i>	50

DAFTAR TABEL

Tabel 2.1 Confusion Matrix	15
Tabel 3.1 Dataset Penelitian.....	19
Tabel 4.1 Pembagian Data	23
Tabel 4.2 Pelatihan Data	27
Tabel 4.3 Lama Pelatihan Data	29

BAB I PENDAHULUAN

1.1 Latar Belakang

Buah merupakan bagian dari suatu tumbuhan yang berasal dari bunga atau putik dari tumbuhan tersebut dan biasanya memiliki biji sedangkan sayur merupakan daun, tumbuhan polong, atau biji – bijian yang dapat dimasak. Buah dan sayuran mengandung berbagai macam vitamin, mineral, dan serat pangan yang berperan untuk membantu proses *metabolisme* dalam tubuh (Takaoka & Kawakami, 2013).

Deep Learning (pembelajaran mendalam) merupakan suatu bidang yang ada pada *Machine Learning* yang diperkenalkan oleh Rina Dechter pada tahun 1986. Setelah itu menjadi sangat populer dalam konteks *Deep Neural Network*, Pembelajaran Jauh paling sukses, yang jauh lebih tua, sejak setengah abad (Schmidhuber, 2015). *Deep Learning* memungkinkan model komputasi yang terdiri dari beberapa lapisan pemrosesan untuk mempelajari representasi data dengan berbagai tingkatan abstrak. *Deep Learning* menemukan struktur rumit dalam kumpulan data besar dengan algoritma *backpropagation* untuk menunjukkan suatu mesin harus mengubah parameter yang digunakan untuk menghitung nilai di setiap lapisan dari nilai di lapisan sebelumnya atau prosesnya berjalan mundur pada setiap lapisannya (LeCun, Bengio, & Hinton, 2015).

Salah satu manfaat *deep learning* pada *Image processing* atau pengolahan citra adalah mampu untuk membantu manusia dalam mengenali atau mengklasifikasi suatu objek atau gambar dengan cepat, tepat dan dengan banyaknya data sekaligus secara bersamaan (Maulana & Rochmawati, 2019).

Dense Convolutional Network (*DenseNet*), yang menghubungkan setiap lapisan/blok ke lapisan/blok lainnya dengan cara umpan balik maju. Sedangkan jaringan *convolution* tradisional dengan L lapisan memiliki

diantaranya masalah *gradient* yang dapat teratasi, memperkuat lapisan, lapisan yang berulang – ulang, dan mengurangi jumlah parameter (Huang, Liu, Maaten, & Weinberger, 2017). koneksi L - satu antara setiap lapisan dan lapisan berikutnya jaringan memiliki koneksi langsung $L(L + 1) / 2$. *DenseNet* memiliki beberapa kelebihan.

Algoritma pada CNN ada banyak jenis arsitekturnya yaitu, seperti *Alex-Net*, *Google-Net*, *Le-Net*, *VVG-Net*, *Res-Net*, *DenseNet*, dan lain – lain. Penelitian ini terdahulu dilakukan oleh Ade Ramadhan dengan judul “Klasifikasi klon teh berbasis *deep* CNN dengan *residual* dan *densely connections*.” Dalam penelitiannya mengusulkan sistem identifikasi klon teh berbasis CNN dengan menerapkan metode *skip connection*, yaitu *residual connections* dan *densely connections*. Penelitian ini dilakukan karena klon teh memiliki kemiripan yang sulit dibedakan sehingga menyulitkan non pakar dalam melakukan identifikasi penyediaan dan penamaan bahan tanam pada perkebunan yang dimiliki area luas (Ramdan, Zilvan, Suryawati, Pardede, & Rahadi, 2020).

Pada penelitian ini menggunakan metode CNN jenis *DenseNet* dengan data foto *fundus* yang didapatkan dari *mesidor database* sejumlah 1.200 data yang kemudian dilakukan augmentasi sehingga menjadi 8.400 data. Hasil nilai *accuracy*, *sensitivitas*, dan *spesifisitas* pada penelitian ini sebesar 90.1%, 89.54%, dan 96.71% dengan pembagian data 80% - 20% dan jumlah *batch size* 4 (Rakamawati, 2021).

Pada penelitian tahun 2020 yaitu mengimplementasi CNN dengan arsitektur *DenseNet* 121 untuk mengidentifikasi kangker kulit melanoma. Metode yang diusulkan mampu melakukan klasifikasi kangker kulit *melanoma* dengan nilai rata – rata *accuracy*, *precision*, *recall*, dan *F-Measure* masing – masing adalah 0.94, 0.95, 0.92. dengan *preprocessing* yang digunakan pada arsitektur *DenseNet* 121 mempengaruhi dari tingkat akurasi. Berdasarkan hasil eksperimen diperoleh bahwa sistem yang diusulkan

memiliki kinerja terbaik jika dibandingkan dengan metode lain yang diusulkan.(Pardede & Putra, 2020).

Berdasarkan penjelasan di atas, maka penelitian ini berkaitan dengan klasifikasi tumbuhan menggunakan metode *Dense Convolutional Network (DenseNet)* dan *transfer learning* dengan menggunakan aktivasi *softmax* untuk mengklasifikasi 16 jenis buah dan sayuran dan penelitian ini menggunakan data citra yang diambil dari <http://www.kaggle.com>. Maka dari itu, penelitian tugas akhir ini berjudul “Implementasi *transfer learning* dalam mengklasifikasikan 16 jenis buah dan sayuran menggunakan arsitektur *DenseNet*”.

1.2 Rumusan Masalah

Berdasarkan uraian permasalahan di atas dapat disimpulkan rumusan masalahnya " Bagaimana menerapkan *transfer learning* dalam menklasifikasi kan 16 jenis buah dan sayuran dari citra dengan menggunakan metode *DenseNet*?”.

1.3 Batasan Masalah

Dalam rancangan sistem ini, pembahasan masalah dibatasi pada beberapa hal berikut:

1. Deep learning menggunakan metode *DenseNet*.
2. Menggunakan *google colab* sebagai media data *training* dan *testing*.

1.4 Tujuan Penelitian

Berdasarkan penjelasan pada latar belakang dan rumusan masalah tersebut, tujuan dalam penelitian ini adalah untuk menerapkan *transfer learning* dalam mengklasifikasi 16 jenis buah dan sayuran dari citra daun dengan menggunakan metode *DenseNet*.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dengan menerapkan *deep learning* dengan menggunakan metode *DenseNet* untuk mendeteksi 16 jenis buah dan sayuran adalah :

1. Membantu dibidang pertanian dalam mengecek jenis - jenis buah dan sayuran.
2. Mengembangkan penelitian mengenai arsitektur *DenseNet* dan *transfer learning* di lingkungan perguruan tinggi.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian terkait

Sebuah penelitian yang berjudul “Implementasi *DenseNet* untuk mengidentifikasi kanker kulit melanoma bertujuan mendeteksi kanker kulit melanoma yang terdapat pada citra dengan menggunakan metode *convolutional neural network* dengan arsitektur *DenseNet* 121”. Metode yang diusulkan mampu melakukan klasifikasi kanker kulit melanoma dengan nilai rata – rata *accuracy*, *precision*, *recall*, dan *F-Measure*, masing – masing adalah 0.94, 0.95, 0.92, dan 0.94. dengan *preprocessing* yang digunakan pada arsitektur *DenseNet* 121 mempengaruhi dari tingkat akurasi. Berdasarkan hasil eksperimen diperoleh bahwa sistem yang diusulkan memiliki kinerja terbaik jika dibandingkan dengan metode lain yang pernah diusulkan (Pardede & Putra, 2020).

Dalam penelitian lainnya berjudul “ Penerapan metode *Convolutional Neural Network* untuk klasifikasi penyakit daun apel pada *Imbalanced data*.” Penelitian ini menggunakan dataset *plant pathology 2020 – FGV C7* memiliki jumlah data sebanyak 1.821 citra daun apel yang diklasifikasikan berdasarkan kelas masing – masing. Data ini merupakan data citra berwarna yang berekstensi jpg pada setiap gambarnya. Akurasi yang didapatkan dari penelitian ini berada di angka sekitar 90% pada data uji. Hanya saja terdapat hasil yang cukup signifikan berbeda pada kelas *multiple disease*, hal tersebut terjadi dikarenakan jumlah data kelas tersebut yang memang sedikit sebelum dilakukan augmentasi sehingga data kurang *variatif* dan penanganan *imbalanced data* pada kelas tersebut kurang baik (Nisa, Puspaningrum, & Maulana, 2020).

Sebuah penelitian berjudul “Implementasi *Deep Learning* Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan

Convolutional Neural Network”. Pengujian ini membagi dataset menjadi sepuluh folder yang didalamnya terdapat 2000 citra yang dibagi menjadi citra uji dan citra latih secara acak. Citra latih pada masing-masing folder berjumlah 1800 citra, sedangkan citra uji berjumlah 200 citra. Setiap folder dilakukan tiga kali percobaan dan dihitung akurasi. Diantara tiga kali percobaan, *Fold* 10 mendapatkan rata-rata tertinggi dengan nilai akurasi 88,6 persen. Sedangkan nilai rata-rata akurasi terkecil didapatkan dari *Fold* 4 dengan nilai akurasi sebesar 83 persen. Presisi tertinggi didapatkan pada *Fold* 7 yang mendapatkan nilai akurasi sebesar 90 persen pada percobaan pertama dari tiga percobaan (Ilahiyah & Nilogiri, 2018)

2.2 Buah dan sayuran

Buah dan sayuran memiliki berbagai macam jenis, yang memiliki bentuk dan warna yang berbeda dari setiap jenisnya. Di antara keanekaragamannya, tentu ada beberapa jenis buah maupun sayuran yang memiliki bentuk dan warna yang hampir serupa namun memiliki kandungan vitamin dan gizi yang berbeda, misalnya buah apel dengan apel yang lain, masing-masing memiliki ukuran, warna, tekstur, bentuk, dan lain-lain. Namun terkadang masih banyak kesulitan yang terjadi dalam membedakan antar jenis buah dan sayur yang memiliki bentuk dan warna yang hampir serupa tersebut. Perkembangan teknologi tentunya dapat dikembangkan untuk membantu melakukan klasifikasi suatu objek berdasarkan karakteristik yang dimiliki oleh objek tersebut (Yohannes, Pribadi, & Chandra, 2020).

2.3 Pengolahan Citra Digital

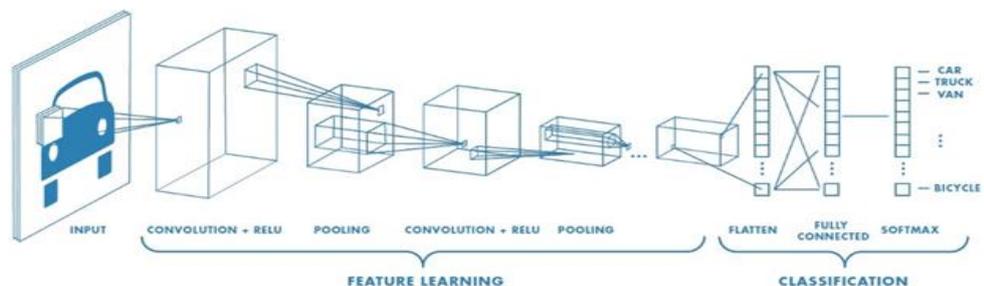
Pengolahan *Citra Digital (Digital Image Processing)* merupakan bidang ilmu yang mempelajari tentang bagaimana suatu citra itu dibentuk, diolah, dan dianalisis sehingga menghasilkan informasi yang dapat dipahami oleh manusia.

2.4 Deep Learning

Deep Learning (pembelajaran mendalam) merupakan suatu bidang yang ada pada *Machine Learning* yang di perkenalkan oleh Dechter pada tahun 1986. Setelah itu menjadi sangat populer dalam konteks *Deep Neural Network*, Pembelajar Jauh paling sukses, yang jauh lebih tua, sejak setengah abad. *Deep Learning* adalah penggunaan layer yang jumlahnya dapat mencapai ratusan sehingga dapat disebut “*deep*”. *Deep Learning* memungkinkan model komputasi yang terdiri dari beberapa lapisan pemrosesan untuk mempelajari *representasi* data dengan berbagai tingkatan abstrak. *Deep Learning* menemukan struktur rumit dalam kumpulan data besar dengan algoritma *backpropagation* untuk menunjukkan suatu mesin harus mengubah parameter yang digunakan untuk menghitung nilai di setiap lapisan dari nilai di lapisan sebelumnya atau proses nya berjalan mundur pada setiap lapisannya (LeCun, Bengio, & Hinton, 2015).

2.5 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu pengembangan dari *Artificial Neural Network* (ANN). CNN merupakan algoritma *neural network* yang digunakan untuk input berupa gambar atau video (Santosa, 2018). Salah satu keuntungan dari CNN yaitu mengurangi jumlah parameter dalam *neural network* (Albawi, 2017). Berikut contoh Arsitektur CNN dapat dilihat pada gambar 2.1



Gambar 2.1 Ilustrasi Arsitektur CNN pada citra mobil

Pada Gambar 2.1, Tahapan pada CNN secara garis besar dibagi menjadi 2, yaitu *feature learning* dan *fully connected layer*. *Feature learning* digunakan sebagai pembelajaran fitur pada sebuah gambar, pada *layer* ini akan dilakukan *convolutional layer*, *Rectified Linear Unit (ReLU) layer*, dan *pooling layer*. Hasil dari *feature learning* dimasukkan kedalam *fully connected layer* dan dilakukan aktivasi fungsi *softmax* sehingga memperoleh nilai probabilitas dari masing-masing kelas (Asmara dkk, 2020).

2.5.1 Convolutional Layer

Pada *convolution layer* dilakukan *ekstraksi fitur* pada citra dengan melakukan proses konvolusi antara *filter matrix* dengan input citra. Dengan penggunaan banyak tingkat *layer* dan *filter matrix* yang berbeda, maka akan didapatkan fitur di level tinggi seperti *edge curve* dan fitur warna (Hariyani, 2020).

Berikut rumus *convolutional* pada persamaan (2.1).

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-n,j-m) \quad (2.1)$$

Keterangan:

S(i,j) = Fungsi hasil konvolusi

I = Input

K = Kernel atau Filter

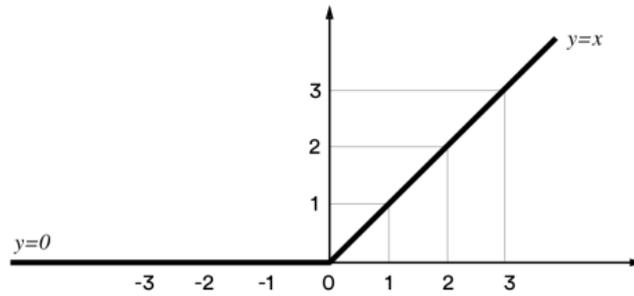
i,j = Pixel input

m,n = Pixel filter

2.5.2 Fungsi Aktivasi ReLU

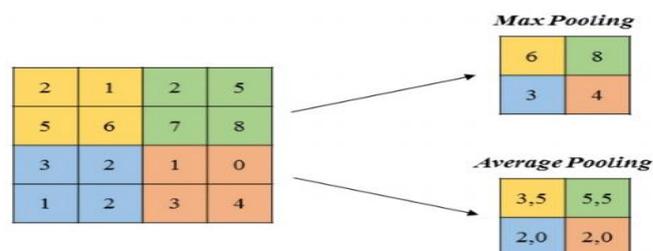
Rectified Linear Units (ReLU) untuk fungsi aktivasi yang diperkenalkan oleh Geoffrey Hinton dan Vinod Nair dan digunakan pada *neural network*, digunakan untuk mengubah nilai x menjadi 0 jika nilai x tersebut bernilai *negatif*, sedangkan sebaliknya untuk nilai x tetap dipertahankan apabila nilai tidak kurang dari 0 (Agarap, 2018).

Berikut dapat dilihat grafik Fungsi *ReLU*

Gambar 2.2 Grafik Fungsi *ReLU*

2.5.3 Pooling Layer

Pooling Layer merupakan lapisan yang menggunakan fungsi dengan *feature map* sebagai masukan dan mengolahnya dengan berbagai macam operasi statistik berdasarkan nilai piksel terdekat. *Pooling layer* pada model CNN biasanya disisipkan secara teratur setelah beberapa *convolution layer*. *Pooling layer* yang dimasukkan di antara lapisan konvolusi secara berturut-turut dalam arsitektur model CNN dapat secara *progresif* mengurangi ukuran volume *output* pada *feature map*, sehingga jumlah parameter dan perhitungan di jaringan berkurang, serta untuk mengurangi kemungkinan *overfitting*. *Pooling layer* digunakan untuk mengambil nilai maksimal (*max-pooling*) atau nilai rata-rata (*average pooling*) dari bagian-bagian piksel pada citra dalam satu *kernel* (Ilahiyah S. & Nilogiri., 2018). Berikut ilustrasi *pooling layer* di bawah ini!

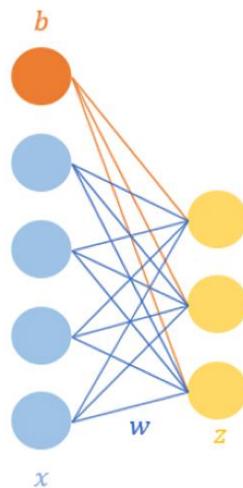
Gambar 2.3 Ilustrasi *Pooling layer*

Pooling layer berfungsi untuk memperkecil ukuran *matrix* tetapi tidak menghilangkan informasi penting (Albert, Gunadi, & Setyati, 2020). *Pooling layer* yang digunakan terdapat dua jenis, yaitu *average pooling* dan *max pooling*. *Average pooling* merupakan pengambilan nilai berdasarkan nilai rata-rata dari setiap *grid*. Sedangkan *max pooling* merupakan pengambilan nilai berdasarkan nilai terbesar (maksimum) dari setiap *grid* (Zhi, Duan, & Huang, 2016). Contoh *average pooling* dan *max pooling* ditunjukkan pada Gambar 2.3

2.5.4 Fully Connected Layer

Fully connected layer merupakan bagian pada CNN yang mirip dengan neural network biasa. Setiap *node* dalam lapisan ini saling terhubung langsung dengan setiap *node* di lapisan sebelum dan berikutnya. *Fully connected layer* ini merupakan lapisan yang paling banyak menggunakan parameter dan membutuhkan waktu komputasi yang tinggi saat *training* (Albawi, 2017).

Nilai yang digunakan untuk proses *fully connected layer* adalah keluaran dari *feature map*. Perhitungan *fully connected layer* didefinisikan pada Persamaan 2.2 dan Contoh Gambar pada 2.4.



Gambar 2.4 Fully Connected Layer

$$z_j = b_j + \sum_i w_{ij}x_i \quad (2.2)$$

Keterangan:

x = merupakan nilai input

w = merupakan bobot

b = merupakan bias dari jaringan

z = merupakan *output* dari *connected layer*

2.5.5 Softmax Activation

Softmax merupakan fungsi aktivasi yang digunakan pada *output layer* terakhir untuk menormalkan jaringan ke distribusi *probabilitas* sehingga nilai keluaran tiap kelas berada dalam interval 0-1 dan jumlah dari setiap keluaran sama dengan satu (Mahmud K. H., 2019). Nilai *probabilitas softmax* dapat digunakan dengan Persamaan 2.3.

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (2.3)$$

Dimana nilai z_j merupakan hasil dari *fully connected layer*.

2.5.6 Cross Entropy Loss

Cross entropy merupakan fungsi untuk kerugian dan *gradien* digunakan dalam *multi-classification*. Untuk mengukur *entropi relatif* antara dua distribusi probabilitas pada data yang sama. *Cross entropy* fungsi kerugian yang digunakan ketika melatih model *neural network*. Cara kerjanya mengurangi *log negatif* dari dataset (Murphy, 2018). Digunakan seperti pada persamaan 2.4 berikut.

$$H(p, q) = - \sum_{x \in X} p(x) \log q(x) \quad (2.4)$$

Keterangan:

$$H(p, q) = \text{Cross Entropy Loss}$$

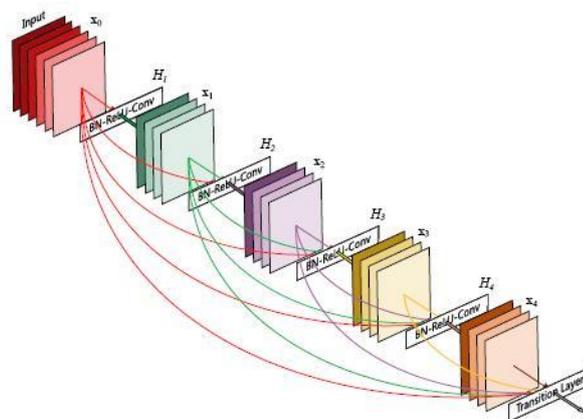
$p(x)$ = Label kelas sebenarnya (*ground truth*)

$q(x)$ = Output prediksi dari Network

2.6 Dense Neural Network (*DenseNet*)

Dense Convolutional Network (DenseNet) *Dense Neural Network* atau dikenal dengan *DenseNet* merupakan salah satu jenis arsitektur CNN pengembangan dari *ResNet*. *DenseNet* memiliki arsitektur yang menghubungkan setiap *layer* ke *layer* lain dengan cara *feed-forward*, sehingga fitur yang didapatkan lebih beragam dan kompleks, serta parameter dan efisiensi komputasi lebih cepat karena telah mempelajari dari pola sebelum-sebelumnya seperti pada ilustrai *Dense Blok* (Huang, Liu, Maaten, & Weinberger, 2017).

Dapat dilihat Arsitektur *Dense Neural Network*



Gambar 2.5 Arsitektur *DenseNet*

Pada gambar 2.5 untuk setiap komposisi lapisan menggunakan *batch normalization*, *ReLU activation* dan *convolution* dengan filter 3×3 . Pada setiap blok ada masukan berupa matriks sesuai dengan pixel citra kemudian masuk ke proses *batch normalization* untuk mengurangi adanya *overfitting* pada saat proses *training*, *ReLU activation* digunakan untuk mengubah nilai

x menjadi 0 jika nilai x tersebut bernilai *negatif*, sedangkan sebaliknya untuk nilai x tetap dipertahankan apabila nilai tidak kurang dari 0, *convolution* dengan filter 3x3 proses citra *matriks* yang sudah dilakukan operasi *ReLU activation* akan dikalikan dengan matriks *convolution* dengan filter 3x3 dan keluaran yang dihasilkan berupa nilai *matriks* yang sudah di proses sebelumnya (Pardede & Putra, 2020).

2.7 Transfer Learning

adalah metode menggunakan jaringan saraf yang sudah dilatih sebelumnya lalu mengurangi jumlah parameter dengan cara mengambil beberapa bagian dari model yang sudah dilatih untuk digunakan dalam mengenali model baru (Abas, Ismail, Yassin, & Taib, 2018).

Ada dua cara pendekatan dasar untuk melatih model *CNN*, yakni dengan membentuk jaringan dari awal (*Scratch*) atau dengan pembelajaran transfer (*Transfer Learning*) (Altuntas, Comert, & Kocamaz, 2019).

Dalam pendekatan pembelajaran *transfer* yang umum, lapisan konvolusi digunakan sebagai fitur *ekstraktor* yang tetap, dan hanya *fully-connected layer* yang disesuaikan dengan tugas klasifikasi yang baru. *Fine-tuning* dimulai dengan mentransfer bobot dari jaringan yang telah dilatih (*pre-trained network*) ke jaringan yang baru yang akan dilatih. Lapisan-lapisan terakhir termasuk di dalam nya *fully-connected layer*, *softmax*, dan *layer* klasifikasi dipisahkan dari jaringan, kemudian konfigurasi baru direalisasikan untuk tugas spesifik baru yang akan dilakukan (Altuntas, Comert, & Kocamaz, 2019).

2.8 TensorFlow

TensorFlow adalah *framework machine learning* yang bekerja dalam skala besar dan dalam *environment* yang *heterogeneous* (Abadi, Agarwal, Barham, Brevdo, Chen, & Citro, 2016). *TensorFlow* digunakan untuk melakukan eksperimen model *deep learning*, melatih model pada *dataset* yang berukuran besar, dan membuatnya layak diproduksi. Selain itu,

TensorFlow juga mendukung training dan *inference* berskala besar dengan menggunakan ratusan server yang menggunakan *Graphic Processing Unit (GPU)* untuk *training* secara *efisien* (Pangestu & Bunyamin, 2018). Penelitian ini menggunakan *TensorFlow* sebagai *backend framework* untuk Keras.

2.9 Evaluasi Kinerja Model

Untuk melakukan evaluasi terhadap model yang telah dibuat, perlu diukur performa dari model tersebut. Secara umum, yang digunakan sebagai ukuran performa model klasifikasi adalah akurasi. Akurasi adalah jumlah prediksi benar untuk semua kelas dibagi dengan jumlah datanya. Namun, untuk mengukur performa lebih lanjut, ukuran yang sangat populer digunakan adalah dengan menggunakan *confusion matrix* (Sentosa & Umam, 2018).

2.9.1 Confusion Matrix

Pada saat melakukan klasifikasi diperlukan evaluasi untuk mengetahui seberapa besar kinerja dari model tersebut. Salah satu model evaluasi yang dapat digunakan pada sistem klasifikasi adalah *confusion matrix*. Pada *confusion matrix* sering dikenal dengan 4 istilah, yaitu *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negatif*. *Confusion Matrix* yang digunakan adalah multi *class* karena akan melakukan evaluasi terhadap klasifikasi yang lebih dari dua kelas yang satu dianggap negatif dan lainnya dianggap positif (Manliguez, 2016).

Tabel 2.1 *Confusion Matrix*

		Prediksi			
		Kelas 1	Kelas 2	Kelas n
Aktual	Kelas 1	x_{11}	x_{12}	x_{1n}
	Kelas 2	x_{21}	x_{22}	x_{2n}

	Kelas n	x_{n1}	x_{n2}	x_{nn}

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.5}$$

$$Precision = \frac{TP}{TP+FP} \quad (2.6)$$

$$Recall = \frac{TP}{TP+FN} \quad (2.7)$$

$$F_1 \text{ Score} = \frac{2 * recall * precision}{recall + precision} \quad (2.8)$$

Keterangan :

TP = Jumlah *True Positive*

TN = Jumlah *True Negative*

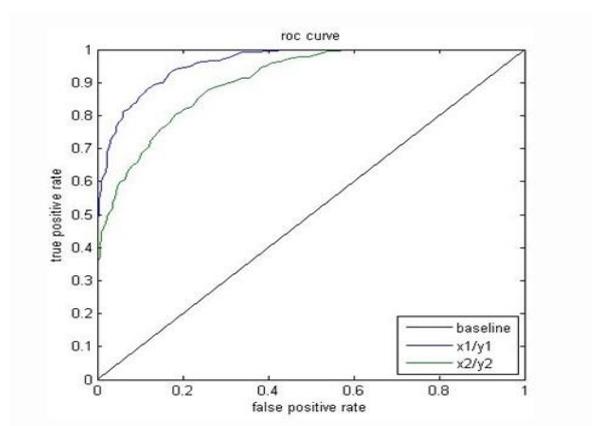
FP = Jumlah *False Positive*

FN = Jumlah *False Negative*

2.9.2 Kurva ROC

Kurva *ROC* adalah *representasi grafis* dari hubungan antara *sensitivitas* dan *1-spesifisitas*. Dalam penelitian medis kurva *ROC* banyak digunakan untuk menggambarkan keakuratan *diagnostik* dan menentukan nilai *cut-off* yang optimal. Keakuratan *diagnosis* berasal dari *area* di bawah kurva *ROC* dan *optimal cut-off* digunakan untuk mengidentifikasi kondisi *positif* dan *negatif* dalam, *diagnosis* (Kusmantoro, 2018).

Berikut contoh Kurva ROC.



Gambar 2.6 Kurva ROC

Pada gambar 2.6, Kurva ROC menunjukkan dalam membaca kinerja klasifikasinya adalah jika garis *baseline* atau garis yang melintang dari titik 0,0 dianggap jelek. Namun, jika garis melintang dari titik 0,1 maka bagus. Berdasarkan kurva diatas dapat disimpulkan kinerja kurva berwarna biru lebih bagus dibandingkan kinerja kurva berwarna hijau. Dengan cara seperti ini maka kita dengan mudah membandingkan bagus atau tidaknya teknik-teknik yang digunakan untuk menyelesaikan kasus klasifikasi pada suatu percobaan (Faisal, 2017).

2.10 Streamlit

Streamlit adalah sebuah *framework* berbasis *Python* dan bersifat *open-source* yang dibuat untuk memudahkan dalam membangun aplikasi *web* di bidang sains data dan *machine learning* yang interaktif. Salah satu hal menarik dari *framework* ini adalah kita tidak perlu mengetahui banyak hal tentang teknologi *web development*. Kita tidak perlu dipusingkan tentang bagaimana mengatur tampilan *website* dengan *CSS*, *HTML*, atau *Javascript*. Untuk menggunakan *Streamlit*, kita cukup memiliki modal dasar mengetahui bahasa *Python* saja (Yulianto, 2021).

2.11 Google Colab

Google Colab atau *Google Colaboratory*, adalah sebuah *executable document* yang dapat digunakan untuk menyimpan, menulis, serta membagikan program yang telah ditulis melalui *Google Drive*. Yang dijalankan secara *online* serta gratis. Selain itu, menyediakan pilihan untuk menjalankan atau *runtime* koding yaitu: *GPU*, *TPU*, dan *None* (Oliver, 2022).