

**EVALUASI KINERJA MODEL TRANSFER LEARNING
MENGUNAKAN ENSEMBLE PADA KLASIFIKASI PENYAKIT DAUN
TOMAT**

SKRIPSI



**MUHAMMAD ISLAHFARI WAHID
H071181501**

**Pembimbing Utama : Dr. Eng. Armin Lawi, S.Si., M.Eng.
Pembimbing Pendamping : A. Muh. Amil Siddik, S.Si., M.Si.
Penguji : 1. Drs. Muhammad Hasbi, M.Sc.
2. Andi Muhammad Anwar, S.Si., M.Si.**

**PROGRAM STUDI SISTEM INFORMASI
DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR
2022**

**EVALUASI KINERJA MODEL TRANSFER LEARNING
MENGUNAKAN ENSEMBLE PADA KLASIFIKASI PENYAKIT DAUN
TOMAT**

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**

MUHAMMAD ISLAHFARI WAHID

H071181501

**PROGRAM STUDI SISTEM INFORMASI
DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN
ALAM UNIVERSITAS HASANUDDIN
AGUSTUS 2022**

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini

Nama : Muhammad Islahfari Wahid

NIM : H071181501

Program Studi : Sistem Informasi

Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul :

**EVALUASI KINERJA MODEL TRANSFER LEARNING
MENGUNAKAN ENSEMBLE PADA KLASIFIKASI PENYAKIT DAUN
TOMAT**

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, 31 Agustus 2022



Muhammad Islahfari Wahid

H071181501

**EVALUASI KINERJA MODEL TRANSFER LEARNING
MENGUNAKAN ENSEMBLE PADA KLASIFIKASI PENYAKIT DAUN
TOMAT**

Disusun dan diajukan oleh :

MUHAMMAD ISLAHFARI WAHID

H071181501

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.

UNIVERSITAS HASANUDDIN

Pembimbing Utama

Pembimbing Pertama


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


A. Muh Amil Siddik S. Si., M.Si.
NIP. 199102242018016001

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Muhammad Islahfari Wahid
NIM : H071181501
Program Studi : Sistem Informasi
Judul Skripsi : Evaluasi Kinerja Model Transfer Learning
Menggunakan Ensemble Pada Klasifikasi
Penyakit Daun Tomat

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

1. Ketua: Dr. Eng. Armin Lawi, S.Si., M.Eng.

(.....)

2. Sekretaris: A. Muh Amil Siddik, S.Si., M.Si.

(.....)

3. Anggota: Drs. Muhammad Hasbi, M.Sc.

(.....)

4. Anggota: A. Muhammad Anwar, S.Si., M.Si.

(.....)

Ditetapkan di : Makassar

Tanggal : Agustus 2022



KATA PENGANTAR

Segala puji bagi Allah *Subhanahu Wa ta'ala*, Tuhan atas langit dan bumi beserta segala isinya. Karena, berkat nikmat dan karuniaNYA sehingga penulisan skripsi ini dapat terselesaikan. Shalawat serta salam semoga senantiasa tercurahkan kepada Baginda *Rasulullah Muhammad Shallallahu Alaihi Wasallam* dan kepada para keluarga serta sahabat beliau, yang senantiasa menjadi teladan yang baik.

Alhamdulillah, skripsi dengan judul “Rancang Bangun Aplikasi *Mobile* Klasifikasi Kanker Kulit Dengan Pemilihan Model *Transfer Learning*” yang disusun sebagai salah satu syarat akademik untuk meraih gelar Sarjana Sains pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin ini dapat dirampungkan. Tentunya, dalam penulisan skripsi ini, penulis mampu melewati berbagai hambatan dan masalah berkat bantuan moril dan materil, serta dorongan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih yang tak terhingga kepada orang tua penulis sebagai tempat kembali setelah pergi, terima kasih atas kasih sayang, doa, dan nasihat yang tulus sebagai bekal kehidupan. Rasa terima kasih juga penulis tujukan kepada saudara(i) tercinta yang telah menjadi motivator, dan rival dalam membanggakan kedua orang tua, terima kasih atas dukungan yang penulis dapatkan selama ini.

Penghargaan dan ucapan terima kasih dengan penuh ketulusan juga penulis ucapkan kepada:

1. Rektor Universitas Hasanuddin Makassar **Prof. Dr. Ir. Jamaluddin Jompa, M.Sc.**, dan seluruh Wakil Rektor dalam Lingkungan Universitas Hasanuddin.
2. Bapak Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam **Dr. Eng Amiruddin** dan para Wakil Dekan serta seluruh staf yang telah memberikan bantuan selama penulis mengikuti pendidikan di FMIPA Universitas Hasanuddin.
3. Bapak **Dr. Nurdin, S.Si. M.Si**, sebagai Ketua Departemen Matematika FMIPA Unhas. Penulis juga berterima kasih atas dedikasi dosen-dosen pengajar, serta staf Departemen atas ilmu dan bantuan yang bermanfaat.

4. Bapak **Dr. Muhammad Hasbi, M.Sc** sebagai Ketua Program Studi Sistem Informasi Universitas Hasanuddin.
5. Bapak **Dr. Eng. Armin Lawi, S.Si., M.Eng.**, dan Bapak **A. Muh Amil Siddik S. Si., M.Si** sebagai dosen pembimbing utama dan dosen pembimbing pertama atas ilmu yang beliau berikan selama proses perkuliahan, dan kesediaan beliau dalam membimbing, serta memotivasi penulis dalam penyusunan skripsi ini.
6. Bapak **Drs. Muhammad Hasbi, M.S.c.** dan Bapak **A. Muhammad Anwar, S.Si., M.Si** sebagai dosen penguji pertama dan penguji kedua atas ilmu yang beliau berikan selama proses perkuliahan, dan saran serta masukan yang telah beliau berikan dalam penyusunan skripsi ini.
7. Kepada saudara(i) ku **Cecilia, Ajrana, Fuad, Nasrullah, Maxi** yang telah senantiasa membantu penulis dalam penyusunan skripsi. Juga kepada saudara **Rifky, Ramdan, Maxi, Luthfi, dan Ulis** yang telah menjadi *partner* tugas matakuliah mulai dari semester 1 sampai dengan semester akhir. Semangat dan motivasi juga di berikan oleh teman-teman **Sistem Informasi Unhas 2018** terimakasih atas kebersamaan, kepedulian, suka-duka, canda tawa yang telah kita lewati selama ini.
8. Kepada **kak Khaiz** yang telah meluangkan waktu dalam membagi ilmu. Terimakasih atas segala ilmu yang telah diberikan.
9. Saudara(i) ku Seperjuangan Ruang Belajar (**Nasrullah, Djihad, Fuad, Ilham, Raynaldi**, serta teman-teman yang lain) terimakasih telah saling menguatkan, Semoga kesuksesan selalu kita dapatkan dalam setiap langkah-langkah kita.
10. Semua pihak yang telah memberikan bantuan kepada penulis baik berupa materi dan non materi yang tidak dapat penulis sebutkan satu per satu, terima kasih untuk bantuan dan dukungannya.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan karena keterbatasan penulis. Oleh karena itu, saran dan kritik demi penyempurnaan skripsi ini sangat penulis harapkan. Akhir kata, semoga skripsi ini membawa manfaat dan semoga Allah Subhanahu Wata'ala membalas semua kebaikan semua pihak yang telah membantu.

Makassar, Agustus 2022

A handwritten signature in black ink, consisting of a stylized 'M' and 'W' with a large loop at the top.

Muhammad Islahfari Wahid

PERNYATAAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai civitas akademik Universitas Hasanuddin, saya yang bertanda tangan di bawah ini:

Nama : Muhammad Islahfari Wahid
NIM : H071181501
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:

“Evaluasi Kinerja Model Transfer Learning Menggunakan Ensemble Pada Klasifikasi Penyakit Daun Tomat”

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 31 Agustus 2022

Yang menyatakan



Muhammad Islahfari Wahid

ABSTRAK

Tomat merupakan salah satu sayuran yang umum dikonsumsi di dunia. Tanaman tomat (*Lycopersicon esculentum* Mill.) termasuk famili Solanaceae merupakan salah satu komoditas sayuran yang sangat potensial untuk dikembangkan. Produksi tomat di Indonesia sendiri sangat luar biasa, pada tahun 2021 produksi tomat mencapai 1.035.475 (ton). Dan konsumsi tomat di Indonesia mencapai 1.053.249 (ton), artinya tomat menjadi salah satu buah yang sangat dibutuhkan di Indonesia. Namun karena kebutuhan yang tinggi seringkali terjadi kenaikan harga yang signifikan pada tomat. Hal ini dikarenakan berbagai jenis penyakit yang menghambat produksi tanaman tomat. Oleh karena itu dibutuhkan sebuah metode yang dapat mengidentifikasi penyakit tanaman tomat dengan hasil yang optimal. Metode yang digunakan adalah Deep Learning dengan menggunakan algoritma Convolutional Neural Network (CNN) beserta arsitektur InceptionV3, Xception dan VGG16. Untuk meningkatkan performa model dapat digunakan metode Ensembled Stacking. Deep Learning adalah sebuah metode machine learning yang bekerja dengan cara meniru system kerja otak manusia, system ini disebut Neural Network. CNN merupakan salah satu jenis dari neural network, yang fungsi utamanya digunakan untuk data citra. Ensembled Stacking merupakan metode yang menggabungkan beberapa model untuk mendapatkan satu model dengan performa terbaik. Penelitian dilakukan dengan 700 data citra penyakit daun tomat yang terbagi menjadi 7 kelas dengan masing-masing kelas memiliki 100 data citra. Dalam penelitian ini dilakukan pembagian data dengan rasio 8:2, data training sebanyak 80% dan data testing sebanyak 20%. Dari tiga arsitektur yang digunakan, model Xception mendapatkan akurasi training sebesar 99% dan akurasi validation sebesar 95%. Untuk model InceptionV3 mendapatkan akurasi training sebesar 99% dan akurasi validation sebesar 91%. Dan untuk model VGG16 mendapatkan hasil paling rendah dibandingkan dua arsitektur lainnya yaitu akurasi training sebesar 87% dan akurasi validation sebesar 90%. Kemudian selanjutnya digunakan metode Ensembled Stacking untuk menggabungkan tiga model arsitektur yang sudah dibuat. Hasil dari metode ini lebih baik dibandingkan tiga model arsitektur sebelumnya dengan mendapatkan akurasi training sebesar 99% dan akurasi validation sebesar 99%.

Kata kunci: *Convolutional Neural Network (CNN)*, *Deep Learning*, pengolahan citra, *Ensemble*, penyakit daun tomat.

ABSTRACT

*Tomato is one of the vegetables commonly consumed in the world. Tomato (*Lycopersicum esculentum* Mill.) belongs to the Solanaceae family, which is one of the most potential vegetable commodities to be developed. Tomato production in Indonesia itself is extraordinary, in 2021 tomato production will reach 1,035,475 (tons). And the consumption of tomatoes in Indonesia reaches 1,053,249 (tons), meaning that tomatoes are one of the most needed fruits in Indonesia. However, due to high demand, there is often a significant increase in the price of tomatoes. This is due to various types of diseases that inhibit the production of tomato plants. Therefore we need a method that can identify tomato plant diseases with optimal results. The method used is Deep Learning using the Convolutional Neural Network (CNN) algorithm along with the InceptionV3, Xception and VGG16 architectures. To improve model performance, the Ensembled Stacking method can be used. Deep Learning is a machine learning method that works by imitating the working system of the human brain, this system is called a Neural Network. CNN is one type of neural network, whose main function is used for image data. Ensembled Stacking is a method that combines several models to get one model with the best performance. The study was conducted with 700 image data of tomato leaf disease which was divided into 7 classes with each class having 100 image data. In this study, the distribution of data with a ratio of 8:2, training data as much as 80% and testing data as much as 20%. Of the three architectures used, the Xception model gets a training accuracy of 99% and a validation accuracy of 95%. For the InceptionV3 model, the training accuracy is 99% and the validation accuracy is 91%. And for the VGG16 model, the lowest results compared to the other two architectures are training accuracy of 87% and validation accuracy of 90%. Then, the Ensembled Stacking method is used to combine the three architectural models that have been created. The results of this method are better than the three previous architectural models by getting training accuracy of 99% and validation accuracy of 99%.*

Keywords: Convolutional Neural Network (CNN), Deep Learning, image processing, Ensemble, tomato leaf disease.

DAFTAR ISI

HALAMAN JUDUL	i
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM	
UNIVERSITAS HASANUDDIN	ii
HALAMAN PERNYATAAN KEOTENTIKAN	iii
LEMBAR PERSETUJUAN PEMBIMBING	iv
HALAMAN PENGESAHAN	v
KATA PENGANTAR.....	vi
PERNYATAAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN	
AKADEMIS.....	ix
ABSTRAK	x
ABSTRACT	xi
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
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.....	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Penelitian Terkait.....	4
2.2 Penyakit Tanaman Tomat.....	7
2.2.1. Tomato late blight	7
2.2.2. Tomato spider mates	8

2.2.3.	Tomato leaf mold	9
2.2.4.	Tomato target spot.....	9
2.2.5.	Tomato early blight	10
2.2.6.	Tomato bacterial spot.....	10
2.3	Citra Digital.....	11
2.4	Convolutional Neural Network	11
2.4.1.	Convolution.....	12
2.4.2.	Pooling	13
2.4.3.	Fungsi Aktivasi	14
2.4.4.	Flatten.....	15
2.4.5.	Fully Connected Layer	16
2.5	Arsitektur Convolutional Neural Network Transfer Learning	17
2.5.1.	InceptionV3.....	18
2.5.2.	Xception	19
2.5.3.	VGG16	20
2.6	Ensemble Learning.....	21
2.7	Evaluasi Kinerja Model.....	22
2.7.1.	Confusion Matrix	22
2.7.2.	Recall.....	23
2.7.3.	Precision.....	23
2.7.4.	Accuracy	23
2.7.5.	F1-Score	24
2.7.6.	Kurva AUC-ROC.....	24
BAB III METODE PENELITIAN		25
3.1	Waktu dan Lokasi Penelitian.....	25
3.2	Perangkat Penelitian	25

3.3	Sumber Data	25
3.4	Tahapan Penelitian	27
3.4.1.	Preprocessing	27
3.4.2.	Split Data.....	28
3.4.3.	Build Model Transfer Learning	28
3.4.4.	Implementasi Model Ensembled Transfer Learning.....	29
3.4.5.	Perbandingan Kinerja Model	29
BAB IV PEMBAHASAN.....		30
4.1	Deskripsi Data	30
4.2	Preprocessing.....	30
4.3	Split Data.....	32
4.4	Augmentasi Data	32
4.5	Arsitektur Transfer Learning <i>InceptionV3</i> , <i>Xception</i> , dan <i>VGG16</i>	33
4.6	Transfer Learning Menggunakan <i>InceptionV3</i> , <i>Xception</i> , dan <i>VGG16</i>	35
4.6.1.	<i>Xception</i>	35
4.6.2.	<i>InceptionV3</i>	37
4.6.3.	<i>VGG16</i>	39
4.7	Ensemble Stacking Model Transfer Learning CNN	42
4.8	Evaluasi Kinerja Model.....	45
4.9	Deployment Model Ensembled Transfer Learning	47
BAB V KESIMPULAN.....		50
5.1	Kesimpulan.....	50
5.2	Saran.....	50
DAFTAR PUSTAKA		51
LAMPIRAN.....		54

DAFTAR GAMBAR

Gambar 2. 1 Penyakit daun tomat late blight.....	8
Gambar 2. 2 Penyakit daun tomat spider mites.....	8
Gambar 2. 3 Penyakit daun tomat leaf mold.....	9
Gambar 2. 4 Penyakit daun tomat target spot	9
Gambar 2. 5 Penyakit daun tomat early blight.....	10
Gambar 2. 6 Penyakit daun tomat bacterial spot.....	10
Gambar 2. 7 Citra digital RGB	11
Gambar 2. 8 Arsitektur Convolutional Neural Network.....	12
Gambar 2. 9 Ilustrasi Convolution Layer.....	13
Gambar 2. 10 Ilustrasi Pooling Layer	14
Gambar 2. 11 Fungsi aktivasi ReLU	15
Gambar 2. 12 Fungsi aktivasi softmax.....	15
Gambar 2. 13 Ilustrasi Flatten layer	16
Gambar 2. 14 Ilustrasi Fully connected layer	16
Gambar 2. 15 Arsitektur InceptionV3.....	18
Gambar 2. 16 Stem blok.....	18
Gambar 2. 17 Modul A, B, dan C	19
Gambar 2. 18 Reduction blok	19
Gambar 2. 19 Arsitektur Xception	20
Gambar 2. 20 Entry flow, middle flow, dan exit flow pada Xception.....	20
Gambar 2. 21 Arsitektur VGG16	21
Gambar 2. 22 Ilustrasi ensemble stacking.....	22
Gambar 2. 23 Kurva AUC-ROC	24
Gambar 3. 1 Diagram alur penelitian.....	27
Gambar 3. 2 Ilustrasi augmentasi data	28
Gambar 4. 1 Data citra RGB	30
Gambar 4. 2 Hasil resize citra ukuran 299x299 pixel.....	31
Gambar 4. 3 Konversi data citra menjadi array	31
Gambar 4. 4 Proses sebelum normalisasi data citra dan setelah normalisasi	32
Gambar 4. 5 Ilustrasi proses hasil augmentasi.....	33

Gambar 4. 6 Ringkasan arsitektur InceptionV3	33
Gambar 4. 7 Ringkasan arsitektur Xception.....	34
Gambar 4. 8 Ringkasan arsitektir VGG16.....	34
Gambar 4. 9Kurva akurasi training dan validation model Xception.....	35
Gambar 4. 10 Confusion matrix model Xception.....	36
Gambar 4. 11 Kurva ROC model Xception	37
Gambar 4. 12 Kurva akurasi training dan validation model InceptionV3	37
Gambar 4. 13 Confusion matrix model InceptionV3.....	38
Gambar 4. 14 Kurva ROC model InceptionV3	39
Gambar 4. 15 Kurva akurasi training dan validation model VGG16.....	40
Gambar 4. 16 Confusion matrix model VGG16.....	40
Gambar 4. 17 Kurva ROC model VGG16	41
Gambar 4. 18 Arsitektur model ensemble stacking	42
Gambar 4. 19 Kurva akurasi training dan validation model ensemble stacking	43
Gambar 4. 20 Confusion matrix model ensemble stacking	43
Gambar 4. 21 Kurva ROC model ensembled stacking	44
Gambar 4. 22 Visualisasi layout aplikasi android	47
Gambar 4. 23 Hasil prediksi aplikasi android	48
Gambar 4. 24 Percobaan prediksi model data citra rotasi.....	49

DAFTAR TABEL

Tabel 2. 1 Ringkasan penelitian terdahulu	6
Tabel 2. 2 Confusion matrix.....	22
Tabel 3. 1 Dataset penyakit daun tomat	26
Tabel 4. 1 Hasil evaluasi kinerja model arsitektur Xception	36
Tabel 4. 2 Hasil evaluasi kinerja model arsitektur InceptionV3	38
Tabel 4. 3 Hasil evaluasi kinerja model arsitektur VGG16	41
Tabel 4. 4 Hasil evaluasi kinerja model ensemble stacking.....	44
Tabel 4. 5 Evaluasi presisi, recall, dan f1-score	45
Tabel 4. 6 Akurasi Training dan Akurasi Validation.....	46
Tabel 4. 7 AUC-ROC.....	46

BAB I

PENDAHULUAN

1.1 Latar Belakang

Tomat merupakan salah satu komoditas hortikultura yang sering menjadi sumber bahan makanan. Tomat sangat potensial dikembangkan dan bernilai jual tinggi. Tomat dapat dinikmati baik sebagai jus, bahan masakan, dan sebagainya. Salah satu manfaat tomat yaitu pada bidang kesehatan, tomat mampu menghambat pertumbuhan beberapa jenis kanker karena tomat mengandung antioksidan berupa likopen. Tingginya permintaan pasar terhadap buah tomat, baik tomat segar maupun tomat olahan membutuhkan produktivitas yang tinggi pada sektor pertanian tomat (Adhikari dkk., 2017).

Dari hasil proyeksi yang dilakukan oleh Pusat Data dan Informasi Pertanian Indonesia bahwa pada tahun 2021 produksi tomat akan mencapai 1.035.475 ton dan konsumsi tomat akan mencapai 1.053.249 ton. Dimana dalam hal ini jumlah konsumsi tomat lebih banyak daripada produksi tomat sehingga akan terjadi kekurangan stok produksi. Kenaikan harga tomat yang cukup signifikan akibat kekurangan stok produksi (Sita & Hadi, 2016). Hingga saat ini peningkatan produksi tomat masih terkendala oleh faktor penyakit tanaman (Astiningrum dkk., 2019). Penyakit yang menyerang tanaman tomat merupakan ancaman yang dapat menghambat produksi tomat global dan berefek pada kerugian pasca panen yang signifikan (Ali dkk., 2019). Sangat penting bagi petani untuk mendeteksi penyakit pada tanaman lebih awal untuk mengontrol penyebaran penyakit (Martinelli dkk., 2015). Penyakit ini dapat dikenali secara visual karena memiliki ciri warna dan tekstur yang unik. Tetapi pengenalan secara visual memiliki kekurangan yaitu sulit dalam mengenali kemiripan antara satu jenis penyakit dengan penyakit lain sehingga berdampak pada subjektivitas dan kurang akuratnya penyakit yang diidentifikasi (Rakhmawati dkk., 2018). Karena sulitnya mengidentifikasi penyakit daun tomat, maka dibutuhkan suatu metode yang dapat mengidentifikasi penyakit daun tomat secara akurat. Salah satu metode yang dapat digunakan adalah Deep Learning,

dengan membuat sebuah model untuk mengklasifikasi penyakit daun tomat menggunakan data citra.

Metode *Deep learning* merupakan salah satu cabang dari *machine learning*. Model *deep learning* dapat mempelajari komputasinya sendiri dengan menggunakan otaknya sendiri. *Deep learning* dirancang untuk terus menganalisis data seperti pada otak manusia dalam mengambil keputusan. Salah satu jenis *neural network* yang biasa digunakan pada data *image* adalah CNN. Karena dalamnya tingkat jaringan maka CNN termasuk dalam jenis *deep neural network* dan sering digunakan dalam data citra (Andi dkk., 2020). CNN diimplementasikan dengan menerima data citra sebagai input. Namun biasanya kinerja model yang dihasilkan masih kurang optimal. Untuk itu digunakan teknik *Transfer Learning* agar mengoptimalkan kinerja model. Metode *transfer learning* merupakan metode yang menggunakan bobot dan fitur yang sudah dilatih menggunakan dataset yang besar, sehingga kaya akan fitur dan dapat mempercepat waktu komputasi (Thiodorus dkk., 2021). *Transfer learning* menggunakan dataset besar yaitu ImageNet untuk melatih model dan dioptimalkan menggunakan dataset kecil. Selanjutnya untuk lebih mengoptimalkan kinerja model dapat menggunakan teknik *ensemble*. Metode *ensemble* merupakan gabungan dari beberapa model yang dijadikan satu sehingga kinerjanya lebih baik dibandingkan single model (Nurmasani & Pristyanto, 2021). Diharapkan dengan mengkombinasikan teknik transfer learning dan ensemble, model dapat menghasilkan kinerja yang lebih baik.

Maka dari itu, penulis tertarik untuk melakukan penelitian dengan mengkombinasikan beberapa model transfer learning CNN menggunakan teknik ensemble. Sehingga peneliti memutuskan untuk membuat penelitian yang berjudul “Evaluasi Kinerja Model Transfer Learning Menggunakan Ensemble Pada Klasifikasi Penyakit Daun Tomat”.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini adalah:

1. Bagaimana mengimplementasi beberapa model *transfer learning CNN* dan model *ensembled stacking transfer learning* ?
2. Bagaimana perbandingan kinerja model antara *ensembled stacking transfer learning CNN* dengan model *arsitektur transfer learning CNN* ?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini adalah:

1. Dataset yang digunakan adalah Dataset *Tomato Leaf Disease*.
2. Arsitektur *transfer learning* yang digunakan adalah *InceptionV3*, *Xception*, dan *VGG16*.
3. Penyakit daun tomat yang diteliti yaitu *Tomato healthy*, *Tomato late blight*, *Tomato spider mites*, *Tomato leaf mold*, *Tomato target spot*, *Tomato early blight*, *Tomato bacterial spot*.
4. Metode *ensemble* yang digunakan adalah *Stacking*.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Mengimplementasi beberapa model *transfer learning CNN* dan model *ensembled stacking transfer learning*.
2. Menganalisis perbandingan kinerja model yang dihasilkan dari model *ensembled transfer learning CNN* dengan model *arsitektur transfer learning CNN*.

1.5 Manfaat Penelitian

Penelitian ini dapat menghasilkan model untuk mengklasifikasi penyakit daun tomat, yang kemudian dapat dideploy pada suatu aplikasi atau alat. Selain itu, penelitian ini dapat menjadi informasi mengenai implementasi metode ensemble pada beberapa model arsitektur transfer learning CNN untuk meningkatkan performa model.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian ini merujuk pada beberapa penelitian yang telah dilakukan sebelumnya. Berikut ini merupakan penelitian terkait:

Penelitian yang dilakukan oleh Felix pada tahun 2019 dengan judul *“Implementasi CNN dan SVM untuk Identifikasi Penyakit Tomat via Daun”* bertujuan membangun sebuah sistem yang dapat mengklasifikasi suatu jenis penyakit daun tomat. Dalam penelitian yang dilakukan, citra daun tomat melewati proses transformasi warna RGB (Red Green Blue) ke HSV (Hue Saturation Value) dan ke Grayscale. Setelah transformasi warna selanjutnya dilakukan ekstraksi fitur menggunakan GLCM, kemudian dilakukan pelatihan model menggunakan dua algoritma yaitu SVM dan CNN. Dengan 160 data latih dan 140 data uji. Didapatkan hasil akurasi pada model SVM sebesar 95% dan pada model CNN sebesar 97.5% (Felix dkk., 2019).

Andi Asrafil melakukan penelitian pada tahun 2020 dengan judul *“Klasifikasi Penyakit Tanaman Apel Dari Citra Daun Dengan Convolutional Neural Network”*. Yang bertujuan untuk memberikan solusi terbaik pada petani dalam menentukan jenis hama/penyakit pada tanaman apel. Peneliti merancang sebuah aplikasi android dengan memanfaatkan algoritma CNN untuk memprediksi jenis penyakit tanaman apel. Dataset yang digunakan berasal dari *Kaggle* yang terdiri dari tiga jenis penyakit yaitu *scab*, *blackrot*, dan *rust*. Sebelum melatih model, peneliti melakukan pra proses dengan *wrapping* dan *cropping*. Model yang dibangun menggunakan algoritma CNN memperoleh akurasi sebesar 97.1%. Terakhir peneliti melakukan implementasi model pada aplikasi android yang sudah dirancang (Paliwang dkk., 2020).

Penelitian yang dilakukan oleh Hardiono adalah *“Mempelajari Dataset Penyakit Daun Tomat menggunakan Arsitektur InceptionV3”*. Hardiono melakukan pelatihan model menggunakan arsitektur *InceptionV3*. Dataset yang digunakan merupakan dataset penyakit daun tomat yang terdiri dari 6 kelas. Peneliti melakukan

training model sebanyak 50 epoch, hasilnya akurasi training yang didapatkan sebesar 95% sedangkan akurasi testing sebesar 90%. (Hardiono dkk., 2020).

Penelitian dengan menggunakan arsitektur CNN pernah dilakukan oleh Dwi dan Daniel pada tahun 2020 dengan judul “*Implementasi Convolutional Neural Network Untuk Identifikasi Penyakit Daun Gambas*”. Pada penelitian ini, peneliti membangun sebuah model menggunakan salah satu arsitektur CNN yaitu *MobileNet*. Data yang digunakan terdiri dari tiga jenis penyakit yaitu *Embun bulu*, *Kumbang daun*, dan *Ulat daun*. Sebelum melakukan pelatihan model, peneliti melakukan proses *resize* gambar menjadi ukuran 224x224. Selanjutnya dilakukan training model sebanyak 50 epoch. Pada epoch ke-25 model memperoleh akurasi terbaik yaitu sebesar 92%. (Sari & Swanjaya, 2020).

Penelitian yang dilakukan Daniel dan Nilton pada tahun 2018 menggunakan salah satu arsitektur CNN dan juga memanfaatkan metode *transfer learning*. Judul dari penelitian ini adalah “*Klasifikasi Lesi Kulit Menggunakan Jaringan Saraf Konvolusi dalam Gambar Klinis*”. Dataset yang digunakan terdiri dari dua belas jenis skin lesi dari dua dataset berbeda yaitu *Atlas Dataset* dan *Edinburgh Dataset*. Peneliti melakukan augmentasi pada data citra yang dimiliki untuk mendapatkan gambar yang memiliki cahaya yang berbeda. Selanjutnya model dibuat dengan menggunakan arsitektur *ResNet-152*. Peneliti juga mengimplementasi *transfer learning* dengan menggunakan bobot yang telah dilatih menggunakan *ImageNet*. Hasil AUC yang didapatkan tiap kelas diatas 0.90. (Mendes & Silva, 2018).

Penelitian dengan menggunakan metode ensemble pernah dilakukan oleh Yoga pada tahun 2022 dengan judul “*Klasifikasi Penyakit Diabetes Pada Imbalanced Class Dataset Menggunakan Algoritme Stacking*”. Penelitian ini bertujuan untuk meningkatkan kinerja model *single classifier* pada dataset yang *imbalanced*. Algoritma yang digunakan pada *single classifier* yaitu *C4.5*, *K-NN*, *SVM*, dan *Random Forest*. Dengan menggunakan metode *ensemble stacking*, peneliti mendapatkan hasil akurasi lebih baik dibandingkan *single classifier*. Pada dataset pertama model *stacking* mendapatkan akurasi sebesar 89% dan pada dataset kedua mendapatkan akurasi sebesar 96% (Pristyanto dkk., 2022).

Tabel 2. 1 Ringkasan penelitian terdahulu

No	Judul	Deskripsi Masalah	Data	Akurasi	Sumber Paper
1	Implementasi CNN dan SVM untuk Identifikasi Penyakit Tomat via Daun	Tanaman tomat yang rentan terhadap penyakit dan sulit membedakannya	4 kelas dengan total 200 data citra	CNN 97.5% dan SVM 95%	https://mikroskil.ac.id/ejournal/index.php/jsm/article/view/670/330
2	Klasifikasi Penyakit Tanaman Apel Dari Citra Daun Dengan Convolutional Neural Network	Pencegahan penyakit pada tanaman apel dapat dilakukan dengan penyemprotan, namun perlu bantuan pakar untuk menentukan solusi terbaik.	4 kelas dengan total 9643 data citra	CNN 97.1%	https://jurnal.wicida.ac.id/index.php/sebatik/article/view/1060/297
3	Mempelajari Dataset Penyakit Daun Tomat menggunakan Arsitektur InceptionV3	Untuk memaksimalkan daya tahan dan produktifitas tanaman tomat	6 kelas dengan total 600 data citra	90%	https://www.academia.edu/42986647/Learning_Tomato-Leaf_Dataset_using_Inception-V3_Architecture_with_Python
4	Implementasi Convolutional Neural Network Untuk Identifikasi Penyakit Daun Gambas	Penyakit dan hama yang menyerang tanamn gambus dapat menyebabkan gagal panen. Selain itu penyakit tanaman gambus sangat sulit diidentifikasi.	4 kelas	90%	https://proceeding.unpkediri.ac.id/index.php/inotek/article/view/76/50

5	Klasifikasi Lesi Kulit Menggunakan Jaringan Saraf Konvolusi dalam Gambar Klinis	Penyakit kanker kulit yang menyebabkan lebih dari 8.2 juta kematian. Serta sulitnya dalam mendeteksi kanker kulit.	12 kelas dengan total 5116 data citra	-	https://arxiv.org/pdf/1812.02316.pdf
6	Klasifikasi Penyakit Diabetes Pada Imbalanced Class Dataset Menggunakan Algoritme Stacking	Penyakit diabetes yang memiliki potensi kematian. Dan deteksi dini diabetes yang dapat mencegah terjadinya kematian	1288 data citra	Ensemble Stacking 96%	https://www.ejournal.stmik-budidarma.ac.id/index.php/mib/article/view/3442/2411

2.2 Penyakit Tanaman Tomat

Penyakit tanaman tomat biasanya disebabkan oleh beberapa faktor, diantaranya bakteri, virus, dan juga serangga. Penyakit tersebut cukup mempengaruhi kualitas dan kuantitas produksi tomat. Fokus penelitian ini adalah penyakit sebagai berikut. *Tomato late blight*, *Tomato spider mites*, *Tomato leaf mold*, *Tomato target spot*, *Tomato healthy*, *Tomato early blight*, *Tomato bacterial spot*.

2.2.1. Tomato late blight

Penyakit busuk daun tomat disebabkan oleh jamur *Phytophthora Infestans* de Bary. Biasanya penyakit terjadi selama musim dingin. Lesi cekung berwarna hijau tua atau coklat yang basah oleh air berkembang di daun tomat. Cincin jamur keputihan yang mengelilingi area yang terkena sering tumbuh di permukaan bawah daun, dengan cepat menginfeksi daun tanaman tomat yang lain (Kankolongo, 2018).



Gambar 2. 1 Penyakit daun tomat late blight

(Sumber gambar : Dataset Tomato Leaf)

2.2.2. **Tomato spider mates**

Tomato spider mates merupakan salah satu yang sering menimbulkan kerusakan parah sehingga menyebabkan kematian tanaman inangnya. Gejala serangan TSM dapat berupa kerusakan langsung terhadap tanaman seperti gugur daun prematur, daun terbakar dan bahkan dapat menyebabkan tanaman mati akibat serangan yang berat, tetapi juga kerusakan secara tidak langsung terhadap tanaman seperti penurunan dalam proses fotosintesis dan transpirasi akibat rusaknya jaringan pada daun (Satria, 2011).



Gambar 2. 2 Penyakit daun tomat spider mites

(Sumber gambar : Dataset Tomato Leaf)

2.2.3. Tomato leaf mold

Bercak daun disebabkan oleh jamur *Cladosporium fulvum*, merupakan salah satu penyakit daun tomat yang paling merusak tanaman dan tumbuh dalam kondisi lingkungan yang lembab (Maulana, 2020).



Gambar 2. 3 Penyakit daun tomat leaf mold

(Sumber gambar : Dataset Tomato Leaf)

2.2.4. Tomato target spot

Target spot banyak tersebar di negara beriklim tropis. Infeksi dari penyakit ini ditandai dengan bintik-bintik berbentuk tidak beraturan dengan garis tepi berwarna kuning. Bintik-bintik membesar hingga menyebar ke seluruh permukaan daun. (Alim, 2020).



Gambar 2. 4 Penyakit daun tomat target spot

(Sumber gambar : Dataset Tomato Leaf)

2.2.5. Tomato early blight

Tomato early blight atau penyakit pembusukan daun dini, biasanya penyakit ini disebabkan oleh jamur *Solani*. Penyakit ini ditandai dengan munculnya bintik-bintik coklat hingga hitam pada daun. Daun yang terinfeksi menunjukkan bercak coklat hingga hitam (lesi) hingga diameter 1-2 cm dengan tepi gelap dan memiliki pola cincin konsentris (Sanoubar & Barbanti, 2017).



Gambar 2. 5 Penyakit daun tomat early blight

(Sumber gambar : Dataset Tomato Leaf)

2.2.6. Tomato bacterial spot

Tomato bacterial spot disebabkan oleh bakteri *Xanthomonas* yang tersebar luas dan merupakan penyakit bakteri yang merusak pada tanaman tomat (Osdaghi dkk., 2017).



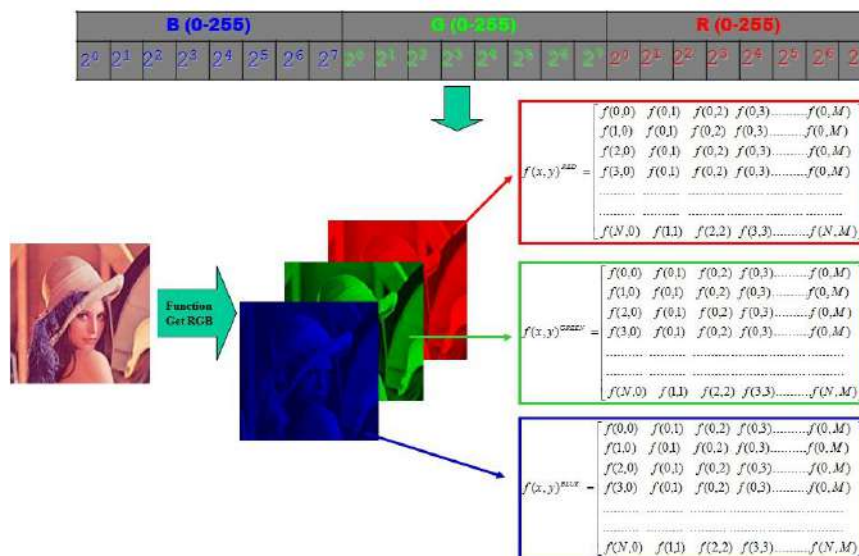
Gambar 2. 6 Penyakit daun tomat bacterial spot

(Sumber gambar : Dataset Tomato Leaf)

2.3 Citra Digital

Citra digital merupakan representasi numerik dari data citra agar dapat diolah. Citra digital direpresentasikan dalam bentuk matriks $f(x,y)$ yang dimana x,y itu merepresentasikan suatu pixel. Citra digital terdiri dari M kolom dan N baris. Sebelum melakukan proses *Convolutional Neural Network* data citra harus direpresentasikan terlebih dahulu dalam bentuk numerik agar dapat diolah. Untuk nilai numerik pada tiap pixel biasanya dalam rentang $[0, 255]$.

Terdapat beberapa macam citra, salah satunya citra *RGB*. Citra *RGB* atau *Red*, *Green*, dan *Blue* merupakan sebuah citra berwarna yang memiliki masing-masing pixel tertentu. Jika tiap warna memiliki rentang nilai $[0, 255]$ maka terdapat tiga matriks dengan masing-masing rentang nilai yang sama sehingga totalnya adalah 255^3 . Seperti yang ditunjukkan pada gambar 2.7.



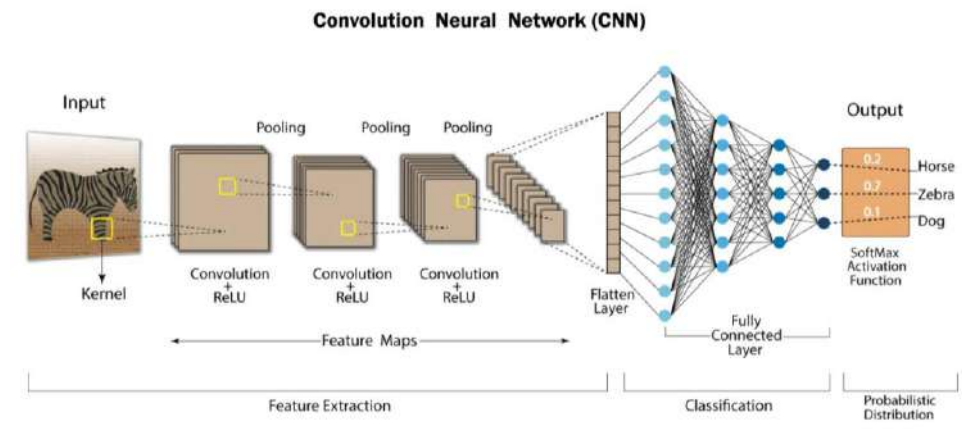
Gambar 2. 7 Citra digital RGB

(Sumber gambar : pemrogramanmatlab.com)

2.4 Convolutional Neural Network

Convolutional Neural Network (CNN) merupakan pengembangan dari Multi-Layer Perceptron (MLP) yang dirancang untuk mengolah data dalam bentuk grid, salah satunya adalah citra dua dimensi. CNN digunakan untuk mengklasifikasikan data yang terlabel dengan menggunakan metode Supervised Learning, yang mana di

dalam Supervised Learning terdapat data yang dilatih dan variable yang ditargetkan sehingga tujuan dari metode ini adalah mengelompokkan suatu data ke data yang sudah ada. CNN telah menunjukkan kinerja yang baik dalam klasifikasi data gambar (Bahuleyan, 2018).



Gambar 2. 8 Arsitektur Convolutional Neural Network

(Sumber gambar : developersbreach.com)

Pada gambar 2.8 terlihat secara garis besar terdapat dua proses pada CNN. Proses pertama adalah *Feature Extraction* yang meliputi *convolution* dan *pooling*. Kemudian proses kedua yaitu *Classification* yang meliputi *flatten*, dan *fully connected layer*. Berikut penjelasan dari proses pada gambar 2.8

2.4.1. Convolution

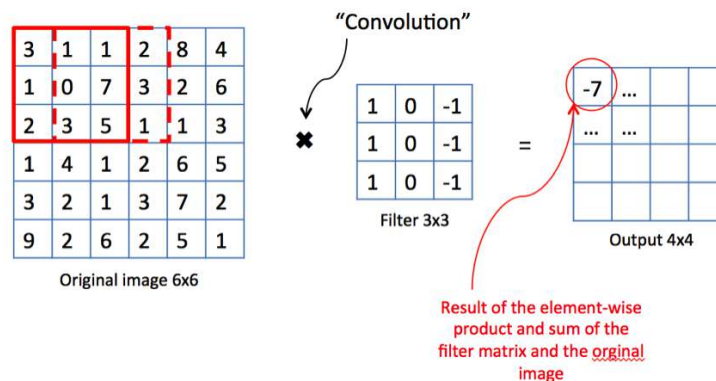
Convolution Layer menggunakan *filter* untuk mengekstraksi objek dari citra input. *Filter* ini berisi bobot yang digunakan untuk mendeteksi karakter dari objek seperti tepi, kurva, atau warna. Konvolusi akan menghasilkan transformasi linear dari citra input yang sesuai dengan informasi spasial pada data. *Filter* diaplikasikan secara berulang sehingga menghasilkan serangkaian bidang reseptif. Terdapat parameter yang dapat diubah untuk memodifikasi sifat tiap lapisan, yaitu ukuran *filter*, *stride* dan *padding*. *Stride* mengontrol bagaimana *filter* diterapkan pada data input dengan bergerak sepanjang ukuran piksel yang telah ditentukan. *Padding* adalah penambahan ukuran piksel dengan nilai tertentu disekitar data input agar hasil dari bidang reseptif tidak terlalu kecil sehingga tidak banyak informasi yang hilang. Nilai ini biasanya nol sehingga disebut dengan *zero padding*. Langkah ini melibatkan

menggeser filter matriks (katakanlah ukuran 3x3) di atas gambar input yang berukuran lebar gambar x tinggi gambar. Filter pertama ditempatkan pada matriks gambar dan kemudian kami menghitung perkalian elemen antara filter dan bagian gambar yang tumpang tindih, diikuti dengan penjumlahan untuk memberikan nilai fitur (Bahuleyan, 2018).

$$S(i, j) = (I \times K)(i, j) = \sum_a \sum_b I(a, b) K(i - a, j - b) \quad (2.1)$$

Keterangan:

$S(i, j)$	=	Fungsi hasil konvolusi
I	=	Input
K	=	Filter
(i, j)	=	Pixel input
(a, b)	=	Pixel kernel



Gambar 2. 9 Ilustrasi Convolution Layer

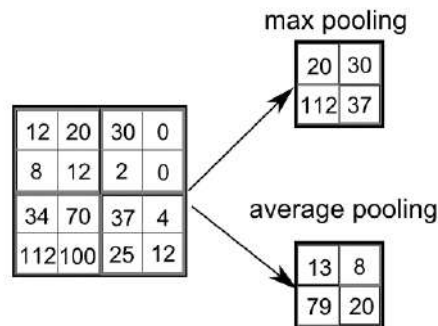
(Sumber gambar : kaggle.com)

Seperti pada gambar 2.9 nantinya citra input akan terbentuk sebuah matriks 3x3 yang bergeser sesuai dengan *stride* yang sudah ditentukan. Kemudian matriks yang terbentuk akan dilakukan kuantisasi pada filter. Hasil dari kuantisasi matriks berupa satu nilai output. Nantinya output kuantisasi dari tiap matriks pada citra akan membentuk sebuah matriks yang merupakan hasil dari konvolusi.

2.4.2. Pooling

Pooling adalah cara untuk mengurangi dimensi peta fitur yang diperoleh dari langkah konvolusi, yang secara formal dikenal sebagai proses down sampling.

Terdapat dua cara yang sering digunakan yaitu *max pooling* dan *average pooling*. Misalnya, dengan *max pooling* dengan ukuran jendela 2x2, kami hanya mempertahankan elemen dengan nilai maksimum di antara 4 elemen peta fitur yang tercakup dalam jendela ini (Bahuleyan, 2018). Sedangkan *average pooling* akan mempertahankan hasil rata-rata dari elemen fitur yang tercakup dalam jendela.



Gambar 2. 10 Ilustrasi Pooling Layer

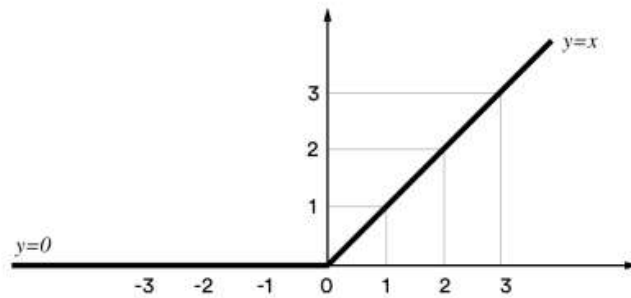
(Sumber gambar : medium.com)

Terdapat dua operasi *pooling* seperti pada gambar 2.10. Pada proses pooling nantinya hasil dari proses konvolusi akan terbentuk sebuah matriks 2x2. Nantinya tiap matriks tersebut akan diubah menjadi satu elemen matriks saja.

2.4.3. Fungsi Aktivasi

Operasi konvolusi adalah linier dan untuk membuat jaringan saraf lebih kuat, perlu memperkenalkan beberapa non-linier. Untuk tujuan ini, dapat menerapkan fungsi aktivasi seperti ReLU (Bahuleyan, 2018). Rectified linear unit (ReLU) merupakan salah satu fungsi aktivasi yang sering digunakan pada convolutional neural network (Chen dkk., 2018). Fungsi aktivasi ReLU mempercepat waktu komputasi karena sangat sederhana. Jika nilai input negatif, maka output nya adalah 0. Jika positif maka outputnya adalah nilainya sendiri.

$$\text{ReLU}(x) = \max(0, x) \quad (2.2)$$

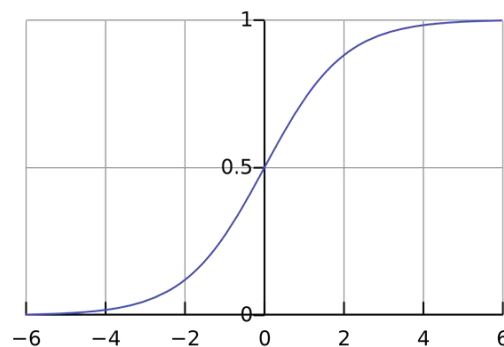


Gambar 2. 11 Fungsi aktivasi ReLU

(Sumber gambar : iq.opengenus.org)

Fungsi aktivasi ReLU digunakan pada proses ekstraksi fitur, tetapi pada proses klasifikasi fungsi aktivasi yang digunakan adalah *softmax*. Fungsi aktivasi *softmax* berada pada layer output yang bertujuan untuk klasifikasi. Biasanya fungsi aktivasi *softmax* digunakan pada multikelas.

$$\text{softmax}(y_i) = \frac{y_i}{\sum_{j=1}^n y_j} \quad (2.3)$$



Gambar 2. 12 Fungsi aktivasi softmax

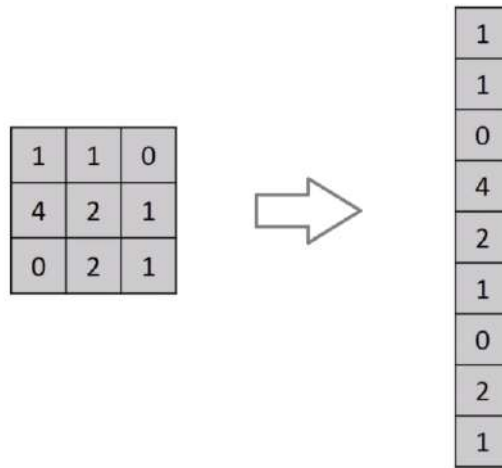
(Sumber gambar : medium.com)

Seperti pada gambar 2.11 dan 2.12 perbedaan fungsi aktivasi softmax dan relu yaitu pada output yang dihasilkan. Pada fungsi aktivasi relu output yang dihasilkan adalah 0 sampai ∞ . Sedangkan fungsi aktivasi softmax output yang dihasilkan adalah 0 sampai 1.

2.4.4. Flatten

Flatten layer merupakan sebuah metode untuk mengubah hasil dari *pooling layer*. *Flatten* mengubah multi-dimensi menjadi satu dimensi.. Proses *flatten* dilakukan agar output dari *feature extraction* dapat diproses oleh *fully connected*

layer. Output dari flatten akan menjadi input pada *fully connected layer*. Pada gambar 2.13 matriks ukuran 3x3 diubah menjadi sebuah vektor ukuran 9.

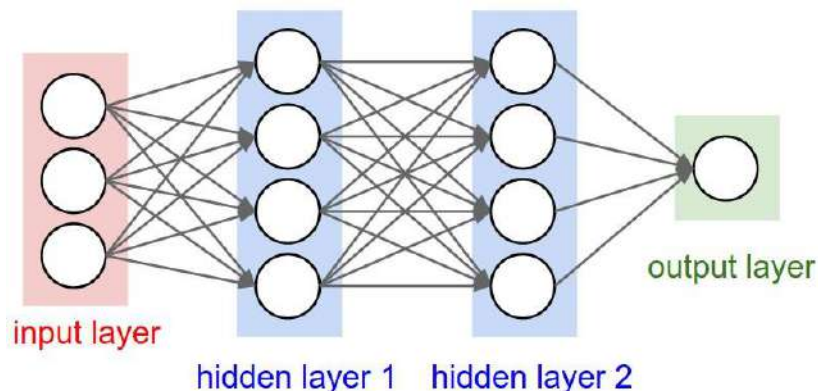


Gambar 2. 13 Ilustrasi Flatten layer

(Sumber gambar : superdatascience.com)

2.4.5. Fully Connected Layer

Fully connected layer biasanya berada pada layer paling akhir. Layer paling akhir digunakan untuk menentukan kelas, lapisan ini mendapatkan input dari proses sebelumnya untuk menentukan fitur mana yang paling berkorelasi dengan kelas tertentu. Fungsi dari lapisan ini adalah untuk menyatukan semua node menjadi satu dimensi (Albelwi & Mahmood, 2017). *Fully connected layer* dikenal sebagai *Multi layer perceptron* (MLP) yang digunakan untuk mengolah data sehingga dapat diklasifikasi.



Gambar 2. 14 Ilustrasi Fully connected layer

(Sumber gambar : cs231n.com)

Pada gambar 2.14 *fully connected layer* terdapat tiga bagian. Pertama *input layer* yang merupakan output dari *flatten layer*. Lalu ada bagian kedua yang merupakan layer-layer seperti *Dense layer*, *Dropout layer*, dan lainnya. Terakhir ada output layer yang merupakan layer untuk menentukan sebuah kelas.

2.5 Arsitektur Convolutional Neural Network Transfer Learning

Terdapat beberapa arsitektur pada Convolutional Neural Network (CNN) untuk pengklasifikasian citra. Pada tahun 2012, arsitektur CNN AlexNet yang berhasil menjadi pemenang ImageNet Competition yaitu kompetisi untuk klasifikasi dan deteksi citra yang terdiri dari jutaan citra dengan puluhan ribu kelas (Setiawan, 2019). Dalam penelitian ini arsitektur yang digunakan adalah *InceptionV3*, *Xception*, dan *VGG16*.

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 dkk., 2018). *Transfer learning* melatih model dengan memanfaatkan *ImageNet* sebagai pengetahuan umum untuk model. *ImageNet* merupakan sebuah dataset umum berjumlah besar dari Google.

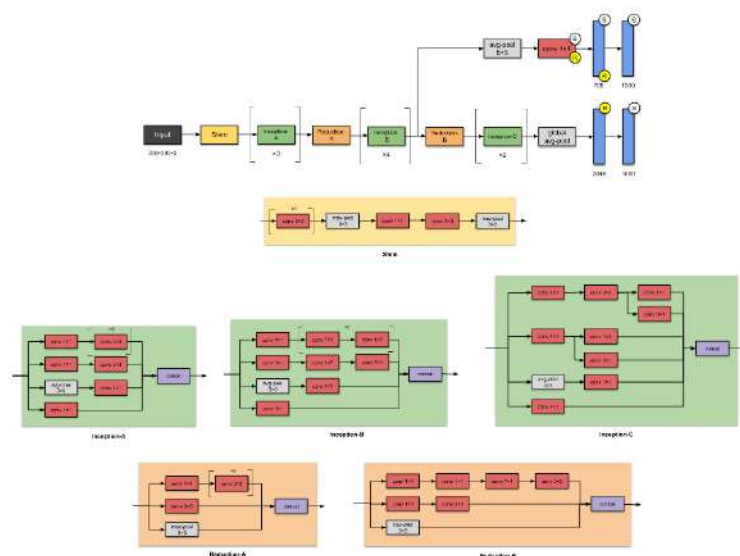
Dalam *transfer learning* terdapat beberapa istilah seperti *pre-trained* dan *fine tuning*. *Transfer learning* nantinya akan melatih model dengan menggunakan data besar, inilah yang disebut *pre-trained*. Pada *tensorflow*, *pre-trained* model dilakukan dengan menggunakan dataset *ImageNet*. Selain *pre-trained*, *fine-tuning* juga merupakan sebuah cara yang biasanya dipakai dalam melatih model. Model yang sudah dilakukan *pre-trained* akan dilatih lagi menggunakan data baru untuk kebutuhan lain.

Pada proses *training* model ada sebuah istilah yang disebut *epoch*. *Epoch* merupakan sebuah proses yang berulang dimana dataset akan dipelajari oleh model. Setelah model mempelajari dataset tersebut, model akan mengoptimalkan *weight* (bobot) yang dimiliki model menggunakan algoritma *Gradient Descent*. Setelah *weight* (bobot) sudah diupdate, dataset akan dipelajari lagi oleh model. Proses ini akan terus berulang sesuai dengan epoch yang ditentukan. Proses training model ini

sama dengan perlakuan statistik yaitu rancangan berulang. Rancangan berulang merupakan suatu percobaan dimana masing-masing unit percobaan diberikan sebuah perlakuan tetapi pengukurannya dilakukan berulang dalam beberapa kali.

2.5.1. InceptionV3

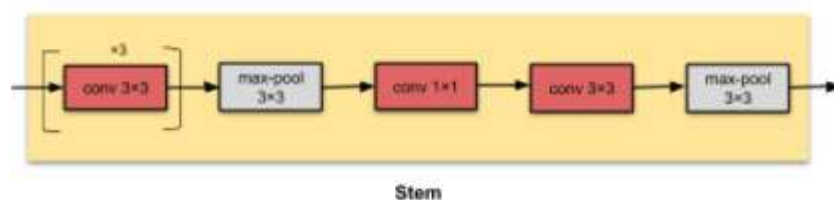
InceptionV3 merupakan pengembangan dari *InceptionV2*. *InceptionV3* dibuat pada tahun 2015 dengan jumlah 42 layer. *InceptionV3* memiliki tingkat error yang lebih rendah dibandingkan *InceptionV2*. Arsitektur *InceptionV3* menerima input dengan ukuran 299x299. Arsitektur ini melakukan beberapa perbaikan seperti penggunaan label smoothing, konvolusi 7x7, dan classifier tambahan.



Gambar 2. 15 Arsitektur InceptionV3

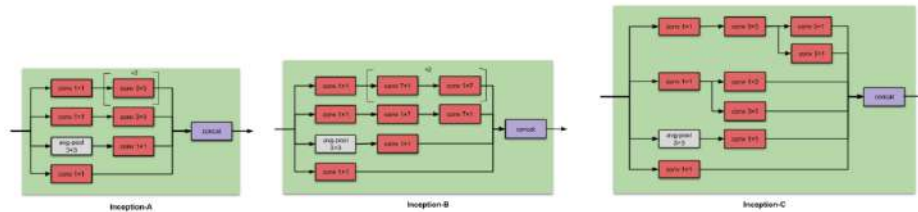
(Sumber gambar : towardsdatascience.com)

Pada gambar 2.15 arsitektur *InceptionV3* terbagi dalam beberapa blok, ada *stem* blok, *inception* blok, dan *reduction* blok. *Stem* blok merupakan set awal untuk memperkenalkan blok *inception*. Berikut *stem* blok pada *InceptionV3*.



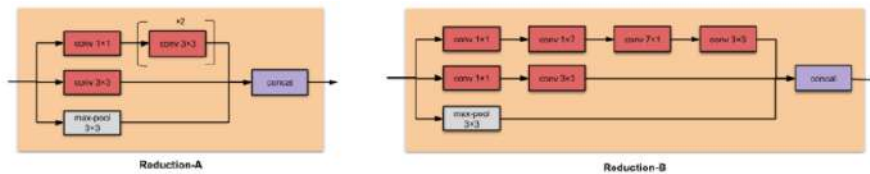
Gambar 2. 16 Stem blok

Selain proses *stem* blok, selanjutnya ada *inception* blok. Pada *InceptionV3* terdapat tiga modul utama *inception* blok yaitu modul A, modul B, dan modul C. Berikut modul A, B, dan C pada *InceptionV3*.



Gambar 2. 17 Modul A, B, dan C

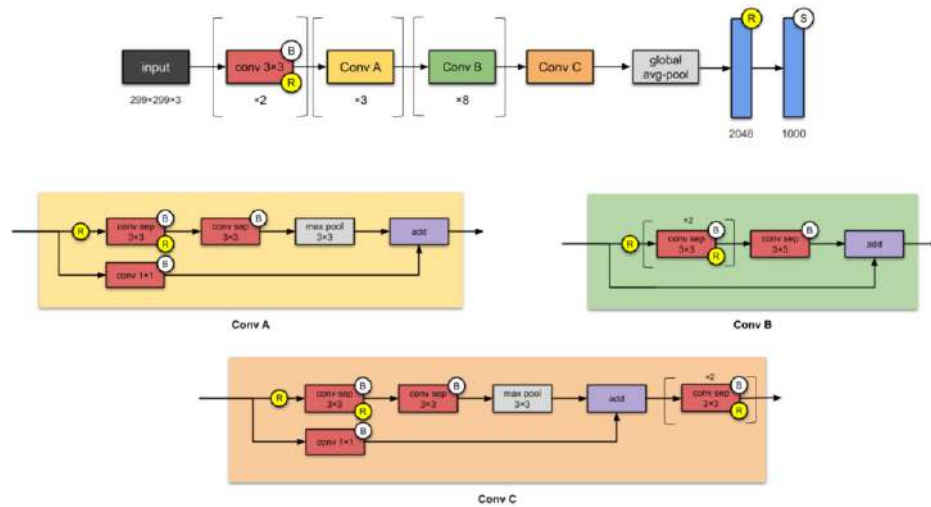
Kemudian ada *reduction* blok yang dirancang untuk mengubah dimensi input. Terdapat dua *reduction* blok yaitu *reduction A* dan *reduction B*. Berikut *reduction* blok pada *InceptionV3*.



Gambar 2. 18 Reduction blok

2.5.2. Xception

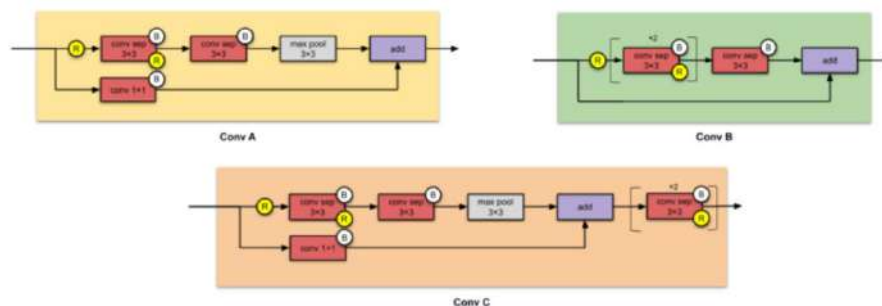
Xception merupakan arsitektur yang melibatkan *Depthwise Separable Convolutions*. *Xception* sebagai interpretasi modul *inception* yang bertujuan sebagai langkah perantara di antara konvolusi reguler dan operasi konvolusi. Arsitektur *Xception* memiliki 71 layer, yang menerima input dengan ukuran 299x299. *Xception* memiliki dua tingkat yang berbeda, masing-masing dengan tiga lapisan konvolusi. Berbeda dengan model *inception*, model *Xception* memiliki dua level, dimana salah satunya memiliki satu layer. Lapisan ini mengiris output menjadi tiga segmen dan meneruskannya ke kumpulan filter berikutnya (Srinivasan dkk., 2021).



Gambar 2. 19 Arsitektur Xception

(Sumber gambar : towardsdatascience.com)

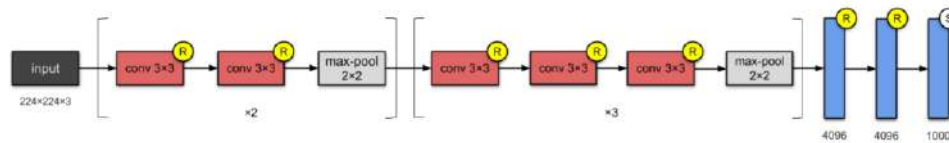
Arsitektur *Xception* terbagi menjadi tiga bagian yaitu *entry flow*, *middle flow*, dan *exit flow*. Pada *Xception* terdapat sebuah layer konvolusi *SeparableConv* yang dapat dimodifikasi secara mendalam atau *Depthwise Separable Convolutions*. Berikut tiga bagian arsitektur *Xception*.



Gambar 2. 20 Entry flow, middle flow, dan exit flow pada Xception

2.5.3. VGG16

Arsitektur *VGG16* merupakan versi pengembangan dari *AlexNet*. Ketika model diuji menggunakan dataset *ImageNet*, menunjukkan akurasi tes sebesar 92.7% (Srinivasan dkk., 2021). *VGG16* menggunakan 16 layer yang terdiri dari 13 layer konvolusi dan 3 layer *fully connected* dan menerima input dengan ukuran 224x224.



Gambar 2. 21 Arsitektur VGG16

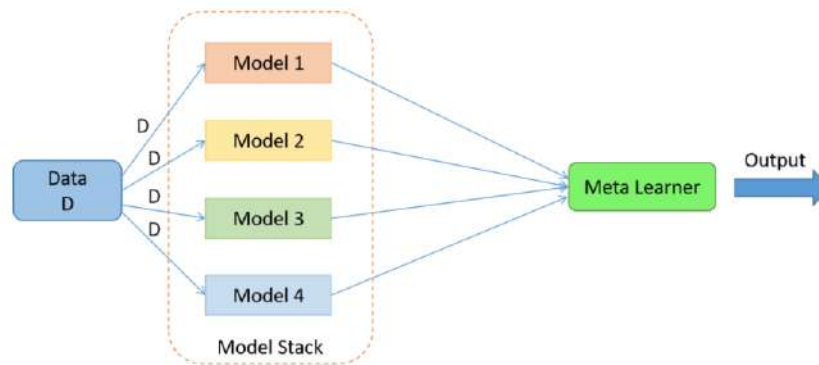
(Sumber gambar : towardsdatascience.com)

Pada gambar 2.21 terdapat layer konvolusi dengan filter 3x3 yang diikuti dengan *max pooling* 2x2. Jika prosesnya dijabarkan terdapat lima kali proses konvolusi. Proses pertama dan kedua konvolusi dilakukan masing-masing sebanyak dua kali dan terakhir diikuti *maxpooling*. Kemudian proses ketiga, keempat, dan kelima proses konvolusi dilakukan masing-masing tiga kali yang diikuti *max pooling*.

2.6 Ensemble Learning

Ensemble learning merupakan metode yang mencari model terbaik dengan cara menggabungkan beberapa model. Model dilatih dengan sendiri-sendiri kemudian mengkombinasikan model tersebut. Teknik ini lebih tahan terhadap noise dibandingkan dengan penggunaan classifier tunggal. Metode ini menggunakan pendekatan “divide and conquer” di mana masalah yang rumit diuraikan menjadi beberapa sub-masalah (Sidik, 2019).

Salah satu metode *ensemble* yaitu *stacking*. Teknik *stacking* bekerja dengan cara menggabungkan beberapa model yang disebut *weak-learner*, yang kemudian digabungkan menggunakan model yang disebut *meta-learner*. *Stacking* terdiri dari dua level yaitu base learner sebagai level-0 dan stacking model learner sebagai level-1 atau meta learner. Base learner (level-0) menggunakan model yang berbeda untuk belajar dari suatu dataset. Output dari masing-masing model dikumpulkan untuk membuat dataset baru. Dalam dataset baru, setiap instance berhubungan dengan nilai sesungguhnya yang seharusnya diprediksi. Kemudian dataset tersebut digunakan oleh stacking model learner (level-1) untuk memberikan hasil akhir (Sidik, 2019).



Gambar 2. 22 Ilustrasi ensemble stacking

(Sumber gambar : analyticsvidhya.com)

Pada gambar 2.22 output dari pelatihan model 1 sampai model 4 akan digabungkan. Setelah itu model meta learner yang dibuat dengan *Multi layer perceptron* (MLP) akan dilatih menggunakan gabungan dari output tersebut.

2.7 Evaluasi Kinerja Model

Evaluasi kinerja model merupakan yang terpenting dalam menentukan suatu model bagus atau tidak. Pada kasus klasifikasi, evaluasi kinerja yang digunakan berupa *Confusion matrix*, *precision*, *recall*, *accuracy* dan *F1-score*. Terdapat beberapa pengukuran kinerja model yang digunakan dalam penelitian ini.

2.7.1. Confusion Matrix

Confusion matrix merupakan pengukuran performa yang digunakan pada masalah klasifikasi, *confusion matrix* dapat berupa keluaran visualisasi yang menunjukkan hasil prediksi dari model yang benar maupun salah.

Tabel 2. 2 Confusion matrix

	Aktual			
		Kelas 1	Kelas 2	Kelas 3
	Kelas 1	T ₁₁	F ₁₂	F ₁₃
	Kelas 2	F ₂₁	T ₂₂	F ₂₃
	Kelas 3	F ₃₁	F ₃₁	T ₃₃

2.7.2. Recall

Recall merupakan rasio prediksi benar positif kelas i dibandingkan dengan keseluruhan data yang benar positif kelas i . Berikut rumus recall yang digunakan.

$$\text{Recall}_i = \frac{T_i}{T_i + \sum_{j=1}^c F_{ij}} \quad (2.4)$$

Keterangan:

$i = 1, 2, 3, \dots, 7$

$j = 1, 2, 3, \dots, 7$

c = Banyaknya kelas

T = Jumlah data yang diklasifikasi *True*

F = Jumlah data yang diklasifikasi *False*

2.7.3. Precision

Precision merupakan rasio prediksi benar positif kelas i dibandingkan dengan keseluruhan hasil yang diprediksi positif kelas i .

$$\text{Precision}_i = \frac{T_i}{T_i + \sum_{i=1}^c F_{ij}} \quad (2.5)$$

Keterangan:

$i = 1, 2, 3, \dots, 7$

$j = 1, 2, 3, \dots, 7$

c = Banyak kelas

T = Jumlah data yang diklasifikasi *True*

F = Jumlah data yang diklasifikasi *False*

2.7.4. Accuracy

Accuracy Merupakan rasio prediksi Benar (positif dan negatif) dengan keseluruhan data. Biasanya akurasi suatu model ditentukan dalam bentuk persen.

$$\text{Accuracy} = \frac{\sum_{i=1}^c T_i}{N} \quad (2.6)$$

Keterangan:

$i = 1, 2, 3, \dots, 7$

c = Banyak kelas

T = Jumlah data yang diklasifikasi *True*

N = Jumlah keseluruhan data

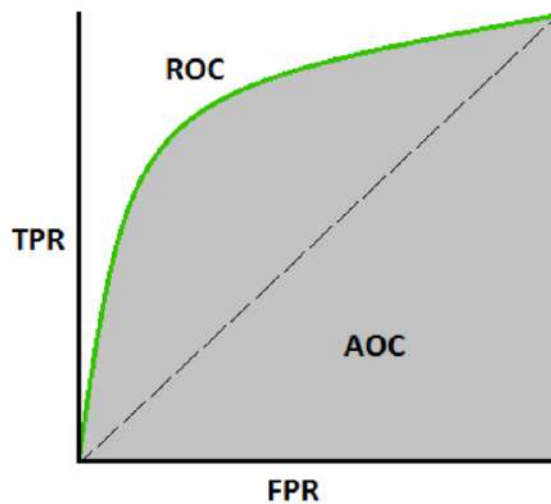
2.7.5. F1-Score

F1 Score merupakan perbandingan rata-rata presisi dan recall. Berikut rumus *F1 Score* yang digunakan.

$$F1-Score = \frac{2 \times Recall_i \times Precision_i}{Recall_i + Precision_i} \quad (2.7)$$

2.7.6. Kurva AUC-ROC

Kurva AUC-ROC merupakan pengukuran klasifikasi dengan menggunakan ambang batas. Kurva AUC-ROC divisualisasikan dalam bentuk kurva yang memiliki garis baseline. Jika kurva mendekati garis baseline maka performa model jelek, tapi jika kurva berada jauh di atas garis baseline maka performa model bagus. Kurva AUC-ROC digunakan untuk melihat seberapa baik model membedakan tiap kelas. Terdapat dua nilai pada kurva AUC-ROC yaitu *True Positive Rate* (TPR) dan *False Positive Rate* (FPR). Berikut rumus TPR dan FPR.



Gambar 2. 23 Kurva AUC-ROC

(Sumber gambar : towardsdatascience.com)

Seperti pada gambar 2.23 sumbu y merepresentasikan TPR sedangkan sumbu x merepresentasikan FPR. Garis putus-putus pada gambar merupakan garis ambang batas, sedangkan garis hijau merupakan pergerakan dari threshold. Semakin tinggi TPR dan semakin kecil FPR maka threshold semakin bagus.