

SKRIPSI

**KLASIFIKASI TINGKAT KEMATANGAN BUAH ANGGUR
TRANSFIGURASI MENGGUNAKAN METODE
CONVOLUTIONAL NEURAL NETWORK**

Disusun dan diajukan oleh:

**IRFADIANA NURHASANAH
D121 18 1501**



**PROGRAM STUDI SARJANA TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
GOWA
2023**

LEMBAR PENGESAHAN SKRIPSI

**KLASIFIKASI TINGKAT KEMATANGAN BUAH ANGGUR
TRANSFIGURASI MENGGUNAKAN METODE
CONVOLUTIONAL NEURAL NETWORK**

Disusun dan diajukan oleh

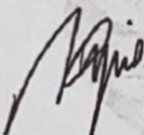
**Irfadiana Nurhasanah
D121181501**

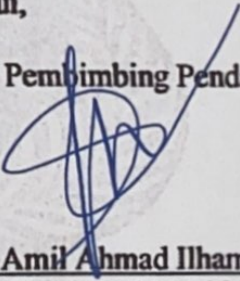
Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian
Studi Program Sarjana Program Studi Teknik Informatika
Fakultas Teknik Universitas Hasanuddin
Pada tanggal 27 Juli 2023
dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

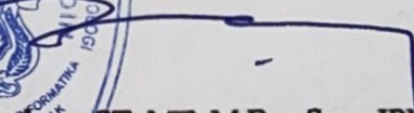
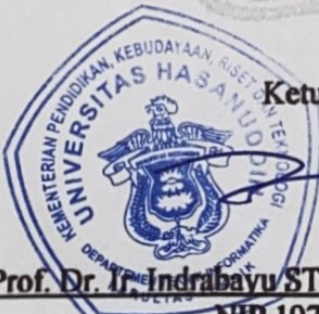
Pembimbing Utama,

Pembimbing Pendamping,


Dr. Ir. Ingrid Nurtanio, M.T.
NIP 19610813 198811 2 001


Dr. Amil Ahmad Ilham, S.T., M.IT
NIP 19731010 199802 1 001

Ketua Program Studi,



Prof. Dr. Ir. Indrabayu ST, MT, M.Bus.Sys., IPM, ASEAN. Eng
NIP 19750716 200212 1 004

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini ;

Nama : Irfadiana Nurhasanah
NIM : D121181501
Program Studi : Teknik Informatika
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

{Klasifikasi Tingkat Kematangan Buah Anggur Transfigurasi Menggunakan Metode Convolutional Neural Network }

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

Semua informasi yang ditulis dalam skripsi yang berasal dari penulis lain telah diberi penghargaan, yakni dengan mengutip sumber dan tahun penerbitannya. Oleh karena itu semua tulisan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis. Apabila ada pihak manapun yang merasa ada kesamaan judul dan atau hasil temuan dalam skripsi ini, maka penulis siap untuk diklarifikasi dan mempertanggungjawabkan segala resiko.

Segala data dan informasi yang diperoleh selama proses pembuatan skripsi, yang akan dipublikasi oleh Penulis di masa depan harus mendapat persetujuan dari Dosen Pembimbing.

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.

Gowa, 27 Juli 2023

Yang Menyatakan



Irfadiana Nurhasanah

ABSTRAK

IRFADIANA NURHASANAH. *Klasifikasi Tingkat Kematangan Buah Anggur Transfigurasi Menggunakan Metode Convolutional Neural Network* (dibimbing oleh Dr. Ir. Ingrid Nurtanio, M.T. dan Dr. Amil Ahmad Ilham, S.T., M.IT)

Anggur merupakan salah satu komoditas hortikultura yang termasuk dalam kelas pomologi atau frutikultur yang sangat digemari dikalangan masyarakat dan memiliki prospek ekonomi yang tinggi untuk dikembangkan. Anggur yang merupakan buah non klimaterik yang dimana pada proses pematangan pola respirasinya akan meningkat, kemudian turun secara perlahan-lahan. Oleh karena itu jika buah anggur dipanen belum cukup umur maka akan mempunyai kualitas rendah. Perkembangan ilmu pengetahuan dan teknologi pengolahan citra digital memungkinkan untuk memilah produk pertanian dan perkebunan secara otomatis.

Penelitian ini bertujuan untuk mengklasifikasikan tingkat kematangan dari buah anggur jenis transfigurasi, dimana akan dibagi ke dalam 3 kategori tingkat kematangan. Dengan menggunakan metode *Convolutional Neural Network* (CNN) yang akan mendeteksi ciri dari setiap citra untuk menentukan kategori kematangan anggur transfigurasi. Penelitian ini juga bertujuan untuk mengetahui berapa unjuk kerja dari sistem yang telah dibuat.

Sistem dibangun dengan menggunakan metode *Deep Learning Convolutional Neural Network* (CNN) dengan *Library Tensorflow*. Arsitektur model yang digunakan adalah *ResNet152v2*. Total data yang digunakan sebanyak 630 citra yang terbagi dalam 3 label kelas kematangan yaitu mentah, mengkal dan matang.

Hasil training dan validasi model dengan menggunakan 570 citra buah anggur transfigurasi diperoleh akurasi sebesar 94.79%. Hasil testing dengan menggunakan 60 citra buah anggur transfigurasi diperoleh akurasi sebesar 91.67%.

Kata Kunci: Anggur, Klasifikasi, *Deep Learning*, *Convolutional Neural Network*, *ResNet152v2*

ABSTRACT

IRFADIANA NURHASANAH. *Transfiguration Grape Ripeness Level Classification Using Convolutional Neural Network Method* (supervised by Dr. Ir. Ingrid Nurtanio, M.T. and Dr. Amil Ahmad Ilham, S.T., M.IT)

Grapes are one of the horticultural commodities that belongs to the pomology or fruticulture class which are very popular and have high economic prospects for development. Grapes are a non-climateric fruit where during the ripening process the respiration pattern will increase, then decrease slowly. Therefore, if the grapes are harvested underage, they will have low quality. The development of science and technology in digital and image processing makes it possible to sort agricultural and plantation products automatically.

This study aims to classify the ripeness level of transfiguration grapes, which will be divided into 3 categories of ripeness. By using the Convolutional Neural Network (CNN) method which will detect the characteristics of each image to determine the ripeness category of transfiguration grapes. This research also aims to know the performance of the system that has been made.

The system was built using the Deep Learning Convolutional Neural Network (CNN) method using Tensorflow Library. The model architecture used is ResNet152v2. Total data used was 630 images which are divided into 3 ripeness class labels, namely unripe, semi-ripe and ripe.

The model training and validation results using 570 transfiguration grapes images obtained an accuracy of 94.79%. The testing results using 60 transfiguration grapes images obtained an accuracy of 91.67%.

Keywords: Grape, Classification, Deep Learning, Convolutional Neural Network, ResNet152v2

DAFTAR ISI

LEMBAR PENGESAHAN SKRIPSI.....	i
PERNYATAAN KEASLIAN.....	ii
ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR	vi
DAFTAR TABEL.....	viii
DAFTAR SINGKATAN DAN ARTI SIMBOL	ix
DAFTAR LAMPIRAN.....	x
KATA PENGANTAR	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian/Perancangan.....	2
1.4 Manfaat Penelitian/Perancangan.....	3
1.5 Ruang Lingkup/Asumsi perancangan	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Anggur	4
2.2 <i>Artificial Neural Network</i> (ANN)	5
2.3 <i>Deep Learning</i>	7
2.4 <i>Convolutional Neural Network</i> (CNN).....	8
2.5 <i>Tensorflow</i>	14
2.6 <i>ResNet152v2</i>	16
2.7 <i>Confusion Matrix</i>	17
2.8 Akurasi, <i>Precision</i> , <i>Recall</i> dan <i>F1 Score</i>	18
BAB 3 METODE PENELITIAN/PERANCANGAN	20
3.1 Tahapan Penelitian.....	20
3.2 Waktu dan Lokasi Penelitian	21
3.3 Instrumen Penelitian.....	21
3.4 Teknik Pengambilan Data	22
3.5 Perancangan Sistem	23
3.6 Analisis Kerja Sistem.....	67
3.7 Implementasi Program	68
BAB 4 HASIL DAN PEMBAHASAN.....	70
4.1 Hasil Penelitian	70
4.2 Pembahasan.....	82
BAB 5 KESIMPULAN DAN SARAN.....	85
5.1 Kesimpulan	85
5.2 Saran.....	85
DAFTAR PUSTAKA	86
LAMPIRAN	88

DAFTAR GAMBAR

Gambar 2.1 Struktur <i>Artificial Neural Network</i>	6
Gambar 2.2 Struktur <i>Convolutional Neural Network</i>	9
Gambar 2.3 <i>Convolution Layer</i>	10
Gambar 2.4 <i>Max Pooling</i>	11
Gambar 2.5 Grafik Fungsi Aktivasi ReLU	12
Gambar 2.6 Ilustrasi Fungsi Aktivasi <i>Softmax</i>	13
Gambar 3.1 Tahapan Penelitian	20
Gambar 3.2 <i>Flowchart</i> Perancangan Sistem	23
Gambar 3.3 Contoh Data Penelitian.....	24
Gambar 3.4 Sampel Dataset Anggur Matang	25
Gambar 3.5 Histogram Anggur Matang.....	27
Gambar 3.6 Sampel Dataset Anggur Mengkal	28
Gambar 3.7 Histogram Anggur Mengkal.....	30
Gambar 3.8 Sampel Dataset Anggur Mentah	31
Gambar 3.9 Histogram Anggur Mentah.....	33
Gambar 3.10 Contoh Proses <i>Cropping</i>	34
Gambar 3.11 <i>Load Dataset from Google Drive</i>	36
Gambar 3.12 Pembagian <i>Dataset</i>	37
Gambar 3.13 Ilustrasi Arsitektur <i>ResNet152v2</i>	39
Gambar 3.14 Model <i>Summary</i>	45
Gambar 3.15 Proses Konvolusi.....	46
Gambar 3.16 Proses <i>Max Pooling</i>	51
Gambar 3.17 Proses <i>Global Average Pooling</i>	52
Gambar 3.18 Sampel Data Anggur Transfigurasi.....	53
Gambar 3.19 Ilustrasi Proses Konvolusi	53
Gambar 3.20 Nilai Matriks <i>Channel Red</i>	54
Gambar 3.21 Nilai Matriks <i>Channel Green</i>	55
Gambar 3.22 Nilai Matriks <i>Channel Blue</i>	55
Gambar 3.23 Proses Konvolusi Pada <i>Channel Red</i>	56
Gambar 3.24 Proses Konvolusi Pada <i>Channel Green</i>	57
Gambar 3.25 Proses Konvolusi Pada <i>Channel Blue</i>	59
Gambar 3.26 Proses <i>Max Pooling</i> Pada <i>Channel Red</i>	60
Gambar 3.27 Proses <i>Max Pooling</i> Pada <i>Channel Green</i>	61
Gambar 3.28 Proses <i>Max Pooling</i> Pada <i>Channel Blue</i>	61
Gambar 3.29 Proses <i>Global Average Pooling</i> Pada <i>Channel Red</i>	62
Gambar 3.30 Proses <i>Global Average Pooling</i> Pada <i>Channel Green</i>	62
Gambar 3.31 Proses <i>Global Average Pooling</i> Pada <i>Channel Blue</i>	63
Gambar 3.32 Proses <i>Flatten</i> dan <i>Fully Connected Layer Channel Red</i>	64
Gambar 3.33 Proses <i>Flatten</i> dan <i>Fully Connected Layer Channel Green</i>	65
Gambar 3.34 Proses <i>Flatten</i> dan <i>Fully Connected Layer Channel Red</i>	65
Gambar 3.35 Hasil Klasifikasi Citra Anggur	66
Gambar 3.36 Visualisasi Hasil Klasifikasi.....	67
Gambar 3.37 <i>Source Code</i> Visualisasi <i>Testing</i>	67
Gambar 3.38 Tampilan Awal <i>Website</i>	68
Gambar 3.39 Tampilan Hasil Klasifikasi.....	69

Gambar 4.1 Grafik <i>Accuracy Training</i> dan <i>Validation</i>	71
Gambar 4.2 Grafik <i>Loss Training</i> dan <i>Validation</i>	71

DAFTAR TABEL

Tabel 2.1 Perbedaan <i>Machine Learning</i> dan <i>Deep Learning</i>	8
Tabel 2.2 <i>Confusion Matrix</i>	17
Tabel 3.1 Rincian Pembagian <i>Dataset</i>	37
Tabel 4.1 Hasil Kesalahan Klasifikasi	74
Tabel 4.2 Hasil Klasifikasi Benar	77
Tabel 4.3 Hasil Perhitungan <i>Confusion Matrix</i>	81
Tabel 4.4 Strategi Evaluasi	82
Tabel 4.5 Strategi Evaluasi Menggunakan Parameter <i>Epoch</i>	83
Tabel 4.6 Strategi Evaluasi Menggunakan Parameter <i>Learning Rate</i>	84

DAFTAR SINGKATAN DAN ARTI SIMBOL

Lambang/Singkatan	Arti dan Keterangan
AIMP	<i>Artificial Intelligence and Multimedia Processing</i>
ANN	<i>Artificial Neural Network</i>
API	<i>Application Programming Interface</i>
BN	<i>Batch Normalization</i>
CNN	<i>Convolutional Neural Network</i>
FC	<i>Fully Connected</i>
GAP	<i>Global Average Pooling</i>
GPU	<i>Graphics Processing Unit</i>
JST	<i>Jaringan Saraf Tiruan</i>
MLP	<i>Multi Layer Perceptron</i>
NLP	<i>Natural Language Processing</i>
ReLU	<i>Rectified Linear Unit</i>
ResNet	<i>Residual Network</i>
∞	<i>Infinity</i>

DAFTAR LAMPIRAN

Lampiran 1 Contoh <i>Dataset</i>	89
Lampiran 2 <i>Source Code</i> Program	90
Lampiran 3 Hasil proses <i>Training</i> dan <i>Validasi</i>	97

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji dan syukur penulis panjatkan kepada Allah SWT atas berkat, rahmat, karunia dan pertolongan-Nya sehingga penulis dapat menyelesaikan tugas akhir dengan judul “Klasifikasi Tingkat Kematangan Buah Anggur Transfigurasi Menggunakan Metode Convolutional Neural Network” sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 di Departemen Teknik Informatika, Fakultas Teknik Universitas Hasanuddin.

Sholawat serta salam penulis haturkan kepada junjungan kita Nabi besar Muhammad SAW yang telah membawa kita dari zaman kebodohan menuju zaman penuh dengan ilmu pengetahuan.

Penulis menyadari bahwa penyusunan dan penulisan tugas akhir ini tidak dapat terselesaikan dengan baik tanpa adanya bantuan, bimbingan, serta dukungan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih banyak kepada :

1. Keluarga penulis, Ibu Sri Alpidawati dan Bapak Lajibir selaku orang tua penulis yang selalu memberikan dukungan, do'a, semangat dan kasih sayang serta selalu sabra dalam mendidik penulis sejak kecil, Irfadiani Nurannisa selaku kakak kandung penulis dan Irfadiani Nurhaliza selaku adik kandung penulis yang selalu memberikan dukungan, do'a, dan semangat yang tiada hentinya.
2. Ibu Dr. Ir. Ingrid Nurtanio, M.T. selaku pembimbing I dan Bapak Dr. Amil Ahmad Ilham, S.T., M.IT selaku pembimbing II, yang senantiasa menyediakan waktu, tenaga, pikiran dan bimbingan dalam mengarahkan penulis untuk menyelesaikan tugas akhir.
3. Segenap Dosen dan Staff Departemen Teknik Informatika, Fakultas Teknik Universitas Hasanuddin yang telah membantu penulis selama masa perkuliahan.
4. Para pengusaha Anggur di Kota Palu & Maura yang telah membantu penulis dalam pengambilan data penelitian, mulai dari survey awal di perkebunan

anggur sampai dengan selesainya proses pengambilan data secara keseluruhan.

5. Teman-teman AIMP Research Group Universitas Hasanuddin yang telah memberikan bantuan selama penelitian dan diskusi terkait penyusunan tugas akhir.
6. Saudara-saudari Synchronous 2018 yang telah memberikan nasihat, bantuan, dan semangat selama proses penyelesaian tugas akhir ini.
7. Sahabat seperjuangan semasa kuliah, Pirda dan Ayu yang telah membantu dan memberikan semangat serta hiburan di masa-masa sulit penyelesaian tugas akhir.
8. Teman-teman yang telah membantu penulis dalam proses penelitian, Pirda, Tami, Kamtina, Arian dan Reza yang telah membantu penulis dalam pengambilan data tambahan dan ikut memberikan support dalam penyelesaian tugas akhir penulis.
9. Serta pihak-pihak lain yang tidak disebutkan dan tanpa sadar telah menjadi inspirasi dan membantu penulis dalam menyelesaikan tugas akhir.

Penulis berharap semoga Allah SWT berkenan membalas segala bentuk bantuan dan kebaikan dari semua pihak yang turut membantu penyelesaian skripsi ini sebagai amalan ikhlas yang akan bermanfaat kelak. Penulis menyadari bahwa tugas akhir ini masih jauh dari kata sempurna. Oleh karena itu penulis megharapkan segala bentuk saran serta masukan yang membangun dari berbagai pihak. Semoga tugas akhir ini dapat memberi manfaat bagi para pembaca dan semua pihak. Aamiin.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Makassar, 27 Juli 2023

Penulis

BAB I PENDAHULUAN

1.1 Latar Belakang

Anggur merupakan salah satu komoditas yang termasuk ke dalam kelas pomologi atau frutikultur yang sangat digemari dikalangan masyarakat dan memiliki prospek ekonomi yang tinggi untuk dikembangkan. Anggur dipercaya dapat mencegah penyakit berbahaya seperti kanker, penyakit jantung, tekanan darah tinggi dan konstipasi. Anggur memiliki tekstur, rasa, dan keragaman jenis serta manfaat yang banyak sehingga buah anggur semakin digemari (Ni Kadek Sriani et al., 2020).

Anggur yang merupakan buah non klimaterik, yaitu pada proses pematangan pola respirasinya akan meningkat, kemudian turun secara perlahan-lahan. Oleh karena itu jika buah anggur dipanen belum cukup umur, maka akan mempunyai kualitas rendah, yaitu rasanya belum manis, adanya perubahan fisik yang meliputi warna, tekstur, susut berat, tebal tipisnya lapisan lilin sehingga daya tarik buah menjadi kurang bagus, disamping itu nilai gizinya terutama vitamin C akan rendah. Dalam keadaan seperti inilah buah tersebut kurang dapat diterima oleh konsumen atau pengelola buah (Budi Purwantiningsih et al., 2012).

Mengidentifikasi kematangan buah dalam jumlah banyak secara manual oleh manusia merupakan pekerjaan yang sulit, banyak memakan waktu dan biaya. Oleh karena itu, prediksi otomatis tingkat kematangan buah diperlukan. Kualitas buah jika dilihat dari citra bergantung pada aspek antara lain warna buah dan ukuran buah. Warna pada dasarnya digunakan untuk menentukan kematangan dari berbagai jenis buah seperti tomat, semangka, pisang, anggur, apel dan kurma. Aspek yang nampak dari buah digunakan untuk menentukan kematangannya, antara lain warna, ukuran, tekstur dan bentuk. Dari berbagai macam aspek ini, warna merupakan faktor utama yang mempengaruhi kualitas dan kematangan buah (Kaveri Kangune et al., 2019).

Perkembangan ilmu pengetahuan dan teknologi pengolahan citra digital memungkinkan untuk memilah produk pertanian dan perkebunan secara otomatis. Pengolahan citra merupakan alternatif untuk mengatasi masalah ini. Cara ini

memiliki kemampuan yang lebih peka karena dilengkapi sensor elektro-optik yang bisa dipastikan akan lebih tepat dan objektif dibandingkan dengan cara visual yang bersifat subjektif dan dipengaruhi oleh kondisi psikis pengamatnya. Kedepannya, pengolahan citra diharapkan akan menjadi salah satu pilihan dalam mengklasifikasikan tingkat kematangan buah anggur tanpa merusak sampel (objek).

Ada banyak metode yang bisa digunakan untuk melakukan pengolahan citra, salah satunya adalah menggunakan teknik Deep Learning (DL) yaitu metode Convolutional Neural Network (CNN) yang saat ini memiliki hasil paling signifikan dalam pengenalan citra. Hal tersebut dikarenakan CNN berusaha meniru sistem pengenalan citra pada visual cortex manusia sehingga memiliki kemampuan mengolah informasi citra.

Berdasarkan hal inilah, penulis mengusulkan judul **“Klasifikasi Tingkat Kematangan Buah Anggur Transfigurasi Menggunakan Metode Convolutional Neural Network”**.

1.2 Rumusan Masalah

1. Bagaimana proses klasifikasi tingkat kematangan buah anggur transfigurasi dari sebuah dataset image menggunakan metode CNN?
2. Bagaimana unjuk kerja sistem klasifikasi tingkat kematangan buah anggur transfigurasi menggunakan metode CNN?

1.3 Tujuan Penelitian/Perancangan

1. Untuk melihat hasil proses klasifikasi tingkat kematangan buah anggur transfigurasi dari sebuah dataset image menggunakan metode CNN.
2. Untuk mengetahui unjuk kerja sistem klasifikasi tingkat kematangan buah anggur transfigurasi menggunakan metode CNN.

1.4 Manfaat Penelitian/Perancangan

Dengan dilakukannya penelitian ini, diharapkan manfaat yang didapatkan antara lain :

1. Bagi Industri, penelitian ini diharapkan dapat menjadi inovasi bagi para pengusaha anggur transfigurasi dalam mengidentifikasi kematangan buah anggur transfigurasi dalam jumlah banyak dan meminimalisir *human error* pada saat penentuan tingkat kematangan buah anggur transfigurasi.
2. Bagi Peneliti, penelitian ini diharapkan dapat dijadikan referensi ilmiah untuk penelitian-penelitian selanjutnya terkait klasifikasi dengan metode CNN.

1.5 Ruang Lingkup/Asumsi perancangan

1. Jenis anggur yang digunakan adalah anggur transfigurasi
2. Parameter yang digunakan adalah warna buah
3. Metode yang digunakan adalah metode Convolutional Neural Network (CNN)

BAB II

TINJAUAN PUSTAKA

2.1 Anggur

Anggur (*Vitis Vinivera*) adalah jenis buah-buahan yang tumbuh dalam kelompok pada tanaman merambat berbatang kayu. Buah ini berasal dari daerah Timur Tengah dan Asia Barat, namun kini telah menyebar ke seluruh dunia dan menjadi buah yang populer di berbagai negara. Buah anggur biasanya dikonsumsi segar, dijadikan jus atau diolah menjadi produk makanan dan minuman seperti *wine*, *jelly*, dan *raisin*.

Anggur merupakan tanaman buah yang memiliki ciri- ciri berbentuk perdu dan batang merambat. Dari batang-batang tersebut buah anggur akan muncul. Buah dari tanaman ini memiliki warna yang bermacam-macam seperti hitam, merah, dan hijau, selain itu buah anggur juga memiliki rasa yang manis dan kandungan vitamin yang tinggi terutama vitamin C. Menurut Marhuman *et al.* (2016) buah anggur mengandung vitamin C sebanyak 23,23 mg/100 g. Kandungan vitamin yang tinggi tersebut menyebabkan daya beli masyarakat juga tinggi.

Anggur adalah buah yang kaya akan nutrisi dan senyawa bioaktif yang berkhasiat untuk kesehatan. Beberapa manfaat kesehatan yang dikaitkan dengan konsumsi anggur antara lain : Mencegah penyakit kardiovaskular karena anggur mengandung senyawa polifenol yang dapat membantu melindungi kesehatan jantung dan mencegah penyakit kardiovaskular. Beberapa penelitian menunjukkan bahwa konsumsi anggur dapat menurunkan kadar kolesterol LDL (kolesterol jahat) dan meningkatkan kadar kolesterol HDL (kolesterol baik) dalam darah. Selain itu, anggur juga dapat membantu menjaga elastisitas pembuluh darah dan meningkatkan sirkulasi darah. (Chiva *et al.* 2017). Senyawa polifenol dalam anggur juga memiliki potensi anti-kanker. Penelitian menunjukkan bahwa senyawa polifenol seperti resveratrol dalam anggur dapat membantu menghambat pertumbuhan sel kanker. (Mgbonyebi *et al.* 2006). Selain itu, senyawa polifenol dalam anggur juga dapat meningkatkan kesehatan otak dan melindungi sel-sel otak dari kerusakan oksidatif dan peradangan. Beberapa penelitian menunjukkan bahwa konsumsi anggur dapat membantu meningkatkan memori, kognisi dan kemampuan

belajar pada orang dewasa yang sehat maupun yang mengalami gangguan kognitif ringan. (Al-Yousuf *et al.* 2018).

Anggur memiliki banyak manfaat kesehatan karena mengandung antioksidan, seperti resveratrol yang dapat membantu melindungi sel-sel dari kerusakan akibat radikal bebas. Selain itu, anggur juga mengandung vitamin C dan K, asam folat dan kalium yang dapat membantu menjaga kesehatan jantung, meningkatkan sistem kekebalan tubuh dan meningkatkan kesehatan tulang. (Melissa, 2023)

Anggur adalah salah satu komoditas pertanian yang memiliki prospek yang menjanjikan di Indonesia. Berikut ini adalah beberapa faktor yang mendukung prospek anggur di bidang ekonomi Indonesia :

1. Permintaan yang terus meningkat : permintaan pasar domestik terhadap anggur semakin meningkat karena semakin banyak masyarakat Indonesia yang memilih minuman anggur untuk minuman olahraga dan minuman bersoda yang menyebabkan permintaan anggur untuk keperluan industri juga meningkat. (Amiruddin, 2017).
2. Potensi ekspor yang besar : Indonesia memiliki potensi untuk mengekspor anggur ke negara-negara di kawasan Asia Tenggara dan Timur Tengah karena anggur Indonesia memiliki kualitas yang baik dan harga yang kompetitif. (Yulianti *et al.* 2019).
3. Potensi pengembangan agrowisata : Anggur dapat menjadi salah satu produk unggulan dalam pengembangan agrowisata di Indonesia karena banyak wisatawan yang tertarik untuk mengunjungi kebun anggur dan membeli produk-produk olahan anggur. (Puspitarini *et al.* 2021).
4. Potensi peningkatan produktivitas : Teknologi pertanian modern dapat membantu meningkatkan produktivitas dan kualitas anggur di Indonesia, sehingga dapat memperkuat daya saing anggur Indonesia di pasar domestik dan internasional. (Amiruddin, 2017).

2.2 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) adalah sebuah model matematis yang terinspirasi oleh cara kerja sistem saraf biologi, terutama pada manusia dan hewan. Jaringan syaraf tiruan terdiri dari sejumlah besar

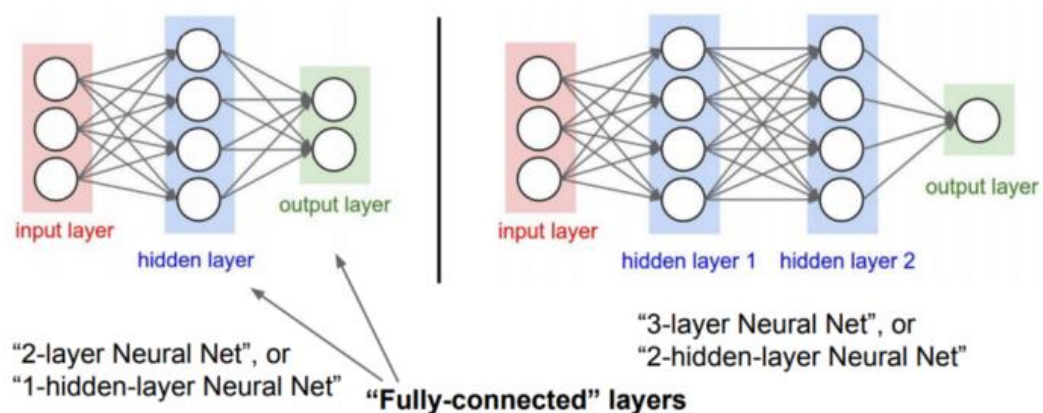
elemen pemrosesan (*neuron*) yang terhubung dalam satu atau beberapa lapisan. Neuron-neuron tersebut menerima *input* dan menghasilkan *output* berdasarkan koneksi dan bobotnya masing-masing, sehingga mampu melakukan tugas seperti klasifikasi, pengenalan pola dan prediksi.

JST sangat berguna dalam bidang-bidang seperti pengenalan suara dan gambar, pengenalan teks, dan penerjemahan bahasa alami atau NLP (*Natural Language Processing*). JST juga digunakan dalam industri untuk memprediksi penjualan, mengoptimalkan rantai pasokan, dan meningkatkan kualitas produksi. (Ian, 2016).

Artificial Neural Network (ANN) merupakan set algoritma yang bekerja seperti jaringan saraf otak manusia, dimana neuron saling terhubung satu dengan lainnya, bekerja untuk memproses informasi. Tujuan utama dari ANN adalah menjadikan komputer memiliki kemampuan kognitif seperti otak manusia, memiliki kemampuan *problem solving* dan dapat melakukan proses pembelajaran. (Laksamana, 2022).

ANN adalah sekumpulan neuron yang terorganisir dalam lapisan-lapisan (*layers*), diantaranya :

- *Input Layer* : lapisan yang membawa data masuk kedalam sistem untuk kemudian diproses pada layer selanjutnya.
- *Hidden Layer* : lapisan antara *input layer* dan *output layer*, dimana *artificial neuron* yang memiliki sekumpulan input pembobot '*weight*' dan prosedur untuk menghasilkan *output* neuron melalui *activation function*.
- *Output Layer* : lapisan terakhir dari neuron yang menghasilkan *output* sistem.



Gambar 2.1 Struktur *Artificial Neural Network*

2.3 Deep Learning

Deep learning adalah suatu cabang dari *machine learning* yang menggunakan algoritma *neural network* yang sangat kompleks dan terdiri dari banyak lapisan (*deep*). Tujuan dari *deep learning* adalah untuk mempelajari suatu representasi fitur dari data dengan menggunakan beberapa lapisan pemrosesan data secara bertahap. Dalam *deep learning*, setiap lapisan *neural network* terdiri dari sejumlah besar unit pengolahan sinyal (*neuron*) yang terhubung satu sama lain dalam pola tertentu.

Secara umum, *deep learning* digunakan untuk memecahkan masalah yang sangat kompleks dan memerlukan representasi data yang sangat abstrak, seperti pengenalan gambar dan suara, pengenalan bahasa alami, dan tugas-tugas lain yang terkait dengan data kompleks. Misalnya, dalam pengenalan gambar, *deep learning* dapat digunakan untuk memperbaiki kinerja pengenalan gambar dengan cara memperkenalkan banyak lapisan pemrosesan gambar yang semakin kompleks dan mampu mengenali fitur yang semakin abstrak. Sebagai hasilnya, *deep learning* mampu memecahkan masalah yang sebelumnya dianggap tidak mungkin dipecahkan oleh mesin. (Ian *et al.* 2016)

Deep learning juga memiliki beberapa keunggulan lain, seperti kemampuan untuk mengenali pola yang sangat kompleks dan abstrak, kemampuan untuk mengekstraksi fitur yang relevan secara otomatis, dan kemampuan untuk memperbaiki performa dengan data yang lebih banyak. Namun, *deep learning* juga memiliki beberapa kelemahan, seperti kebutuhan akan data yang sangat besar dan kemampuan komputasi yang sangat tinggi. (Yan *et al.* 2015)

Beberapa contoh aplikasi *deep learning* yang populer saat ini adalah pengenalan wajah seperti pada sistem keamanan atau pembuka kunci ponsel, pendeteksian objek seperti pada sistem keamanan atau kendaraan otonom, prediksi harga saham dengan menggunakan data pasar yang akan dikenali pola-pola harga sahamnya dan memberikan prediksi yang akurat, pemrosesan bahasa alami seperti asisten virtual atau sistem terjemahan otomatis yang dilatih dengan menggunakan data teks, pengolahan citra medis, pengenalan suara, dan pembelajaran penguasaan permainan seperti permainan *Go*. Aplikasi lain dari *deep learning* meliputi bidang kesehatan, keuangan, dan teknologi informasi.

Perbedaan antara *machine learning* dan *deep learning* dapat dilihat pada tabel berikut :

Tabel 2.1 Perbedaan *Machine Learning* dan *Deep Learning*

<i>Machine Learning</i>	<i>Deep Learning</i>
Hasil bagus dengan <i>dataset</i> sedikit	Membutuhkan <i>dataset</i> yang banyak
Melatih model dengan cepat	Komputasi secara intensif
Perlu mencoba berbagai fitur dan klasifikasi manual untuk mendapatkan hasil terbaik	Mempelajari fitur dan pengklasifikasi otomatis
Tingkat akurasi tinggi	Akurasi tidak terbatas

Sumber : The MathWorks, Inc (2017, p.10)

2.4 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah tipe *neural network* yang khusus dirancang untuk memproses data berupa gambar dan citra. CNN menggunakan operasi konvolusi untuk mengekstrak fitur atau pola dari gambar, dan lapisan-lapisan yang berbeda pada CNN dapat mempelajari fitur-fitur yang semakin kompleks pada gambar.

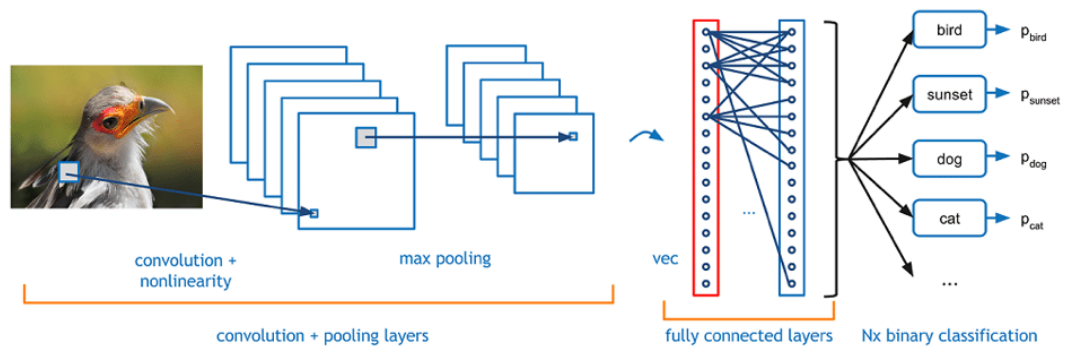
CNN terdiri dari lapisan *input*, lapisan konvolusi, lapisan *pooling*, dan lapisan *fully connected*. Lapisan konvolusi adalah lapisan yang paling penting pada CNN karena itulah tempat dimana proses pengenalan pola dan fitur terjadi. Sedangkan, lapisan *pooling* berfungsi untuk mengurangi dimensi dari data dan mengurangi pengaruh dari variabilitas lokasi pada gambar.

Beberapa contoh penggunaan CNN adalah dalam pengenalan objek, klasifikasi gambar, dan deteksi wajah. Dalam pengenalan objek, CNN dapat mengenali objek pada gambar dengan akurasi yang tinggi berdasarkan fitur-fitur yang dikenalnya. (Bolei *et al.* 2014).

Convolutional Neural Network (CNN) dalam *Deep Learning* tergolong kedalam *deep neural network* yang banyak digunakan dalam analisis citra (*visual imagery*). CNN merupakan versi *regularization* dari *multi-layer perceptron* dan

dibangun oleh banyak *convolution layer* dengan fungsi aktivasi yang kebanyakan adalah *ReLU*. (Laksamana, 2022).

Convolutional Neural Network (CNN) merupakan jaringan saraf yang digunakan khusus untuk mengolah data berstruktur grid, salah satunya berupa citra dua dimensi. Proses konvolusi merupakan operasi aljabar linear yang mengkalikan matriks dari filter pada citra yang akan diproses. Proses tersebut dinamakan dengan lapisan konvolusi dan merupakan salah satu jenis dari banyak lapisan yang bisa dimiliki dalam satu jaringan.

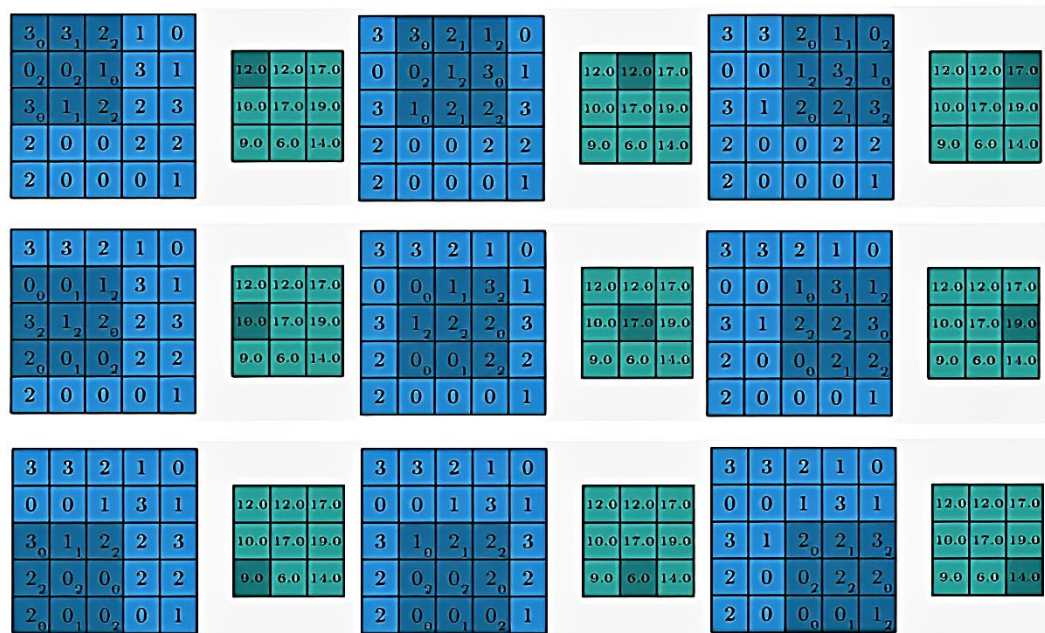


Gambar 2.2 Struktur *Convolutional Neural Network*

2.4.1 *Convolution Layer*

Convolution Layer pada CNN bekerja dengan melakukan operasi konvolusi pada gambar *input* atau *feature map* dengan *set filter* atau *kernel*. *Filter* atau *kernel* adalah matriks kecil yang digunakan untuk mengekstrak fitur-fitur penting dari gambar, seperti garis, sudut dan pola-pola kompleks lainnya. Proses konvolusi menghasilkan *output* berupa *feature map* yang merupakan representasi dari gambar *input* yang telah dilakukan ekstraksi fitur.

Gambar 2.3 merupakan contoh cara kerja dari *convolution layer* pada CNN:



Gambar 2.3 *Convolution Layer*

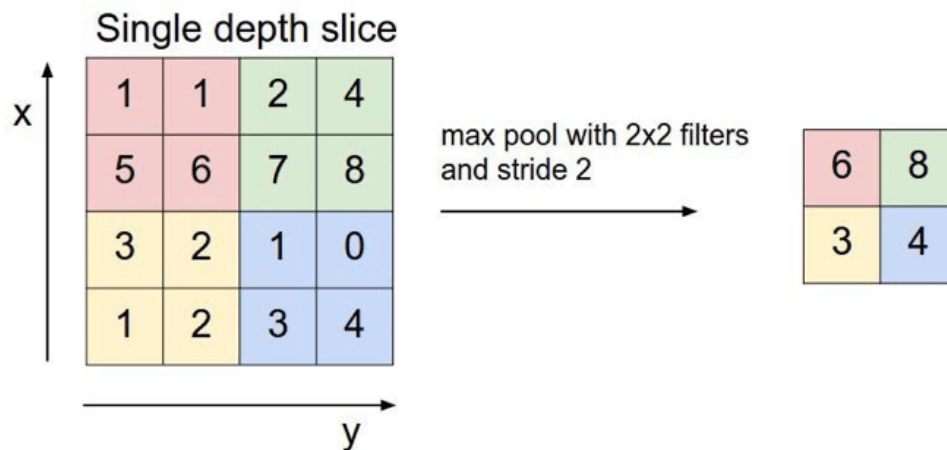
Pada Gambar 2.3 diatas, filter bergerak di atas gambar input dan melakukan operasi konvolusi untuk menghasilkan *feature map*. Contoh *filter* yang digunakan pada gambar adalah filter *edge detection* (menemukan tepi gambar). Setelah itu, *feature map* dihasilkan dan selanjutnya dapat digunakan sebagai *input* untuk *layer* berikutnya pada CNN.

Setiap *filter* pada *convolution layer* mempunyai bobot atau *weight* yang diatur secara acak pada awalnya dan kemudian diperbaiki melalui proses pelatihan CNN. Proses pelatihan atau *training* ini dilakukan dengan mengoptimalkan parameter-parameter bobot pada setiap *filter*, sehingga CNN dapat mengenali objek-objek dalam gambar dengan akurasi yang tinggi. (Irhum, 2018).

2.4.2 *Pooling Layer*

Pooling Layer ada setelah *convolutional layer*. Terdiri dari sebuah filter dengan ukuran dan *stride* tertentu yang akan bergeser pada seluruh *activation map*. *Pooling* yang sering digunakan adalah *max pooling* dan *average pooling*. Contoh penggunaannya apabila *max pooling* yang digunakan 2 x 2 dengan *stride* 2, maka disetiap pergeseran filter, nilai maksimum pada area 2 x 2 pixel

tersebut yang akan dipilih, sedangkan *average pooling* akan memilih nilai rata-ratanya. *Pooling layer* digunakan untuk mempercepat komputasi karena parameter yang harus di update semakin sedikit dan mengatasi *overfitting*. (Nabila, 2018).



Gambar 2.4 *Max Pooling*

2.4.3 *Fully Connected Layer*

Fully Connected Layer merupakan lapisan dimana semua neuron dari lapisan-lapisan sebelumnya dijadikan satu untuk dilakukan proses klasifikasi menggunakan *neural network*. Pada dasarnya lapisan ini sama halnya dengan lapisan *neural network* biasa, bisa dalam bentuk *single net* ataupun MLP. Sebelum dilakukan proses klasifikasi, *feature map* yang dihasilkan dari *feature learning* tersebut masih berbentuk *multi dimensional array*, sehingga perlu mengubahnya menjadi bentuk vektor, teknik ini disebut dengan *flatten*. *Flatten* merupakan teknik untuk *reshape feature map* menjadi sebuah vektor agar bisa digunakan sebagai *input* dari *fully connected layer*. Jadi *input* dari *fully connected layer* terdiri dari satu *neuron* hasil *reshape feature map* tadi menjadi vektor. Setelah dilakukan *flatten*, semua bobot tersebut akan diklasifikasi sesuai dengan banyaknya kelas.

Pada *fully connected layer* ini, tidak ada ketentuan pasti untuk menggunakan jaringan *single net* ataupun MLP. Karena proses ekstraksi dan *feature learning* sudah dilakukan pada *layer* sebelumnya, beban jaringan *neural network* untuk melakukan proses klasifikasi relative lebih ringan. Tapi dalam beberapa penelitian sebelumnya, menunjukkan dengan menggunakan

jaringan MLP dapat meningkatkan akurasi klasifikasi meskipun perbedaannya tidak terlalu signifikan.

2.4.4 Fungsi Aktivasi

Fungsi aktivasi merupakan operasi matematik yang dikenakan pada sinyal output y . Fungsi aktivasi berfungsi menentukan apakah suatu *neuron* aktif atau tidak berdasarkan *weighter sum* dari suatu *input*. Penelitian ini menggunakan dua fungsi aktivasi yaitu fungsi aktivasi ReLU dan fungsi aktivasi softmax. Penjelasan dari kedua fungsi tersebut sebagai berikut.

a. ReLU

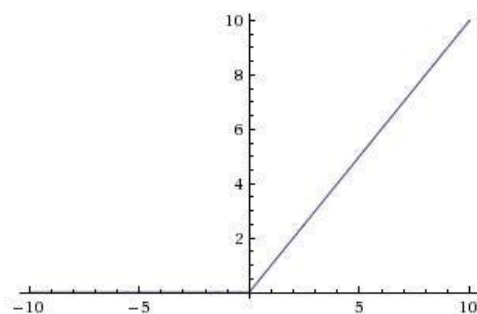
Fungsi aktivasi *Rectified Linear Unit* (ReLU) merupakan fungsi aktivasi sederhana yang memiliki kepentingan praktis khusus karena perhitungannya yang cepat. Kelebihan fungsi aktivasi ReLU dibandingkan dengan fungsi aktivasi lain adalah sebagai berikut.

1. Fungsi aktivasi ReLU merupakan fungsi aktivasi *default* ketika mengembangkan *multilayer perceptron* dan *convolutional neural network*.
2. Fungsi aktivasi ReLU mengatasi masalah *gradient descent* yang hilang, yang memungkinkan model belajar lebih cepat dan berkinerja lebih baik.
3. Menemukan cara melatih jaringan dengan lebih cepat, sehingga mengurangi kemungkinan terjadinya *overfitting*.

Fungsi aktivasi ReLU memetakan input ke 0 jika negatif dan mempertahankan nilainya jika positif. Representasi fungsi ReLU adalah sebagai berikut

$$f_{ReLU}(x) = \max(0, x) \quad (2.1)$$

Adapun grafiknya dapat dilihat pada Gambar 2.5



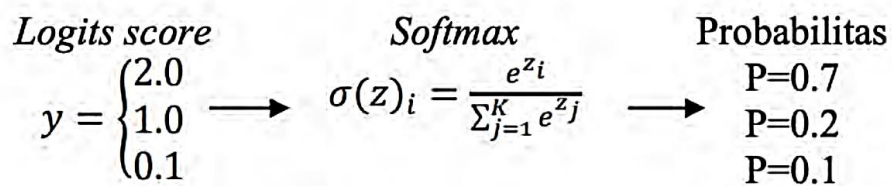
Gambar 2.5 Grafik Fungsi Aktivasi ReLU

b. *Softmax*

Fungsi aktivasi *softmax* merupakan fungsi input vektor dari bilangan real K , yang kemudian dinormalkan menjadi distribusi probabilitas yang terdiri atas probabilitas K yang proporsional ke eksponensial input. Komponen vektor pada *softmax* memiliki interval $(0,1)$. Fungsi *softmax* merupakan lapisan yang menghubungkan antara *fully connected layer* dengan *dense connection*. *Softmax* berfungsi untuk menghitung probabilitas pada setiap kelas target yang memungkinkan dan akan membantu menentukan kelas target pada *input* yang diberikan. Nilai *softmax* berada pada rentang probabilitas *output* dari 0 hingga 1 dan jumlah semua probabilitas sama dengan satu. Definisi fungsi *softmax* $\sigma = \mathbb{R}^K \rightarrow \mathbb{R}^K$ dituliskan sebagai persamaan berikut:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (2.2)$$

dimana $i = 1, \dots, K$ dan $z = (z_1, \dots, z_K) \in \mathbb{R}^K$. Adapun contoh ilustrasi fungsi aktivasi *softmax* dapat dilihat pada Gambar 2.6 sebagai berikut.



Gambar 2.6 Ilustrasi Fungsi Aktivasi *Softmax*

Logits score menunjukkan lapisan *neuron* terakhir sebagai *output* mentah pada lapisan terakhir *neural network* sebelum proses aktivasi berlangsung. Setelah *output* diproses dengan *softmax* akan menghasilkan nilai probabilitas dengan jumlah 1.

2.4.5 Operasi Konvolusi

Operasi konvolusi adalah operasi pada dua fungsi argument bernilai nyata. Operasi ini menerapkan fungsi *output* sebagai *feature map* dari *input* citra. *Input* dan *output* ini dapat dilihat sebagai dua argument bernilai riil. secara formal operasi konvolusi dapat ditulis dengan rumus berikut :

$$s(t) = (x * w)(t) \quad (2.3)$$

Fungsi $s(t)$ memberikan output tunggal berupa *feature map*, argumen pertama adalah *input* yang merupakan x dan argumen kedua w sebagai *kernel* atau filter. Jika kita melihat *input* sebagai citra dua dimensi, maka kita bisa mengasumsikan t sebagai piksel dan menggantinya dengan i dan j . Operasi untuk konvolusi ke *input* dengan lebih dari satu dimensi dapat ditulis sebagai berikut :

$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n) \quad (2.4)$$

Persamaan di atas merupakan perhitungan dasar dalam operasi konvolusi dimana i dan j adalah piksel dari citra. Perhitungannya bersifat komutatif dan muncul saat K sebagai *kernel*(filter), I sebagai *input* dan *kernel* yang relatif dapat dibalik terhadap *input*. Sebagai alternatif, operasi konvolusi dapat dilihat sebagai perkalian matriks antara citra masukan dan *kernel* dimana keluarannya dapat dihitung dengan *dot product*.

2.5 Tensorflow

Tensorflow merupakan *open source framework* yang dapat digunakan untuk mengembangkan, melatih, dan menggunakan model deteksi objek. Sistem ini sudah banyak diterapkan pada berbagai produk *Google* antara lain pencarian *image*, deteksi wajah, dan plat nomor kendaraan pada *google street, view, google assistant, way mo* atau *self driving car*, dan lain-lain.

Tensorflow bekerja dengan *computational* untuk membuat model *machine learning*. *Tensorflow* menyediakan berbagai *toolkit* yang memungkinkan anda membuat model pada tingkat yang lebih rendah untuk membuat model dengan menentukan serangkaian matematis. Sebagai alternatif, anda dapat menggunakan API dengan tingkat yang lebih rendah untuk membuat model dengan menentukan API dengan tingkat yang lebih tinggi (seperti *tf.estimator*) untuk menentukan arsitektur yang telah ditetapkan, seperti regresi linear atau *neural network*.

Framework Tensorflow digunakan pada proses pembuatan sistem deteksi objek agar memudahkan implementasi algoritma dan penggunaan bahasa pemrograman, terdapat juga GPU untuk mempercepat proses *training*. *Tensorflow* sebagai kerangka *machine learning* yang dapat digunakan untuk mengolah banyak data atau ingin mempelajari kecerdasan buatan (*artificial intelligence*) secara mendalam.

Tensorflow bekerja dengan cara membangun sebuah *graph* komputasi yang terdiri dari serangkaian operasi matematika yang saling terkait. Setiap operasi dalam *graph* tersebut dapat menerima satu atau lebih *tensor* sebagai *input* dan menghasilkan satu atau lebih *tensor* sebagai *output*. *Tensor* sendiri adalah sebuah *array* multidimensi yang digunakan untuk merepresentasikan data dalam model *machine learning*.

Tensorflow memungkinkan pengguna untuk membangun model *machine learning* dengan mudah dan cepat dengan menggunakan API yang intuitif. Pengguna dapat memilih antara menggunakan API *high-level* seperti *Keras* atau API *low-level* yang lebih fleksibel untuk membangun model sesuai dengan kebutuhan. Selain itu, *Tensorflow* juga menyediakan *tools* untuk memvisualisasikan *graph* komputasi, melakukan *debugging*, dan melakukan distribusi komputasi di *multiple GPUs* atau mesin. (Martin *et al.* 2016).

Berikut ini adalah beberapa kelebihan dari *Tensorflow* :

1. Kompatibilitas *Platform* yang luas : *tensorflow* dapat digunakan pada berbagai *platform* seperti *desktop*, *server*, *mobile*, dan *edge devices*. Hal ini membuat *tensorflow* sangat fleksibel dan dapat diintegrasikan ke dalam berbagai aplikasi.
2. Kecepatan dan Skalabilitas : *tensorflow* dirancang untuk dapat memproses data yang sangat besar dan kompleks dengan kecepatan yang tinggi. Hal ini menjadikan *tensorflow* sebagai salah satu *framework* tercepat yang tersedia saat ini.
3. Dukungan komunitas yang luas : *tensorflow* memiliki komunitas yang besar dan aktif. Hal ini berarti bahwa ada banyak sumber daya dan dukungan yang tersedia untuk para pengguna termasuk dokumentasi, *tutorial*, dan forum diskusi.
4. Memiliki banyak fitur : *tensorflow* menyediakan berbagai jenis *layer* dan model yang dapat digunakan untuk membangun berbagai jenis model *machine learning*, seperti model *deep learning* dan *reinforcement learning*. *Tensorflow* juga menyediakan berbagai algoritma optimasi yang dapat digunakan untuk meningkatkan performa model. (Martin *et al.* 2016).

2.6 ResNet152v2

ResNet152v2 adalah sebuah model jaringan saraf tiruan dalam bidang *computer vision* yang digunakan untuk memecahkan masalah klasifikasi gambar. *ResNet152v2* merupakan salah satu varian dari model *ResNet* yang dikembangkan oleh *Microsoft Research* pada tahun 2015. Model ini terdiri dari 152 lapisan atau *layer* yang lebih dalam dibandingkan model *ResNet* yang lebih awal.

ResNet152v2 didasarkan pada konsep “*residual learning*” yang memungkinkan model untuk mengatasi permasalahan yang dihadapi oleh model yang sangat dalam dengan memungkinkan informasi untuk “melewati” beberapa lapisan. Dalam *ResNet152v2*, blok-blok dasar terdiri dari lapisan konvolusi, *batch normalization*, dan aktivasi *ReLU*. Setiap blok juga memiliki jalur *skip connection* atau *shortcut connection* yang memungkinkan informasi untuk melewati beberapa blok. *ResNet152v2* telah terbukti berhasil dalam berbagai tugas pengenalan gambar, termasuk klasifikasi gambar dan deteksi objek. Model ini telah dicatat memiliki akurasi yang lebih baik dibandingkan dengan model-model yang lebih pendek seperti *ResNet50* atau *ResNet101*. (Kaiming *et al.* 2016).

ResNet152v2 merupakan *pre trained model* atau model yang telah dilatih sebelumnya sehingga tidak memerlukan konfigurasi khusus untuk mengatur *layer* di dalamnya. *ResNet* merupakan kepanjangan dari *Residual Network* yang merupakan jenis jaringan saraf tertentu yang diperkenalkan pada tahun 2015 oleh Kaiming He, Xiangyu Zhang, Shaoqing Ren dan Jian Sun. Dasar dari arsitektur *ResNet* adalah CNN yang terdiri dari *convolutional layer*, *pooling layer*, dan *fully connected layer* serta penambahan *skip connection* pada setiap beberapa *convolution layer*. Prinsip kerja *ResNet* adalah membangun jaringan yang lebih dalam dibandingkan dengan jaringan biasa lainnya dan secara bersamaan menemukan jumlah lapisan yang dioptimalkan untuk meniadakan masalah gradien yang hilang. (Andreanov *et al.* 2022).

Cara kerja *ResNet152v2* adalah dengan memanfaatkan *residual block* yang berisi *shortcut connection* atau *skip connection*. *Shortcut connection* adalah cara untuk menghubungkan *layer-layer* pada jaringan yang tidak bersebelahan dengan *layer-layer* pada jaringan yang lebih dalam. Hal ini memungkinkan informasi yang hilang pada *layer-layer* sebelumnya dapat tetap diteruskan ke *layer-layer*

selanjutnya pada jaringan yang lebih dalam. Dalam *ResNet152v2*, terdapat beberapa jenis *residual block* yang digunakan, yaitu *residual block* dengan 2 layer, *residual block* dengan 3 layer, dan *residual block* dengan 4 layer. (Kaiming *et al.* 2016).

Beberapa kelebihan *ResNet152v2* dibandingkan dengan arsitektur CNN lainnya adalah :

1. Kemampuan untuk melatih model yang lebih dalam (*deep*) : *ResNet152v2* memiliki lebih banyak lapisan daripada arsitektur CNN lainnya, sehingga dapat memperdalam model dan meningkatkan akurasi.
2. Kemampuan untuk menghindari *overfitting* : *ResNet152v2* menggunakan teknik “*skip connection*” yang memungkinkan gradien informasi melintasi beberapa lapisan sekaligus sehingga membantu mencegah *overfitting*.
3. Kemampuan untuk mempercepat proses pelatihan : *ResNet152v2* menggunakan teknik “*batch normalization*” yang memungkinkan proses pelatihan atau *training* menjadi lebih cepat dan lebih stabil.
4. Kemampuan untuk menghasilkan fitur yang lebih baik : *ResNet152v2* memiliki kemampuan untuk mengekstrak fitur-fitur yang lebih baik dari gambar. (Kaiming *et al.* 2016).

2.7 Confusion Matrix

Confusion Matrix adalah alat evaluasi visual yang digunakan dalam sistem klasifikasi. *Confusion matrix* ini berguna untuk mengukur seberapa baik model klasifikasi yang telah dibuat. *Confusion matrix* berukuran $n \times n$, dimana n merupakan jumlah kelas yang berbeda. *Confusion matrix* menentukan akurasi yang didapat dari nilai beberapa parameter, seperti *true positive* (TP), *false positive* (FP), *true negative* (TN), dan *false negative* (FN).

Tabel *confusion matrix* ditunjukkan pada Tabel 2.2 berikut :

Tabel 2.2 *Confusion Matrix*

Kelas Sebenarnya	Kelas Hasil Klasifikasi	
	<i>Predicted</i>	<i>Predicted</i>
<i>Actual</i>	<i>True Positive</i> (TP)	<i>True Negative</i> (TN)
<i>Actual</i>	<i>False Positive</i> (FP)	<i>False Negative</i> (FN)

- a. TP adalah *true positive*, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem.
- b. TN adalah *true negative*, yaitu jumlah data negative yang terklasifikasi dengan benar oleh sistem.
- c. FN adalah *false negative*, yaitu jumlah data negative namun terklasifikasi salah oleh sistem.
- d. FP adalah *false positive*, yaitu jumlah data positif namun terklasifikasi salah oleh sistem.

Berdasarkan nilai-nilai *true positive* (TP), *false positive* (FP), *true negative* (TN), dan *false negative* (FN), evaluasi klasifikasi dianalisis dari beberapa indikator yaitu diantaranya indikator akurasi, spesifisitas, dan sensitifitas. Akurasi merupakan rasio antara jumlah terprediksi benar dari semua data. Spesifisitas merupakan nilai yang menunjukkan banyak data bernilai negatif yang mampu terklasifikasi dengan benar masuk ke dalam kelas negatif. Sensitifitas merupakan nilai yang menunjukkan banyak data bernilai positif yang mampi terklasifikasi dengan benar masuk ke dalam kelas positif. Indikator tersebut dihitung dengan persamaan berikut :

$$Akurasi = \left(\frac{TP+TN}{TP+TN+FP+FN} \right) \times 100\% \quad (2.5)$$

$$Spesifisitas = \left(\frac{TN}{TN+FP} \right) \times 100\% \quad (2.6)$$

$$Sensitifitas = \left(\frac{TP}{TP+FN} \right) \times 100\% \quad (2.7)$$

2.8 Akurasi, *Precision*, *Recall* dan *F1 Score*

Dalam mengevaluasi *performance* algoritma dari *Machine Learning* (ML) khususnya *Supervised Learning*, kita menggunakan acuan *Confusion Matrix*. *Confusion Matrix* merepresentasikan prediksi dan kondisi sebenarnya (aktual) dari data yang dihasilkan oleh algoritma ML. Berdasarkan *Confusion Matrix*, kita bisa menentukan *accuracy*, *precision* dan *recall*. (Resika, 2019).

Accuracy didefinisikan sebagai persentase dari data uji yang diklasifikasikan ke kelas yang benar. *accuracy* dapat dinyatakan dalam persamaan berikut.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2.8)$$

Selain *accuracy*, nilai *precision* dan *recall* juga dapat digunakan untuk mengetahui kinerja model klasifikasi. *Precision* merupakan rasio prediksi benar positif (TP) dibandingkan dengan keseluruhan hasil yang diprediksi positif. Berdasarkan *confusion matrix* yang digunakan rumus dari presisi pada penelitian ini dinyatakan pada persamaan berikut.

$$Precision = \frac{TP}{TP+FP} \quad (2.9)$$

Sedangkan *recall* atau rasio TP adalah ukuran untuk berapa banyak TP yang diprediksi dari semua positif dalam kumpulan data, kadang juga disebut kepekaan (*sensitivity*). Rumus *recall* pada penelitian ini dapat dituliskan seperti pada persamaan berikut.

$$Recall = \frac{TP}{TP+FN} \quad (2.10)$$

F1 Score merupakan perbandingan rata-rata *precision* dan *recall*. Rumus *F1 score* dituliskan seperti pada persamaan berikut.

$$F_1Score = 2 \times \frac{precision \times recall}{precision + recall} \quad (2.11)$$

Akurasi memiliki tingkat nilai diagnose yaitu :

- a. Akurasi dengan nilai 0.90-1.00 dapat dikategorikan sebagai *excellent classification*
- b. Akurasi dengan nilai 0.80-0.90 dapat dikategorikan sebagai *good classification*
- c. Akurasi dengan nilai 0.70-0.80 dapat dikategorikan sebagai *fair classification*
- d. Akurasi dengan nilai 0.60-0.70 dapat dikategorikan sebagai *poor classification*
- e. Akurasi dengan nilai 0.50-0.0 dapat dikategorikan sebagai *failure*