

**KLASIFIKASI PENURUNAN KUALITAS TELUR AYAM RAS
BERDASARKAN WARNA KERABANG TELUR MENGGUNAKAN
ALGORITMA *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

SKRIPSI

Oleh:
NABILA MAULIDA
NIM. 17650034



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

**KLASIFIKASI PENURUNAN KUALITAS TELUR AYAM RAS
BERDASARKAN WARNA KERABANG TELUR MENGGUNAKAN
ALGORITMA *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

SKRIPSI

Oleh:
NABILA MAULIDA
NIM. 17650034

**Diajukan kepada:
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

HALAMAN PERSETUJUAN

**KLASIFIKASI PENURUNAN KUALITAS TELUR AYAM RAS
BERDASARKAN WARNA KERABANG TELUR MENGGUNAKAN
ALGORITMA *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

SKRIPSI

Oleh:
NABILA MAULIDA
NIM. 17650034

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal: 13 Desember 2022

Dosen Pembimbing I



Dr. M. Faisal, M.T
NIP. 19740510 200501 1 007

Dosen Pembimbing II



Dr. Yunifa Miftachul Arif, M.T
NIP. 19830616 201101 1 004

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Fachrul Kurniawan, M.MT, IPM
NIP. 19771020 200912 1 001

LEMBAR PENGESAHAN

KLASIFIKASI PENURUNAN KUALITAS TELUR AYAM RAS BERDASARKAN WARNA KERABANG TELUR MENGGUNAKAN ALGORITMA *CONVOLUTIONAL NEURAL NETWORK* (CNN)

SKRIPSI

Oleh:
NABILA MAULIDA
NIM. 17650034

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S. Kom)
Tanggal: 13 Desember 2022

Susunan Dewan Penguji

Ketua Penguji : Hani Nurhayati, M.T
NIP. 19780625 200801 2 006

Anggota Penguji I : Juniardi Nur Fadila, M.T
NIP. 19920605 201903 1 015

Anggota Penguji II : Dr. M. Faisal, M.T
NIP. 19740510 200501 1 007

Anggota Penguji III : Dr. Yunifa Miftachul Arif, M.T
NIP. 19830616 201101 1 004

()
()
()
()

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Fachrul Kurniawan, M.MT, IPM
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Nabila Maulida

NIM : 17650034

Program Studi : Teknik Informatika

Fakultas : Sains dan Teknologi

Judul Skripsi : Klasifikasi Penurunan Kualitas Telur Ayam Ras Berdasarkan
Warna Kerabang Menggunakan Algoritma *Convolutional Neural
Network (CNN)*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar – benar merupakan hasil karya saya sendiri, bukan merupakan pengambilan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila di kemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 16 Desember 2022
Yang membuat pernyataan,

The image shows an official red stamp of the institution, featuring the Garuda Pancasila emblem and the text 'MEXERA TEMPEL' and 'BF3AKX20541826'. To the right of the stamp is a handwritten signature in black ink.

Nabila Maulida
NIM. 17650034

HALAMAN MOTTO

لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

“Allah tidak akan membebani seseorang, melainkan sesuai dengan kesanggupannya”

HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Penulis persembahkan skripsi ini kepada keluarga penulis, terutama untuk kedua orang tua ayahanda Arif Rosyidin dan ibunda Dhikroh yang telah banyak memberikan pelajaran kehidupan kepada penulis melalui kerja keras, kesabaran, dan kesederhanaan beliau. Penulis tidak akan bisa seperti ini tanpa ridlo dari beliau. Teruntuk diri sendiri terimakasih banyak telah berjuang dan bertahan sampai akhir masa kuliah. Dan terakhir teruntuk mas fahmi terimakasih banyak atas saran dan semangat yang diberikannya kepada penulis agar mampu menyelesaikan skripsi ini dengan baik. Semoga kasih sayang Allah subhanahu wa ta'ala selalu menyertai.

KATA PENGANTAR

Assalamu`alaikum Warohmatullahi Wabarokatuh

Segala puja dan puji syukur penulis haturkan kepada Allah SWT atas Rahmat, Taufik dan Hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul **“Klasifikasi Penurunan Kualitas Telur Ayam Ras Berdasarkan Warna Kerabang Menggunakan Algoritma *Convolutional Neural Network* (CNN)”** sebagai syarat kelulusan dalam menyelesaikan masa – masa perkuliahan di Universitas Islam Negeri Maulana Malik Ibrahim Malang. Sholawat serta salam semoga tetap terlimpahkan kepada junjungan kita Nabi Muhammad SAW yang telah membimbing dan mewariskan Agama Islam sehingga sampai pada sekarang ini.

Keberhasilan penulis skripsi ini tidak lepas dari dorongan dan bimbingan dari berbagai pihak. Untuk itu dalam kesempatan ini penulis mengucapkan terima kasih yang sebesar – besar nya kepada:

1. Prof. Dr. H. M. Zainuddin, MA, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M. Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Fachrul Kurniawan, ST., M.MT., IPM selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Bapak Dr. M. Faisal, M.T, selaku dosen pembimbing 1 yang selalu meluangkan waktu dan bersedia memberikan bimbingan serta arahan kepada penulis sehingga dapat menyelesaikan skripsi ini.
5. Bapak Dr. Yunifa Miftachul Arif, M.T selaku dosen pembimbing 2 yang selalu meluangkan waktunya dalam membimbing penulis sehingga dapat menyelesaikan skripsi ini.
6. Ibu Hani NurHayati, M.T serta Bapak Juniardi Nur Fadila, M.T selaku Dosen Penguji yang telah memberikan kritik dan masukan untuk membangun kepada penulis selama proses penyelesaian skripsi ini.

7. Seluruh Dosen dan Jajaran Staff Program Studi Teknik Informatika yang telah mengajarkan ilmu yang bermanfaat kepada penulis.
8. Teman – teman Teknik Informatika Angkatan 2017 UNOCORE, khususnya Nurul Khafidoh, Winda Yulistiana, Layla Qomariah, Nur Ainayah Izzah yang telah menemani dan memberikan pengalaman serta *support* untuk menyelesaikan skripsi ini.
9. Pemilik NIM 16650063, terimakasih banyak telah menemani dengan sabar, memberi dukungan, kebaikan dan perhatian untuk penulis.
10. Semua pihak yang telah membantu dalam penyelesaian skripsi ini yang tidak dapat disebutkan satu persatu.

Penulis sangat menyadari sebagai mahasiswa yang jauh dari kata cerdas dan pintar ini, masih terdapat banyak kekurangan dalam penelitian sehingga masih jauh dari kata sempurna dan memuaskan selama penyusunan skripsi ini, oleh karena itu, penulis memohon maaf atas segala kekurangan yang terjadi selama proses perkuliahan dan penyusunan skripsi ini. Semoga tugas akhir ini selalu memberikan kontribusi serta bermanfaat bagi penulis khususnya bagi para pembaca.

Terakhir salam dari penulis, teruslah semangat, jangan takut gagal dan jangan mudah menyerah untuk mencapai kesuksesan. Karena awal dari kesuksesan adalah dari kegagalan.

Wassalamu'alaikum Warohmatullahi Wabarokatuh

Malang, 16 Desember 2022

Penulis

DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN TULISAN	Error! Bookmark not defined.
HALAMAN MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
ABSTRAK	xiv
ABSTRACT	xv
المخلص	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Pernyataan Masalah	6
1.3 Tujuan Penelitian	6
1.4 Manfaat Penelitian	7
1.5 Batasan Masalah	7
BAB II TINJAUAN PUSTAKA	8
2.1 Penelitian Terkait	8
2.2 <i>Image Classification</i> (Klasifikasi Gambar)	9
2.3 Telur Ayam Ras	10
2.3.1 Faktor Kualitas Telur	11
2.4 Citra Digital	13
2.4.1 Pengolahan Citra Digital	15
2.4.2 Citra RGB (<i>Red, Green, Blue</i>)	16
2.5 <i>Artificial Intellegence</i> (AI)	18
2.5.1 <i>Machine Learning</i>	19
2.5.2 Komponen <i>Machine Learning</i>	20
2.5.3 Tipe <i>Machine Learning</i>	21

2.6	<i>Deep Learning</i>	23
2.7	<i>Artificial Neural Network (ANN)</i>	24
2.7.1	<i>Komponen Neural Network</i>	26
2.7.2	<i>Arsitektur Neural Network</i>	28
2.8	<i>Convolutional Neural Network (CNN)</i>	30
2.8.1	<i>Feature Learning</i>	32
2.8.2	<i>Classification</i>	38
2.8.3	<i>Softmax Classifier</i>	40
2.8.4	<i>Cross Entropy Loss Function</i>	41
2.9	<i>Confussion Matriks</i>	41
2.10	<i>Python</i>	42
2.11	<i>Keras</i>	43
2.12	<i>Tensorflow</i>	44
2.13	<i>Google Colaboratory</i>	45
	BAB III ANALISIS DAN PERANCANGAN SISTEM	46
3.1	<i>Perancangan Sistem</i>	46
3.1.1	<i>Input Image</i>	47
3.1.2	<i>Preprocessing Image</i>	50
3.1.3	<i>Pemodelan Klasifikasi Menggunakan CNN</i>	52
3.2	<i>Rancangan Percobaan</i>	56
3.3	<i>Rancangan Pengujian</i>	58
3.4	<i>Pengukuran Akurasi</i>	58
	BAB IV HASIL DAN PEMBAHASAN	61
4.1	<i>Hasil Uji Coba</i>	61
4.1.1	<i>Hasil Proses Training</i>	61
4.1.2	<i>Hasil Proses Testing</i>	66
4.1.3	<i>Pengaruh Parameter terhadap Akurasi</i>	70
4.2	<i>Integrasi</i>	72
	BAB V KESIMPULAN DAN SARAN	78
5.1	<i>Kesimpulan</i>	78
5.2	<i>Saran</i>	79
	DAFTAR PUSTAKA	

DAFTAR GAMBAR

Gambar 2.1 Matriks Citra Digital (Putra, 2013).....	14
Gambar 2.2 Representasi Citra Digital 2 dimensi	15
Gambar 2.3 Representasi Warna RGB pada Citra	17
Gambar 2. 4 Bagian utama <i>Artificial Intelligence</i> (AI).....	18
Gambar 2.5 Jaringan Syaraf Manusia	25
Gambar 2.6 Struktur <i>Neural Network</i>	26
Gambar 2.7 Struktur <i>Single Layers Neural Networks</i>	29
Gambar 2.8 Struktur <i>Multiple Layer Neural Networks</i>	29
Gambar 2.9 <i>Competitive Layers</i>	30
Gambar 2.10 Arsitektur CNN.....	32
Gambar 2.11 Proses <i>Convolutional layer</i>	33
Gambar 2.12 <i>Stride</i> konvolusi.....	34
Gambar 2.13 Penambahan <i>Padding</i>	34
Gambar 2.14 Proses pembuatan <i>convolution layer</i> dari beberap <i>feature map</i>	35
Gambar 2.15 Grafik fungsi aktivasi <i>ReLU</i>	36
Gambar 2.16 Teknik Pooling	37
Gambar 2.17 Proses gambar dari <i>convolution layer</i> ke <i>pooling layer</i>	38
Gambar 2.18 Proses <i>flatten</i>	39
Gambar 2.19 Proses <i>fully connected layer</i>	40
Gambar 3.1 Desain Sistem	47
Gambar 3.2 Telur warna kerabang coklat tua.....	48
Gambar 3.3 Telur warna kerabang coklat	48
Gambar 3.4 Telur warna kerabang coklat muda.....	49
Gambar 3.5 <i>source code</i> proses <i>split</i> data	49
Gambar 3. 6 <i>Source code</i> Augmentasi data.....	51
Gambar 3.7 Hasil Proses Augmentasi data	51
Gambar 3.8 Artitektur CNN	52
Gambar 3.9 Model CNN	54
Gambar 3.10 Proses konvolusi	55
Gambar 3.11 <i>Training</i> dan <i>Testing</i>	56
Gambar 4.1 Hasil <i>Training</i> Gambar Ukuran 256 dan <i>Epoch</i> 50.....	62
Gambar 4.2 Hasil <i>Training</i> Gambar Ukuran 256 dan <i>Epoch</i> 75.....	62
Gambar 4.3 Hasil <i>Training</i> Gambar Ukuran 256 dan <i>Epoch</i> 100.....	62
Gambar 4.4 Hasil <i>Training</i> Gambar Ukuran 128 dan <i>Epoch</i> 50.....	63
Gambar 4.5 Hasil <i>Training</i> Gambar Ukuran 128 dan <i>Epoch</i> 75.....	63
Gambar 4.6 Hasil <i>Training</i> Gambar Ukuran 128 dan <i>Epoch</i> 100.....	64
Gambar 4.7 Hasil <i>Training</i> Gambar Ukuran 64 dan <i>Epoch</i> 50.....	64
Gambar 4.8 Hasil <i>Training</i> Gambar Ukuran 64 dan <i>Epoch</i> 75.....	64
Gambar 4.9 Hasil <i>Training</i> Gambar Ukuran 64 dan <i>Epoch</i> 100.....	65
Gambar 4. 10 Gambar grafik Hasil Perbandingan Parameter	70
Gambar 4.11 Hasil <i>Preprocessing</i> menggunakan teknik augmentasi	73
Gambar 4.12 Hasil Visualisasi <i>feature map</i>	74

DAFTAR TABEL

Tabel 3.1 Pembagian Dataset	50
Tabel 3. 2 <i>Confussion Matrix</i>	59
Tabel 4.1 Pengaruh Ukuran Gambar dan Besar Nilai <i>Epoch</i>	65
Tabel 4.2 Hasil uji coba klasifikasi telur kerabang coklat tua.....	66
Tabel 4.3 Hasil uji coba klasifikasi telur kerabang coklat.....	67
Tabel 4.4 Hasil uji coba klasifikasi telur kerabang coklat muda.....	68
Tabel 4.5 Hasil prediksi proses <i>testing</i>	69
Tabel 4.6 Pengaruh <i>epoch</i>	71
Tabel 4.7 Pengaruh ukuran citra.....	72
Tabel 4.8 Pengaruh Jumlah Data.....	72

ABSTRAK

Maulida, Nabila. 2022. **Klasifikasi Penurunan Kualitas Telur Ayam Ras Berdasarkan Warna Kerabang Telur Menggunakan Algoritma *Convolutional Neural Network* (CNN)**. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
Pembimbing: (1) Dr. M. Faisal, M.T, (2) Dr. Yunifa Miftachul Arif, M.T

Kata kunci: Telur Ayam Ras, *Deep Learning*, CNN, Klasifikasi Gambar

Telur ayam ras banyak dikenal masyarakat Indonesia karena harga nya yang cukup relatif murah, serta memiliki kandungan sumber protein dan gizi yang sangat bermanfaat bagi tubuh. Untuk mengetahui baik buruknya kualitas telur, dapat dilihat dari bagian luar (kerabang telur) dan bagian dalam telur (putih telur). Pada penelitian ini, penulis akan melakukan klasifikasi kualitas telur ayam ras berdasarkan citra telur dengan kategori warna kerabang telur yang berbeda menggunakan algoritma *Convolutional Neural Network* (CNN). Tujuan dari penelitian ini untuk mengetahui hasil tingkat akurasi penggunaan algoritma CNN dalam mengklasifikasi penurunan kualitas telur ayam ras. Proses pengambilan gambar telur ayam ras dilakukan menggunakan kamera *smartphone* dengan resolusi 8 *megapixel* dan diambil dari sisi atas telur dengan menggunakan *background* atau latar belakang warna putih sebagai alas telur. Citra kemungkinan objek tersebut *diinputkan* ke dalam CNN untuk dilakukan beberapa scenario uji coba dengan skala ukuran *input* (64x64, 128x128 dan 256x256) dengan jumlah *epoch* (50, 75 dan 100). Sehingga didapatkan hasil (97%, 97.67%, 97.67%), (98%, 98%, 94.16%) dan (91%, 93.39%, 98 Hal ini yang menunjukkan pengaruh ukuran citra dengan jumlah *epoch* sangat berpengaruh ketika dilakukan proses *training*.

ABSTRACT

Maulida, Nabila. 2022. **Classification of Declining Quality of Purebred Chicken Eggs Based on Eggshell Color Using the Convolutional Neural Network (CNN) Algorithm.** Undergraduate Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Supervisor: (1) Dr. M. Faisal, M.T, (2) Dr. Yunifa Miftachul Arif, M.T

Keywords: *Broiler Chicken Eggs, Deep Learning, CNN, Image Classification*

Purebred chicken eggs are widely known by the people of Indonesia because the price is relatively cheap, and contains a source of protein and nutrition which is very beneficial for the body. To find out whether egg quality is good or bad, it can be seen from the outside (egg shell) and the inside of the egg (egg white). In this study, the authors will classify the quality of purebred chicken eggs based on egg images with different eggshell color categories using the Convolutional Neural Network (CNN) algorithm. The purpose of this study was to determine the results of the accuracy of the use of the CNN algorithm in classifying the decline in the quality of broiler chicken eggs. The process of taking pictures of purebred chicken eggs is carried out using a smartphone camera with a resolution of 8 megapixels and taken from the top side of the egg using a white background or background as the egg base. Images of possible objects are input into CNN for several test scenarios with input size scales (64x64, 128x128 and 256x256) with the number of epochs (50, 75 and 100). So that the results (97%, 97.67%, 97.67%), (98%, 98%, 94.16%) and (91%, 93.39%, 98) show that the effect of image size on the number of epochs is very influential when the training process is carried out.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Berdasarkan laporan dari Menteri PPN, Andrinof Chaniago mengatakan “bahwa Negara Indonesia merupakan salah satu dari beberapa Negara di dunia yang mengalami permasalahan terhadap gizi buruk. Penyebab dari gizi buruk tersebut diantaranya yaitu kurangnya kualitas dari bahan pangan yang dikonsumsi sehingga, menyebabkan tidak seimbangnya energi dan kalori yang ada pada tubuh manusia. Oleh karena itu, memerhatikan kualitas pada bahan pangan serta mengkonsumsi gizi dengan cukup adalah hal yang sangat penting, seperti halnya dalam memilih daging, ikan, telur, susu, buah dan sayuran” (bappenas, 2015).

Telur merupakan salah satu produk peternakan unggas yang memiliki kandungan gizi yang lengkap dan kaya dengan sumber protein hewani. Secara umum, telur mempunyai tiga komponen pokok, yaitu kerabang atau kulit telur atau cangkang, putih telur dan kuning telur. Masing – masing dari komponen telur tersebut memiliki bobot yang berbeda, diantaranya untuk kerabang telur memiliki bobot 11%, putih telur 57% dan kuning telur memiliki bobot 32% (Yang et al., 2009).

Telur mudah mengalami penurunan pada kualitasnya, diantaranya disebabkan oleh adanya kontaminasi mikroba, kerusakan secara fisik serta penguapan gas ammonia, karbondioksida dan hydrogen sulfide dari dalam telur (Ibrahim et al., 2012). Kualitas telur dapat dilihat melalui dua bagian telur, yaitu bagian dalam telur yang terdiri dari kantong udara, putih telur dan kuning telur. Sedangkan kualitas telur bagian luar terdiri dari warna kerabang, bentuk dan

kebersihan telur (Thipakorn et al., 2017). Semakin lama telur disimpan, maka akan terjadi penguapan yang mempengaruhi bobot telur menjadi menyusut dan putih telur juga menjadi lebih encer. Selain dipengaruhi oleh lamanya dalam penyimpanan, terjadinya penguapan juga bisa dipengaruhi oleh kelembaban yang relative, suhu dan kualitas pada kerabang telur (Ibrahim et al., 2012).

Sebagian besar masyarakat pada umumnya mengkonsumsi telur ayam setiap hari. Adapun telur ayam yang sering dikonsumsi masyarakat pada umumnya adalah telur ayam ras, karena banyak beredar di pasaran dengan harga yang cukup relatif murah. Selain itu telur ayam mudah diolah dan tentunya memiliki kandungan sumber protein dan nutrisi penting yang sangat bermanfaat bagi tubuh. Namun, disisi lain telur ayam juga memiliki sifat yang mudah mengalami penurunan terhadap kualitasnya. Pada sebutir telur, kualitas telur yang akan menjadi acuan agar telur tersebut tetap terjaga kesegarannya sehingga layak untuk dikonsumsi. Oleh karena itu, sebelum mengkonsumsi telur ayam sebaiknya kita harus lebih teliti saat memilihnya, karena tidak semua telur ayam memiliki kualitas yang baik untuk dikonsumsi.

Pada dasarnya manusia mampu melihat kualitas telur ayam yang akan dikonsumsi dengan cara melihat kondisi luar telur seperti keutuhan cangkang atau kerabang pada telur tersebut. Selain itu, dapat dilakukan dengan memecahkan telur untuk melihat keadaan bagian dalam telur. Apabila putih dan kuning telur dalam keadaan mencair dan mengeluarkan aroma yang kurang sedap maka, telur tersebut sudah tidak layak untuk dikonsumsi. Akan tetapi, jika melakukan pemilihan kualitas telur dengan cara manual tersebut akan memakan waktu yang cukup lama. Oleh karena itu, penentuan kualitas telur sangat diperlukan. Kualitas telur

merupakan karakteristik dari telur yang menentukan kesegaran pada telur tersebut. Kualitas kerabang telur dapat dilihat berdasarkan kondisi kebersihan pada kerabang telur. Identifikasi kualitas telur berdasarkan kebersihan kerabang telur menggunakan jaringan syaraf tiruan menghasilkan tingkat akurasi sebesar 93,33% (Maimunah & Rokhman, 2018).

Untuk mengatasi masalah tersebut, terdapat penelitian sebelumnya yang dilakukan oleh Bacheramsyah, T. F., (2018) yang telah melakukan penelitian mengenai deteksi terhadap kualitas telur ayam berdasarkan kuning telur serta ketebalan putih telur untuk mendapatkan nilai *haugh unit* menggunakan metode *Discrete Cosine Transform* (DCT) dengan metode klasifikasi *K-Nearest Neighbor* yang menghasilkan nilai akurasi sebesar 83,6363%. Namun, pada penelitian ini proses pengambilan citra masih dilakukan secara manual dengan memecahkan telur terlebih dahulu untuk mengacu pada nilai *haugh unit*. Selanjutnya terdapat penelitian tentang klasifikasi kualitas telur yang dilakukan secara invasive dengan menggunakan metode KNN dan diimplementasikan pada perangkat android dengan tingkat akurasi 80% (IBRAHIM et al., 2022).

Selanjutnya terdapat penelitian tentang deteksi kebersihan kerabang telur ayam berdasarkan pengolahan citra digital. Penelitian tersebut menggunakan metode *thresholding* dengan menggunakan citra telur RGB sebagai inputan yang kemudian diubah menjadi citra *grayscale* sebagai outputnya (Maimunah & Rokhman, 2018). Berdasarkan penelitian tersebut dapat diambil kesimpulan bahwa metode pengolahan citra digital sangat mempengaruhi hasil klasifikasi telur berdasarkan kebersihan dari kerabang telur tersebut. Berdasarkan penelitian – penelitian sebelumnya, terdapat kekurangan pada proses pengukurannya yang

dilakukan secara *invasive*, yaitu dengan memecahkan telur untuk mengambil citra dari isi telur tersebut untuk diproses.

Penggunaan metode berbasis *Convolutional Neural Network* (CNN) belakangan ini telah mencapai keberhasilan yang signifikan dalam mengatasi beberapa masalah seperti di atas, terutama pada proses klasifikasi citra dengan resolusi tinggi (Arrofiqoh & Harintaka, 2018). *Convolutional Neural Network* (CNN) merupakan modifikasi dari metode *ann* yang memanfaatkan proses konvolusi dengan cara menggeser sebuah kernel konvolusi (filter) berukuran tertentu ke sebuah gambar karena pada dasarnya metode CNN ini dirancang khusus untuk dapat memproses data gambar dua dimensi. Tujuan proses konvolusi ini adalah untuk mendapatkan feature map dari gambar yang di inputkan. Sedangkan tujuan dari feature map sendiri agar dapat memahami apa saja fitur dari input yang terdeteksi atau yang dipertahankan dari gambar sehingga didapati suatu ciri khusus dari gambar yang diinputkan. (Kelleher, J. D., 2019).

Berdasarkan penjelasan di atas maka, pada penelitian ini akan mengimplementasikan algoritma *Convolutional Neural Network* (CNN) dalam pengklasifikasian penurunan kualitas telur ayam berdasarkan warna kerabang. Dalam penggunaan metode CNN ini diharapkan dapat memberikan hasil yang sesuai, sehingga dapat dimanfaatkan oleh masyarakat terutama distributor dan peternak telur ayam agar lebih mudah dalam melakukan pemilihan kualitas telur berdasarkan citra telur yang beredar di pasaran. Sebagaimana telah dijelaskan dalam Al-Qur'an Surat Al - Baqarah ayat 168 yang berbunyi:

يَا أَيُّهَا النَّاسُ كُلُوا مِمَّا فِي الْأَرْضِ حَلَالًا طَيِّبًا وَلَا تَتَّبِعُوا خُطُوَاتِ الشَّيْطَانِ إِنَّهُ لَكُمْ عَدُوٌّ مُّبِينٌ

Artinya: *"Wahai manusia! Makanlah dari (makanan) yang halal dan thayyib (baik) yang terdapat di bumi, dan janganlah kamu mengikuti langkah-langkah setan. Sungguh setan itu musuh yang nyata bagimu."* (QS. Al-Baqarah : 168)

Menurut penjelasan ayat diatas, pada dasarnya menjaga kesehatan tubuh dengan mengkonsumsi makanan dan minuman yang halal dan *thayyib* adalah sebuah kewajiban bagi setiap muslim, apalagi di era pandemi Covid-19 saat ini. Menurut Islam, makanan yang halal dan *thayyib* adalah makanan dan minuman yang dianjurkan untuk dikonsumsi. Halal menurut pemahaman fiqih adalah halal dari segi zatnya dan segi prosesnya. Sedangkan makanan yang disebut *thayyib* adalah apabila makanan tersebut dalam keadaan baik dan aman, sehingga jika dikonsumsi dapat memberi manfaat bagi tubuh manusia.

Dalam hal ini juga sesuai dengan hadits Rasulullah SAW yang diriwayatkan oleh Ath-Thabrani, Ahmad dan Ad-Daruguni dalam Shahihul Jami' yang berbunyi:

حَيْرُ النَّاسِ أَنْفَعُهُمْ لِلنَّاسِ

Artinya: *"Sebaik – baik manusia adalah yang paling bermanfaat bagi manusia lain."*

Berdasarkan hadits tersebut dijelaskan manusia merupakan makhluk sosial yang mempunyai hubungan timbal balik terhadap manusia satu sama lain. Sehingga dalam hal ini Rasulullah menganjurkan kepada umatnya untuk selalu berbuat baik dan bermanfaat bagi manusia satu sama lain. Dari hadits tersebut dapat disimpulkan untuk mengetahui tingkat derajat kemuliaan seseorang adalah dengan melihat sejauh mana manusia tersebut mampu memberikan kebaikan dan manfaat kepada manusia lain. Seperti halnya dalam penelitian ini tujuannya agar dapat dimanfaatkan oleh masyarakat terutama distributor dan peternak telur ayam agar

lebih mudah dalam melakukan pemilihan kualitas telur untuk menjaga kondisi tubuh.

Mempertahankan kondisi tubuh agar tetap sehat adalah merupakan bagian dari ibadah, baik dari sisi menjalankan sunnah agar tidak mudah tertular virus, serta kuat dalam menjalankan ibadah kepada Allah SWT. Selain itu, merupakan bagian dari wujud rasa syukur kita sebagai umat manusia kepada Allah yang telah memberikan karunia kenikmatan tubuh yang sehat, sehingga wajib kita jaga dan rawat dengan sebaik - baiknya.

1.2 Pernyataan Masalah

Berdasarkan penjelasan dari latar belakang diatas, maka terdapat pertanyaan yang akan diangkat dalam penelitian ini yaitu:

- a. Bagaimana implementasi penggunaan algoritma CNN dalam melakukan klasifikasi penurunan kualitas telur ayam ras berdasarkan warna kerabang?
- b. Seberapa besar tingkat akurasi penggunaan algoritma *Convolutional Neural Network* (CNN) dalam melakukan klasifikasi penurunan kualitas telur ayam ras berbasis citra?

1.3 Tujuan Penelitian

Tujuan pada penelitian ini adalah:

- a. Mengetahui hasil implementasi penggunaan algoritma CNN dalam melakukan klasifikasi penurunan kualitas telur ayam ras berdasarkan warna kerabang.

- b. Mengetahui hasil tingkat akurasi penggunaan algoritma *Convolutional Neural Network* (CNN) dalam melakukan klasifikasi penurunan kualitas telur ayam ras berbasis citra.

1.4 Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini adalah dapat membantu masyarakat terutama distributor dan peternak dalam proses pemeriksaan kualitas telur yang akan diedarkan di pasaran.

1.5 Batasan Masalah

Adapun batasan masalah dari penelitian ini yaitu:

- a. Pada penelitian ini dilakukan untuk mengklasifikasi penurunan kualitas telur ayam berdasarkan citra warna kerabang telur pada gambar.
- b. Data yang digunakan pada penelitian ini adalah berupa foto gambar telur ukuran $128 \times 128 \text{ pixels}$ dengan warna kerabang yang berbeda - beda.
- c. Pengambilan gambar dilakukan menggunakan kamera *smartphone* dengan resolusi 8 megapixel .
- d. Proses pengambilan gambar telur ayam diambil dari sisi atas telur dengan menggunakan background warna putih sebagai alas.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Terdapat beberapa metode yang telah digunakan untuk menghasilkan solusi yang tepat dalam mengembangkan *machine learning* dan algoritma komputer terutama untuk melakukan proses pengklasifikasian kualitas pada telur ayam. Seperti halnya Wijaya & Prayudi (2010) mengusulkan teknik segmentasi citra dan analisis regresi untuk melakukan klasifikasi bobot pada telur ayam ras. Teknik segmentasi citra ditetapkan untuk memisahkan objek dari latar berdasarkan intensitas warna. Sedangkan analisis regresi ditetapkan untuk mengukur bobot telur.

Penelitian selanjutnya terdapat teknik pengolahan citra dengan menggunakan metode jaringan syaraf tiruan untuk mengklasifikasi telur retak dan telur yang tidak retak. Terdapat beberapa tahap proses yang dilakukan yaitu, pengambilan citra, pengolahan citra analisis nilai citra dan training data citra. Berdasarkan uji coba yang dilakukan, pada penelitian ini memiliki tingkat akurasi 80% berdasarkan 40 data yang telah diuji (Bun et al., 2018).

Penggunaan metode *Convolutional Neural Network* (CNN) dalam beberapa tahun terakhir telah mendapatkan popularitas yang luar biasa karena memiliki tingkat akurasi yang tinggi jika dibandingkan dengan metode konvensional sebelumnya. Seperti penelitian dari Shimizu et al (2018) yang menerapkan CNN dengan arsitektur AlexNet dalam melakukan klasifikasi kualitas telur. Dalam metode ini menggunakan 5 *convolutional layer* dengan aktivasi ReLu yang diikuti dengan proses *max pooling*, kemudian dilanjutkan ke *fully connected layer*. Dalam

penelitian tersebut melakukan klasifikasi dari 91711 gambar telur yang diambil dari 4 sudut dan 1 sudut, setiap gambar berukuran 512440 piksel dan diberi label dari salah satu 6 kualitas yang sudah ditentukan. Dalam penelitian ini menunjukkan bahwa klasifikasi gambar dengan 4 sudut memiliki nilai akurasi lebih tinggi yaitu 93,3% daripada gambar 1 sudut. Hal ini karena gambar yang memiliki fitur multi sudut lebih efektif untuk melakukan klasifikasi pada benda 3 dimensi.

Ramadhan, S. H., (2020) mengusulkan pendekatan *deep learning* dan metode klasifikasi menggunakan *Convolutional Neural Network* (CNN) dengan arsitektur *Residual Network* (ResNet) untuk mengklasifikasi kualitas telur berdasarkan citra yang tidak dipecahkan dengan jarak pengambilan gambar sejauh 10 cm. Berdasarkan uji coba yang dilakukan, pada penelitian ini dapat dilakukan lebih mudah dan efektif dengan nilai akurasi sebesar 83,33%.

Berdasarkan penelitian sebelumnya, maka pada penelitian ini akan menerapkan metode *Convolutional Neural Network* (CNN) untuk melakukan klasifikasi penurunan kualitas telur ayam. Metode CNN dipilih pada penelitian ini karena memiliki kedalaman *layer* (lapisan) yang cukup banyak dan sangat mempengaruhi untuk mendapatkan hasil yang lebih baik serta mampu mendeteksi citra digital pada dataset yang mempunyai kategori *training* dan *testing*.

2.2 *Image Classification* (Klasifikasi Gambar)

Klasifikasi gambar adalah suatu teknik *komputer vision* untuk mengenali gambar dan memberi label atau kategori yang telah ditentukan sebelumnya (Clarifai, 2021) sehingga memungkinkan komputer dapat mengklasifikasi gambar ke dalam beberapa kategori. Klasifikasi mengacu pada jenis pelabelan dimana suatu gambar

diberi ciri - ciri tertentu dengan tujuan untuk lebih mudah dalam mengklasifikasikan gambar berdasarkan ciri - ciri yang mewakili setiap kategori yang berbeda.

Secara umum, teknik klasifikasi gambar dapat dibedakan menjadi dua yaitu *supervised classification* dan *unsupervised classification*. Pada *supervised classification* dimana algoritma dilatih dengan sekumpulan gambar dengan label yang sesuai. Selama pelatihan, algoritma mengekstrak fitur dari setiap matriks gambar sebagai data yang penting untuk diproses. Dengan demikian, hal ini dapat membantu algoritma dalam memprediksi tanda yang sesuai untuk gambar yang belum dilihatnya dengan bantuan informasi dari hasil ekstraksi data sampel berlabel. Sedangkan pada *unsupervised classification*, dimana algoritma hanya menggunakan data mentah sebagai pelatihan. *Convolutional Neural Network* (CNN) merupakan salah satu metode dalam *deep learning* yang digunakan untuk mengklasifikasikan data berlabel dengan menggunakan teknik *supervised learning*. Oleh karena itu, pada penelitian ini akan mengimplementasikan metode CNN dalam melakukan klasifikasi pada *image*.

2.3 Telur Ayam Ras

Telur ayam ras merupakan salah satu sumber pangan protein hewani yang paling dimintai oleh masyarakat. Hampir seluruh masyarakat dapat mengonsumsi telur ayam ras untuk memenuhi kebutuhan protein hewani. Hal ini dikarenakan telur mudah didapatkan dengan harga yang relative murah serta memiliki kandungan gizi yang lengkap (Jazil *et al.*, 2013). Menurut Badan Standarisasi Nasional, (2008) Telur ayam ras yang segar adalah telur yang tidak mengalami proses pendinginan dan tidak mengalami pengawetan serta tidak menunjukkan ciri

– ciri pertumbuhan embrio yang jelas, yang mana belum tercampurnya *yolk* dengan albumen, fisik pada telur masih utuh dan bersih.

Telur merupakan salah satu bahan makanan yang bernilai gizi tinggi karena dalam telur mengandung zat-zat makanan yang sangat dibutuhkan oleh tubuh manusia (Umar et al., 2000). Sebutir telur didapatkan gizi yang cukup sempurna karena mengandung zat-zat gizi yang lengkap dan mudah dicerna. Kandungan protein kuning telur yaitu sebanyak 16,5% dan pada putih telur sebanyak 10,9%, sedangkan kandungan lemak pada kuning telur mencapai 32% dan pada putih telur terdapat dalam jumlah yang sedikit. Telur juga mengandung berbagai vitamin dan mineral, termasuk vitamin A, riboflacin, asam folat, vitamin B6, vitamin B12, choline, besi, kalsium, fosfor dan potassium (Sudaryani, 2003).

2.3.1 Faktor Kualitas Telur

Kualitas telur merupakan karakteristik dari telur yang menentukan kesegaran pada telur itu sendiri. Telur mudah mengalami penurunan kualitas yang disebabkan oleh kontaminasi mikroba, penguapan air dan gas – gas seperti karbondioksida, ammonia, nitrogen dan hydrogen dari dalam telur serta disebabkan oleh kerusakan secara fisik pada telur (Jazil, 2013). Semakin lama telur disimpan, maka penguapan yang terjadi juga akan membuat bobot telur tersebut menyusut dan menyebabkan putih telur menjadi lebih encer. Selain dipengaruhi oleh penyimpanan yang lama, penguapan ini juga dapat dipengaruhi oleh suhu, kelembaban dan kualitas pada kerabang telur. Oleh karena itu, penentuan kualitas telur sangat diperlukan.

Sebelum menentukan kualitas pada telur, perlu diketahui pada dasarnya dalam melakukan penentuan kualitas pada telur meliputi penentuan kualitas secara eksterior dan penentuan kualitas secara interior. Kualitas telur secara eksterior diantaranya meliputi kualitas kerabang pada telur, sedangkan kualitas interior merupakan kualitas bagian dalam telur.

Kerabang telur merupakan bagian terluar yang membungkus isi telur yang berfungsi mengurangi kerusakan fisik maupun biologis serta kerabang ini dilengkapi dengan pori – pori kerabang yang berguna untuk proses pertukaran gas dari dalam dan luar kerabang telur. Kualitas kerabang telur dapat ditentukan oleh bentuk, kehalusan, ketebalan, keutuhan serta kebersihan pada kerabang tersebut. Dimana kerabang telur yang tipis relative berpori – pori lebih banyak dan besar sehingga mempercepat penurunan kualitas pada telur yang terjadi akibat penguapan. Tebal tipisnya pada kerabang telur dipengaruhi oleh strain ayam, pakan, umur induk, *stress* dan penyakit pada induk. Semakin tua umur induk, maka semakin tipis kerabang telur nyakarena ayam tidak mampu memproduksi kalsium yang cukup guna untuk memenuhi kebutuhan kalsium dalam pembentukan kerabang (Sakroni et.al., 2015).

Tekstur kerabang telur serta ketebalan pada kerabang dapat berkurang seiring memudarnya warna kerabang telur, namun pori – pori kerabang semakin bertambah. Terdapat kolerasi yang signifikan antar warna kerabang telur dengan kekuatan kerabang, bobot kerabang dan ketebalan kerabang. Sedangkan tidak ada kolerasi yang jelas antara warna kerabang dengan berat telur, berat albumen, eras kuning telur, warna kuning telur dan kandungan kalsium dalam albumen. Sehingga dapat disimpulkan bahwa beberapa karakteristik kualitas telur dapat dilihat dari

kekuatan pada kerabang, ketebalan kerabang dan berat kerabang yang dapat dinilai berdasarkan warna kerabang (Yang et.al., 2009).

Untuk warna kerabang telur dapat ditentukan berdasarkan intensitas dari warna coklat yang dibedakan menjadi 3 warna yaitu warna coklat tua (*dark brown eggshell*), coklat (*brown eggshell*) dan coklat muda (*light brown eggshell*). Waktu penyimpanan yang lama serta warna dari kerabang telur yang berbeda dapat mempengaruhi penyusutan berat telur, nilai *haugh* serta kedalaman rongga udara. Berdasarkan dari ketiga warna kerabang telur telur dapat menentukan penurunan kualitas pada telur, jika semakin muda warna coklat kerabang telur, maka semakin tinggi pula terjadinya penurunan kualitas pada telur. Dengan demikian, konsumen dianjurkan untuk memilih telur untuk dikonsumsi dengan kerabang yang berwarna coklat tua, dimana kerabang dengan warna coklat tua ini memiliki penurunan kualitas yang paling rendah selama penyimpanan (Jazil, 2013).

2.4 Citra Digital

Citra digital merupakan gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang mana kontinu menjadi gambar diskrit melalui proses *sampling* dan kuantisasi. Dimana *sampling* sendiri menyatakan besar kecil nya ukuran *pixel* pada suatu citra. Sedangkan kuantisasi menyatakan jumlah warna yang ada pada citra. Proses perubahan dari sebuah citra menjadi citra digital disebut digitasi. Digitasi adalah suatu proses mengubah sebuah gambar, teks bahkan suara dari benda yang dapat dilihat ke dalam data elektronik dan dapat disimpan atau diproses untuk keperluan yang lain. Dalam pengolahan citra lebih mengacu pada pemrosesan pada data dua dimensi, yang mempelajari tentang bagaimana teknik

pengolahan terhadap sebuah citra. Citra yang dimaksud adalah sebuah gambar (foto) dan video. Sedangkan digital disini yang dimaksud adalah bahwa pengolahan citra / gambar dilakukan secara digital menggunakan komputer (Sutoyo, 2009). Citra digital memiliki sebuah larik (*array*) yang berisi nilai – nilai kompleks yang dipresentasikan dengan deretan bit tertentu.

Dalam komputer, citra digital dibagi menjadi bentuk *grid* dan elemen piksel berbentuk matriks 2 dimensi. Dimana setiap piksel – piksel tersebut memiliki angka yang mempresentasikan *channel* warna. Kemudian angka pada setiap piksel tersebut disimpan secara berurutan oleh sebuah komputer. Sebuah citra digital mewakili sebuah matriks yang terdiri dari **M** sebagai kolom dan **N** sebagai baris, dimana perpotongan antara kolom dan baris disebut sebagai piksel yang merupakan elemen paling kecil dari sebuah citra. Piksel sendiri memiliki dua parameter, yaitu koordinat dan intensitas warna. Nilai yang terdapat pada koordinat (x,y) adalah $f(x,y)$, dimana x dan y merupakan koordinat spasial dan nilai $f(x, y)$ merupakan intensitas citra pada koordinat tersebut. Oleh karena itu, citra dapat dituliskan ke dalam sebuah matriks seperti dibawah:

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Gambar 2.1 Matriks Citra Digital (Putra, 2013)

Berdasarkan rumus di atas, maka suatu citra $f(x,y)$ dapat dituliskan ke dalam fungsi matematis seperti berikut:

$$0 \leq x \leq M-1$$

$$0 \leq y \leq N-1$$

$$0 \leq f(x,y) \leq G-1$$

Dimana:

M = Jumlah piksel baris pada *array* sebuah citra

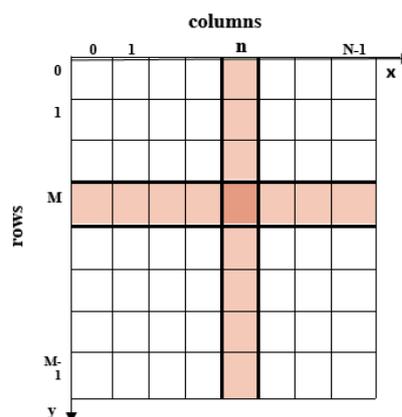
N = Jumlah piksel kolom pada *array* sebuah citra

G = Jumlah skala ke abu – abuan (*grayscale*)

Nilai di atas merupakan perpangkatan dari dua seperti persamaan berikut:

$$M = 2^m ; N = 2^n ; G = 2^k$$

Dimana nilai m , n , k merupakan bilangan positif. Integral $(0,G)$ disebut dengan (*grayscale*) . Besarnya nilai G tergantung pada proses digitalisasinya. Biasanya warna keabuan dengan nilai 0 menyatakan intensitas warna hitam sedangkan nilai 1 menyatakan intensitas warna putih. Untuk citra 8 bit, nilai G sama dengan $2^8 = 256$ warna (derajat keabu – abuan).



Gambar 2.2 Representasi Citra Digital 2 dimensi

2.4.1 Pengolahan Citra Digital

Pengolahan citra merupakan suatu proses pengolahan serta analisis dari sebuah citra yang melibatkan persepsi secara visual. Ciri dari proses ini adalah menggunakan data *inputan* berupa citra. Pengolahan citra melakukan berbagai teknik dalam memanipulasi atau memodifikasi citra dengan berbagai cara. Tujuan

pengolahan citra adalah memperbaiki kualitas dari suatu citra sehingga mampu diinterpretasi dengan lebih mudah oleh manusia atau sebuah komputer.

Foto adalah contoh gambar yang memiliki dua dimensi yang mudah diolah. Setiap foto dalam bentuk citra digital bisa diolah menggunakan perangkat lunak tertentu. Misalnya apabila hasil pengambilan kamera pada gambar terlihat agak gelap, citra tersebut dapat diolah menjadi lebih terang. Proses gambaran tersebut menunjukkan cara sederhana yang dapat dilakukan dalam pengolahan citra digital. Pada umumnya pengolahan citra mempunyai beberapa tujuan diantaranya:

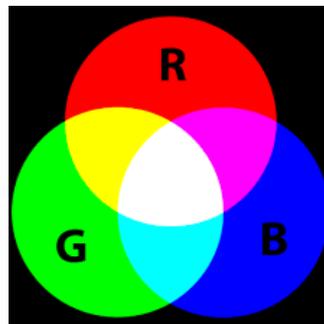
- a. Memodifikasi sebuah citra guna untuk meningkatkan kualitas serta menekankan sejumlah informasi yang terdapat pada citra tersebut
- b. Membagi bagian citra yang ingin dihilangkan atau digabungkan dengan bagian citra lain
- c. Mampu mengklasifikasikan, mencocokkan serta mengukur bagian – bagian tertentu dari sebuah citra
- d. Dapat memisahkan objek dari latarbelakangnya
- e. Menghilangkan *noise* atau bitnik – bitnik yang menodai pada gambar.

Pengolahan citra digital ini berhubungan dengan pengenalan pola (*pattern recognition*), yang bertujuan untuk mengenali suatu objek dengan cara melakukan ekstraksi pada informasi penting yang terdapat dalam suatu citra.

2.4.2 Citra RGB (*Red, Green, Blue*)

Teknologi dasar dalam menampilkan warna pada citra digital adalah mengkombinasikan dari tiga warna dasar yaitu, merah, hijau dan biru (*Red, Green, Blue*). *RGB* merupakan suatu model warna yang terdiri dari merah, hijau dan biru

yang mana ketiga warna tersebut digabungkan untuk membentuk suatu susunan warna yang luas. Dimana setiap warna tersebut dapat diberi rentang nilai antara 0 – 256. Pilihan skala 256 ini didasarkan pada 8 digit bilangan biner yang digunakan oleh mesin komputer. Dengan cara ini akan diperoleh warna campuran sebanyak $256 \times 256 \times 256 = 16777216$ jenis warna. Sebuah jenis warna dapat diilustrasikan sebagai sebuah vektor di ruang 3 dimensi yang biasanya dipakai dalam matematika, koordinatnya dinyatakan dalam bentuk tiga bilangan, diantaranya komponen-x, komponen-y dan komponen-z. Misalnya sebuah vektor dituliskan sebagai $r = (x,y,z)$. Untuk warna, komponen-komponen tersebut digantikan oleh komponen R(ed), G(reen), B(lue). Jadi, sebuah jenis warna dapat dituliskan sebagai berikut: warna = RGB (30, 75, 255), warna putih = RGB (255,255,255), sedangkan untuk warna hitam= RGB (0,0,0). Berikut bentuk representasi warna dari sebuah citra digital dapat dilihat pada gambar di bawah.



Gambar 2.3 Representasi Warna RGB pada Citra

Misalnya terdapat gambar dengan ukuran 100 x 100 piksel dengan *color encoding* 24 bit dengan R = 8 bit, G = 8 bit dan B = 8 bit, maka *color encoding* mampu mewakili 0 ... 16.777.215 (mewakili 16 juta warna) dengan membutuhkan ruang *disk* sebesar $100 \times 100 \times 3 \text{ bit} = 20.000 \text{ bit} = 20 \text{ KB}$ atau $100 \times 100 \times 24 \text{ bit} = 240.000 \text{ bit}$.

2.5 *Artificial Intelligence* (AI)

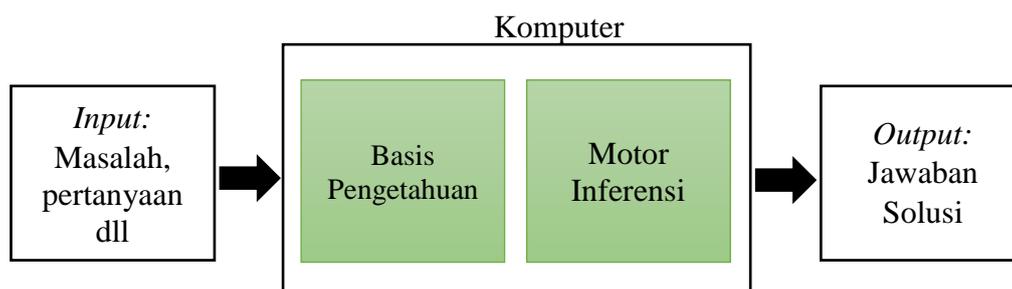
Artificial Intelligence (AI) merupakan bagian dari ilmu komputer yang mempelajari bagaimana suatu mesin mampu melakukan pekerjaan sebagaimana yang dilakukan oleh manusia, bahkan bisa lebih baik. *Artificial Intelligence* (AI) bertujuan untuk mengetahui atau memodelkan suatu proses cara berfikir manusia serta mendesain suatu mesin sehingga dapat menirukan suatu perilaku manusia (Arif & Nurhayati, 2022). Dalam pembuatan aplikasi kecerdasan buatan terdapat dua hal yang menjadi bagian penting yang dibutuhkan yaitu:

1. *Knowledge Base* (Basis pengetahuan)

Pada bagian ini berisi tentang fakta – fakt, teori, pemikiran serta hubungan antara satu dengan yang lainnya.

2. *Inference Engine* (Motor inferensi)

Berbeda dengan *Knowledge Base*, pada bagian *Inference Engine* (Motor inferensi) ini merupakan suatu kemampuan dalam menarik kesimpulan berdasarkan pengalaman.



Gambar 2. 4 Bagian utama *Artificial Intelligence* (AI)

Artificial Intelligence (AI) dibuat berdasarkan sistem yang mempunyai keahlian seperti halnya manusia pada domain tertentu yang disebut sebagai *soft computing*. *Soft computing* merupakan inovasi baru dalam membangun sistem cerdas yang mampu beradaptasi dan mampu bekerja lebih baik jika terjadinya

perubahan lingkungan. Selain itu, *soft computing* juga mengeksploitasi adanya toleransi terhadap ketidakpastian, kebenaran dan ketidaktepatan sehingga dapat diselesaikan dan dikendalikan dengan mudah dengan tujuan agar sesuai dengan realita. Metodologi yang sering digunakan dalam *soft computing* salah satunya adalah jaringan syaraf. Jaringan syaraf tiruan (*Artificial Neural Network / ANN*). Metodologi lain yang juga digunakan adalah sistem *fuzzy* (mengakomodasi ketepatan), *Evolutionary Computing* (optimasi) dan *Probabilistic Reasoning* (mengakomodasi ketidakpastian).

Menurut Rich dan Knight (1991), *Artificial Intelligence* (AI) adalah sebuah studi yang mempelajari tentang bagaimana membuat sebuah komputer melakukan hal – hal yang pada saat ini dapat dilakukan oleh manusia. Pada awal diciptakannya sebuah komputer, fungsi komputer hanya sebagai alat perhitungan saja. Namun, seiring dengan perkembangan zaman, peran komputer semakin mendominasi terhadap kehidupan manusia. Dimana komputer tidak hanya dijadikan sebagai alat hitung saja, namun diharapkan komputer agar mampu mengerjakan segala sesuatu yang mampu dikerjakan oleh manusia.

2.5.1 *Machine Learning*

Machine learning merupakan sekumpulan teknik yang bertujuan untuk menangani sekaligus memprediksi sekumpulan data dengan cara mempresentasikan data – data tersebut menggunakan algoritma pembelajaran. Dengan adanya *machine learning* ini, komputer mampu melakukan pembelajaran secara mandiri dalam mempresentasikan data – data yang telah diberikan (Danukusumo, 2017).

Machine learning dapat didefinisikan sebagai metode komputasi berdasarkan pengalaman untuk meningkatkan performa atau membuat suatu prediksi yang akurat. Pengalaman disini dimaksud dengan informasi sebelumnya yang sudah tersedia dan dapat dijadikan sebagai data pembelajaran.

Populernya istilah *big data*, teknik *machine learning* ini menjaadi kunci dalam mengatasi berbagai permasalahan tersebut, seperti:

- Biomedika untuk pengenalan sidik jari, mendeteksi penyakit tumor, penyesuaian DNA serta berbagai bidang lainnya
- Sistem keuangan untuk *credit scoring*, *trading* dan berbagai maslaah yang berhubungan dengan akuntansi
- *Image processing* untuk mendeteksi gerakan, pengenalan wajah atau objek.

2.5.2 Komponen *Machine Learning*

Machine learning terdiri dari beberapa komponen, diantaranya:

1. Pembelajaran

Pada bagian pembelajaran, algoritma dijadikan sebagai dasar untuk melakukan proses pembelajaran pada mesin. Algoritma dipilih berdasarkan masalah yang ingin diselesaikan, karena tidak semua algoritma cocok untuk menyelesaikan suatu masalah.

2. Kumpulan data

Kumpulan data atau *dataset* merupakan kumpulan dari beberapa data mentah yang dijadikan sebagai pelatihan untuk dipelajari. Memiliki data sampel dengan jumlah yang cukup sangat penting dalam proses

pembelajaran agar dapat memahami struktur sebuah masalah. Kumpulan data ini ada yang berlabel maupun tidak berlabel.

3. Representasi

Representasi merupakan suatu proses bagaimana menggambarkan data – data berdasarkan pada fitur yang dipilih, sehingga dapat digunakan untuk proses pembelajaran. Misalnya, untuk mengklasifikasikan sebuah gambar tulisan tangan, maka gambar tersebut akan dijaadikan sebagai nilai *array*. Setiap sel akan berisi nilai dalam warna setiap piksel. Untuk mendapatkan hasil yang baik, maka dapat dilakukan dengan pemilihan representasi yang baik pula.

4. Tujuan atau sasaran

Pada bagian ini menggambarkan suatu alasan untuk mempelajari dari data – data dalam menyelesaikan sebuah masalah. Tujuan ini membantu untuk menentukan bagaimana dan apa yang harus digunakan dalam melakukan suatu proses pembelajaran serta representasi apa yang akan digunakan.

5. Target

Target merepresentasikan apa saja yang telah dipelajari, serta merupakan hasil akhir yang didapatkan. Target sendiri dapat berupa klasifikasi data yang tidak memiliki label, representasi data yang sesuai dengan pola, memprediksi masa depan serta dapat berupa respon terhadap suatu strategi.

2.5.3 Tipe *Machine Learning*

Dalam proses pembelajaran *machine learning*, terdapat 2 tipe berdasarkan cara pembelajarannya, diantaranya:

a. *Supervised Learning*

Proses pembelajaran pada tipe ini, algoritma yang digunakan seolah – olah dilatih terlebih dahulu sebelum melakukan suatu prediksi ataupun klasifikasi suatu objek, sehingga mencapai target yang sesuai dengan yang ditetapkan oleh *user*. Bisaanya algoritma akan dilath dengan beberapa macam data latih yang berbeda agar mencapai target tertentu. Ketika algoritma sudah mencapai target, maka algoritma tersebut sudah siap untuk di uji.

Misalnya untuk mengklasifikasi jenis – jenis bunga. Sebelum melakukan klasifikasi, maka yang perlu disiapkan adalah data latih berupa beberapa foto bunga dengan jenis yang berbeda – beda. Selanjutnya data latih tersebut dilatih menggunakan algoritma *neural network* dengan target yang berbeda sesuai dengan yang ditentukan oleh *user*.

Supervised learning menggunakan 2 teknik dalam membangun sebuah model, yaitu model klasifikasi dan model regresi. Terdapat beberapa algoritma yang termasuk dalam *supervised learning*, yaitu *neural network*, *decision tree*, analisis deret waktu, *naïve bayes classifier* dan beberapa algoritma lainnya.

b. *Unsupervised Learning*

Berbeda dengan teknik *supervised*, pada teknik *unsupervised* ini tidak menggunakan data latih saat melatih sebuah algoritma. Pendekatan yang digunakan teknik ini adalah melakukan pembelajaran dengan pola tersembunyi yang terkandung pada data tersebut. Setelah pola pada data tersebut terbentuk, selanjutnya algoritma akan mengelompokkan data sesuai dengan ciri – ciri data lain yang sejenis.

Berdasarkan model matematikanya, teknik *unsupervised* ini tidak mempunyai target variabel. Pemodelan yang paling sering digunakan oleh teknik ini adalah *clustering model*. Dimana model ini digunakan untuk menganalisis suatu data untuk menemukan pola tersembunyi. Algoritma yang bisaanya digunakan untuk teknik ini diantaranya; *k-means*, *hierarchial clustering*, *fuzzy* dan algoritma yang lainnya.

2.6 *Deep Learning*

Deep learning merupakan salah satu bidang *machine learning* yang memanfaatkan jaringan saraf tiruan (JST) dalam mengimplementasikan suatu permasalahan dengan dataset yang besar (Arif et al., 2020). Teknik *deep learning* dapat memberikan arsitektur yang sangat kuat untuk *supervised learning*. Dengan menambahkan *layer* lebih banyak lagi, sehingga model pembelajaran tersebut mampu mewakili data citra berlabel dengan lebih baik lagi. Pada *machine learning* memiliki teknik dalam menggunakan ekstraksi sebuah fitur dari data pelatihan dan algoritma pembelajaran khusus untuk mengklasifikasikan citra maupun untuk pengenalan suara. Namun, metode ini juga masih memiliki beberapa kekurangan baik dalam hal kecepatan dan tingkat akurasi.

Algoritma yang digunakan pada *feature engineering* dapat menentukan pola umum yang penting untuk membedakan antara kelas dalam *deep learning*. Metode *Convolutional Neural Network* (CNN) sangatlah bagus dalam menemukan fitur yang baik pada citra ke *layer* berikutnya untuk membentuk hipotesis nonlinier yang mampu meningkatkan kekompleksian sebuah model. Suatu model yang kompleks tentunya akan membutuhkan waktu *training* yang lama, sehingga di dunia *deep learning* penggunaan CPU sudah sangatlah umum (Danukusumo, 2017). Pada

proses ini, sebuah komputer seolah dilatih dengan memanfaatkan dataset berlabel dengan jumlah yang sangat besar untuk mengubah nilai piksel pada suatu gambar menjadi *feature vector* (representasi internal) yang dimanfaatkan untuk melakukan deteksi atau mengklasifikasikan pola pada suatu *input* (LeCun et al, 2015).

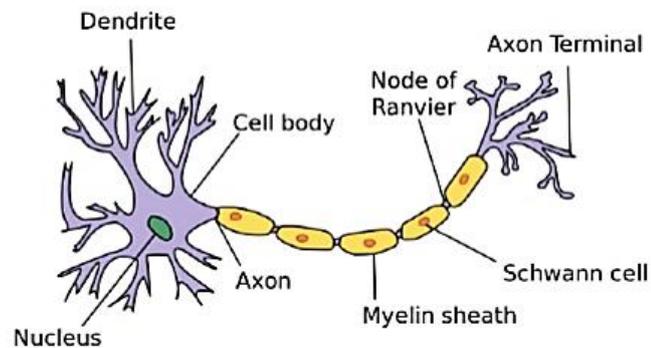
Metode *deep learning* adalah suatu teknik pembelajaran dengan banyak tingkat representasi yang memiliki beberapa lapisan. Lapisan yang ada pada *deep learning* dibagi menjadi tiga bagian diantaranya, *input layer*, *hidden layer*, dan *output layer*. Pada bagian *hidden layer* dibuat menjadi beberapa *layer* (lapisan), karena semakin banyak *layer* yang dihasilkan, maka semakin kecil pula nilai *error* yang dikeluarkan (Goodfellow et al, 2016).

Salah satu manfaat dari penggunaan *Deep Learning* adalah *image processing* (pengolahan citra digital). Pengolahan citra digital dapat dimanfaatkan manusia sebagai pengenalan objek atau dapat mengklasifikasikan objek secara efisien. Dalam *deep learning*, salah satu algoritma yang memanfaatkan *deep neural network* adalah *Convolutional Neural Network* (CNN). Dalam melakukan pemrosesan data pada gambar, metode ini menggabungkan antara algoritma konvolusi dengan *neural network*. Metode CNN ini sangat tepat dalam menentukan fitur yang baik pada gambar ke lapisan selanjutnya untuk meningkatkan suatu model yang kompleks.

2.7 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) merupakan suatu model komputasi parallel yang meniru fungsi dari sistem jaringan syaraf biologi otak manusia. Dimana dalam otak biologi manusia terdiri dari milyaran neuron yang saling berhubungan yang disebut dengan *Synapses*. Komponen neuron terdiri dari satu inti

sel yang akan melakukan pemrosesan suatu informasi, satu *axson* (akson) dan minimal satu *dendrit*. Informasi yang masuk akan diterima oleh *dendrit*. Selain itu *dendrit* juga menyertasi akson sebagai *output* dari suatu pemrosesan informasi.



Gambar 2.5 Jaringan Syaraf Manusia

Cara kerja dari sistem syaraf di atas adalah bermula pada sinyal yang masuk melalui *dendrit* kemudian menuju *cell body*. Selanjutnya syaraf akan di proses di dalam *cell body* berdasarkan dengan fungsi tertentu. Jika sinyal hasil proses melebihi nilai ambang batas (*reshold*) tertentu, maka sinyal tersebut akan membangkitkan neuron untuk meneruskan sinyal tersebut. Sedangkan jika hasil proses di bawah nilai ambang batas, maka sinyal tersebut akan dihalangi (*inhibited*). Kemudian sinyal yang diteruskan akan menuju ke akson dan selanjutnya menuju ke neuron lainnya melalui *synapse*.

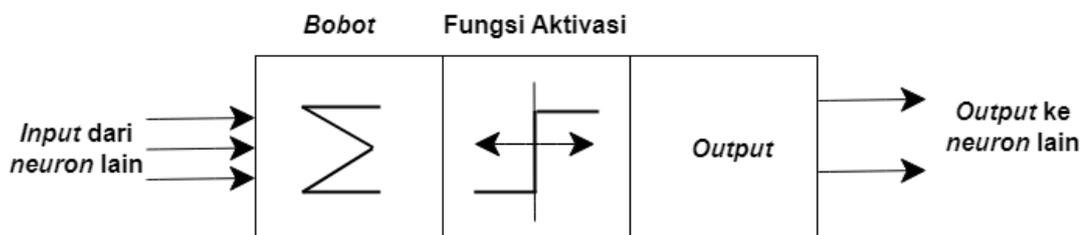
ANN merupakan sistem adaptif yang dapat mengubah strukturnya untuk memecahkan suatu masalah berdasarkan informasi internal maupun eksternal. ANN bersifat fleksibel terhadap inputan dan menghasilkan *output* respon secara konsisten. ANN telah banyak digunakan dalam area yang luas (Pham, 1994). Penerapan ANN dapat digunakan dalam mengidentifikasi beberapa aplikasi yaitu:

1. Pengenalan pola (klasifikasi, diagnosis dan analisis diskriminan)

2. Prediksi / estimasi (aproksimasi fungsi, peramalan)
3. Klustering (pengelompokan tanpa adanya pengetahuan sebelumnya).

2.7.1 Komponen *Neural Network*

Neural network mempunyai beberapa tipe yang berbeda – beda, akan tetapi hampir semua komponen yang dimiliki sama. Seperti halnya dengan jaringan syaraf pada otak manusia, *neural network* juga memiliki beberapa neuron unit yang saling terhubung. Masing – masing dari neuron tersebut akan melakukan transformasi informasi yang diterima melalui sambungan keduanya menuju neuron yang lain. Hubungan ini biasanya disebut dengan bobot (*weight*). Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tertentu. Berikut struktur neuron pada *neural network*:



Gambar 2.6 Struktur *Neural Network*

Berdasarkan gambar 2.6 menunjukkan bahwa struktur yang dimiliki oleh *neural network*. Pada struktur di atas terdapat komponen – komponen sebagai berikut:

1. *Input* terdiri dari variabel *independent* ($X_1, X_2, X_3, \dots, X_n$) yang merupakan sebuah sinyal yang masuk ke syaraf.
2. Bobot (*weight*) terdiri dari beberapa bobot ($W_1, W_2, W_3, \dots, W_n$) yang berhubungan dengan masing – masing node.

3. *Threshold* merupakan nilai ambang batas internal dari node. Besar kecilnya nilai ini akan mempengaruhi aktivitas dari *output* node *y*.

4. Fungsi aktivasi merupakan suatu operasi matematika yang dikenal pada sinyal *output* *y*.

Cara kerja struktur *neural network* di atas tidak jauh beda dengan struktur jaringan syaraf pada otak manusia. Informasi (*input*) akan dikirim dengan bobot tertentu. *Input* tersebut kemudian diproses oleh suatu fungsi perambatan yang nantinya akan menjumlahkan nilai – nilai semua bobot yang datang. Hasil dari pejumlahan ini kemudian akan dibandingkan dengan nilai ambang batas tertentu melalui fungsi aktivasi setiap neuron. Jika *input* tersebut melewati suatu nilai ambang batas tertentu, maka neuron tersebut akan diaktifkan. Jika sebaliknya, maka *neuron* tersebut tidak diaktifkan. Apabila neuron tersebut diaktifkan, selanjutnya neuron akan mengirimkan *output* melalui bobot – bobot *outputnya* ke semua neuron yang terhubung, begitu seterusnya.

Pada neuron *layer*, penempatan *neuron* akan dikelompokkan dalam *neuron layer*, kemudian *neuron* pada satu lapisan akan di hubungkan dengan *layer* sebelumnya dan setelahnya, kecuali pada *input layer* dan *output layer*. Informasi yang dibawa ketika *input* awal, akan disalurkan dari lapisan *input* ke lapisan *output*. Lapisan ini disebut dengan *hidden layer* (lapisan tersembunyi). Paa umumnya setiap neuron terletak pada lapisan yang sama dengan memiliki keadaan yang sama. Sehingga pada setiap lapisan, setiap neuron akan memiliki fungsi aktivasi yang sama, dengan syarat koneksi antara lapisan dengan neuron harus selalu terhubung.

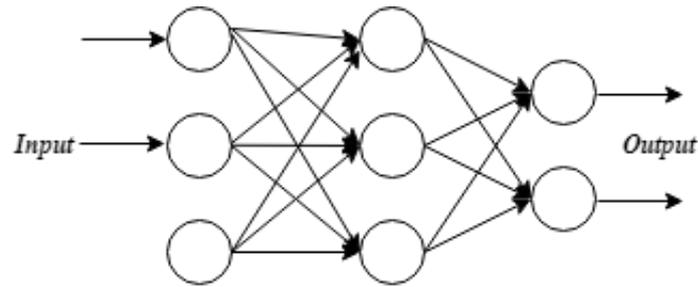
Dalam menentukan kelakuan suatu neuron terdapat faktor penting yaitu terletak pada pola bobot dan fungsi aktivasi nya.

2.7.2 *Arsitektur Neural Network*

Pada *neural network*, *neuron – neuron* yang ada pada *layer* yang sama mempunyai keadaan yang sama. Terdapat faktor penting dalam menentukan sifat pada suatu neuron yaitu dengan bobot dan penggunaan fungsi aktivasi dari neuron tersebut. Dimana setiap lapisan pada neuron memiliki fungsi aktivasi yang sama. Arsitektur yang dapat dibentuk oleh ANN ada bermacam – macam, dari yang paling sederhana yaitu hanya memiliki satu neuron (*single neuron*) sampai yang paling rumit menjadi multi *neuron (multiple neuron)* dalam satu *layer*, sampai jaringan *multiple neuron* dalam *multiple layers*. Beberapa jaringan tersebut mempunyai kemampuan yang berbeda – beda. Dimana semakin rumit suatu jaringan, maka permasalahan yang dapat diselesaikan akan menjadi lebih luas. Namun, terdapat kelemahannya juga yaitu kerumitan tersebut akan menimbulkan permasalahan tersendiri pada kebutuhan proses *training* dan proses *testing* yang akan memerlukan waktu yang lebih lama lagi. Menurut Hemawan (2006), Arsitektur neural network dibagi berdasarkan jumlah lapisannya, diantaranya:

1. *Single Layer Neural Network*

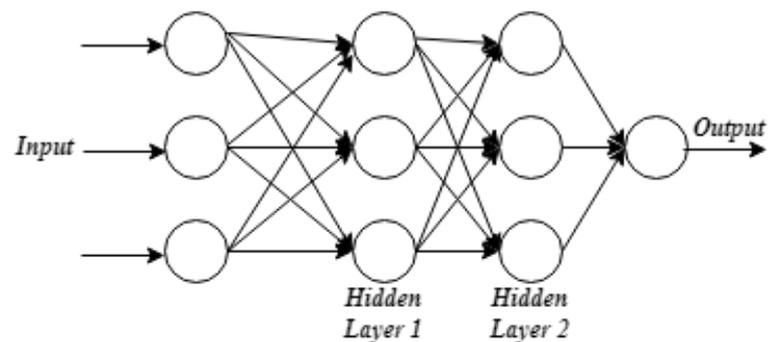
Pada jaringan ini terdiri dari 1 lapisan *input* dan 1 lapisan *output*. Setiap neuron yang terdapat di dalam lapisan *input* akan selalu terhubung dengan setiap neuron yang terdapat pada lapisan *output*. Pada jaringan ini hanya menerima *input*, kemudian secara langsung akan diolah menjadi *output* tanpa harus melewati lapisan tersembunyi. Berikut struktur *single layer neural network*.



Gambar 2.7 Struktur *Single Layers Neural Networks*

2. Multiple Layer Neural Network

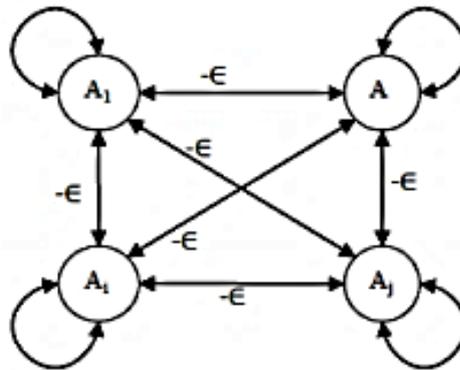
Pada jaringan ini memiliki ciri khas tertentu yaitu mempunyai 3 jenis lapisan diantaranya *input layer*, *output layer* dan *hidden layer*. Jaringan dengan 3 lapisan ini mampu menyelesaikan permasalahan yang lebih kompleks dibandingkan dengan jaringan lapisan tunggal. Akan tetapi, proses *training* pada lapisan ini akan membutuhkan waktu yang cenderung lama. Berikut gambar struktur *multiple layer neural network*.



Gambar 2.8 Struktur *Multiple Layer Neural Networks*

3. Competitive Layer

Pada jaringan ini, sekumpulan *neuron* akan bersaing untuk mendapatkan posisi menjadi aktif. Contoh algoritma yang menggunakan jaringan ini adalah LVQ. Berikut gambar struktur *competitive layers* seperti di bawah.



Gambar 2.9 Competitive Layers

2.8 Convolutional Neural Network (CNN)

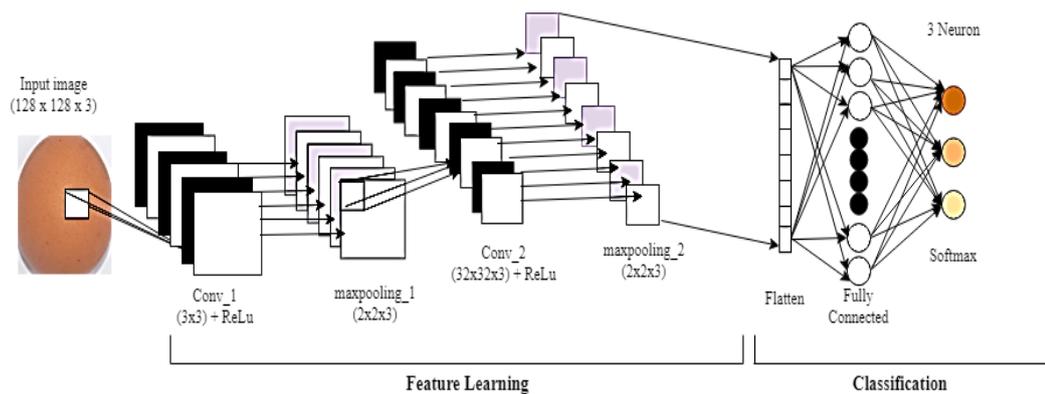
Convolutional Neural Network (CNN) adalah sebuah algoritma pembelajaran mendalam (*deep learning*) yang dapat melatih kumpulan data besar dengan jutaan parameter dan menggunakan sebuah bentuk gambar 2D sebagai inputan, kemudian gambar tersebut digabungkan dengan filter untuk menghasilkan sebuah output yang diinginkan. CNN merupakan jenis *neural network* yang mendominasi dari berbagai tugas komputer dan telah menarik perhatian dari berbagai bidang. CNN terdiri dari tiga lapisan, yaitu *convolutional layer*, *pooling layer* dan *fully connected layer*. Pada *convolutional layer* dan *pooling layer* akan melakukan proses ekstraksi fitur, sedangkan pada *fully connected layer* akan melakukan pemetaan fitur yang telah diekstraksi untuk dijadikan sebagai input.

Convolutional layer merupakan suatu perkalian antara pixel pada *input* dengan filter berukuran tertentu, sehingga menghasilkan sebuah *output* berupa *feature map* yang berisi fitur – fitur yang mirip dengan filter. Dengan filter maka proses pengolahan gambar akan semakin cepat karena data pixel yang diolah semakin sedikit. Namun tidak perlu khawatir, tujuan utama dari filter ini adalah mencari atau mendeteksi fitur – fitur tertentu yang bersifat integral. Seperti halnya

dengan *Convolutional layer*, pada *pooling layer* akan mengurangi ukuran spasial dari *feature map* yang bertujuan untuk mengurangi daya komputasi yang diperlukan dalam memproses data melalui pengurangan dimensi. *Fully connected layer* merupakan suatu lapisan dari *convolutional neural network* yang dibangun dengan *feature map* hasil output dari konvolusi sebagai inputan pada layer ini.

CNN merupakan modifikasi dari metode ann yang memanfaatkan proses konvolusi dengan cara menggeser sebuah kernel konvolusi (filter) berukuran tertentu ke sebuah gambar, karena pada dasarnya metode CNN ini dirancang khusus untuk dapat memproses data gambar dua dimensi. Tujuan proses konvolusi ini adalah untuk mendapatkan *feature map* dari gambar yang di inputkan. Sedangkan tujuan dari *feature map* sendiri agar dapat memahami apa saja fitur dari input yang terdeteksi atau yang dipertahankan dari gambar sehingga didapati suatu ciri khusus dari gambar yang diinputkan.

Secara umum, metode *Convolutional Neural Network* (CNN) tidak jauh beda dengan *neural network* yang bisaanya. CNN terdiri dari neuron yang memiliki *weight* (bobot), bisa dan *activation function*. Cara kerja CNN sendiri adalah memanfaatkan proses konvolusi dengan menggeser sebuah kernel konvolusi (filter) berukuran tertentu ke sebuah gambar. Dengan demikian, komputer mendapatkan informasi baru mengenai ciri - ciri dan identitas dari gambar yang dijadikan sebagai inputan. Berikut tahapan pada algoritma CNN dalam memproses suatu *input* gambar.



Gambar 2.10 Arsitektur CNN

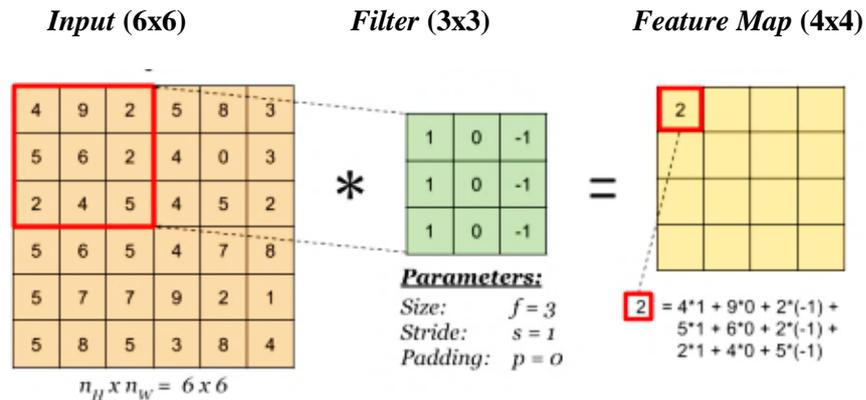
Berdasarkan gambar di atas, maka tahapan arsitektur CNN dibagi menjadi dua komponen utama yaitu, *Feature Learning* yang terdiri dari 2 layer yaitu *convolutional layer* dan *pooling layer* dan *Classification*.

2.8.1 *Feature Learning*

Feature learning melakukan proses *encoding* dari sebuah *image* ke matriks yang berupa angka - angka, dimana angka tersebut menampilkan sebuah *image* tersebut. Pada tahap ini memiliki dua *layer* yang saling bekerjasama untuk mengambil ciri – ciri berupa informasi dari sebuah gambar, yaitu *convolutional layer* dan *pooling layer*.

1) *Convolutional Layer*

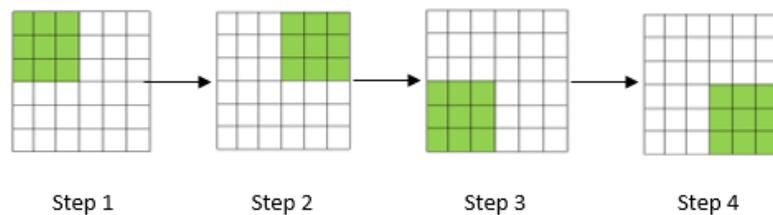
Convolutional layer melakukan operasi konvolusi pada pengolahan citra yang berarti mengalikan antara input dengan filter berukuran tertentu. Untuk dapat memahami dengan mudah, berikut contoh proses *convolutional layer*:

Gambar 2.11 Proses *Convolutional layer*

Pada Gambar 2.11 terdapat *input image* yang telah dikonversi ke dalam piksel dengan ukuran matriks panjang x lebar (6x6). Selanjutnya pada bagian tengah terdapat filter berukuran 3x3. Pada proses *convolution layer* ini akan mengalikan setiap piksel pada gambar (kotak berwarna merah) dengan *filter*. Kotak berwarna merah akan bergerak dari sudut kiri atas ke sudut kanan bawah dengan *stride* (pergeseran) sebanyak 1. Kemudian hasil dari perkalian tiap pixelnya dijumlahkan. Begitu seterusnya sampai pixel pada *feature map* terpenuhi. Tujuan konvolusi pada citra sendiri mengekstraksi fitur dari citra input untuk mendapatkan *feature map* yang berisi fitur – fitur atau informasi dari input yang terdeteksi dari gambar sehingga didapati suatu ciri khusus dari gambar yang diinputkan.

a) *Stide*

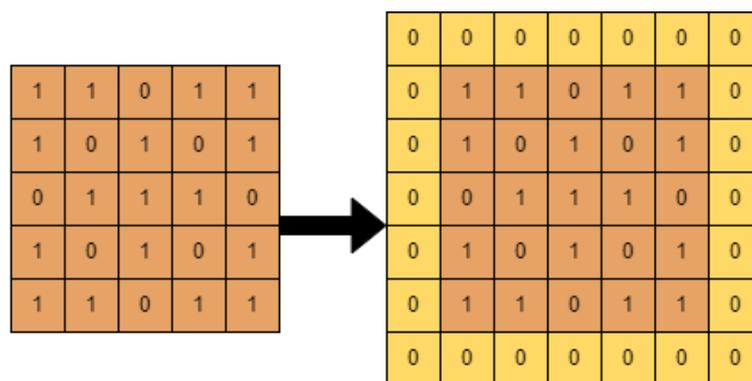
Stride (S) merupakan sebuah parameter yang akan menentukan berapa jumlah pergeseran pada *filter* yang akan digunakan. Jika nilai *stride* 1, maka *filter* akan bergeser sebanyak 1 piksel secara horizontal ke kanan sampai selesai. Setelah pergeseran piksel secara horizontal selesai, kemudian dilanjutkan bergeser lagi secara vertikal ke bawah.

Gambar 2.12 *Stride* konvolusi

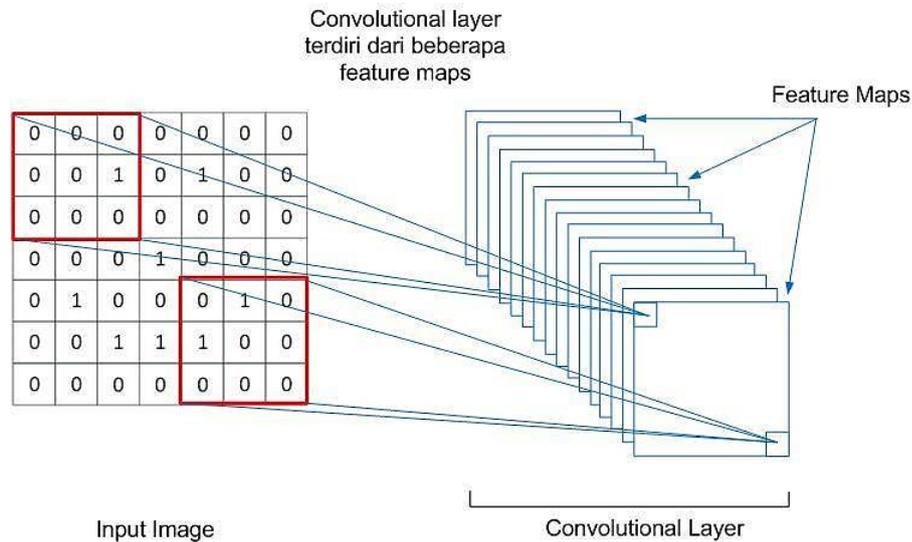
Pada gambar 2.12 menunjukkan proses konvolusi dengan menggunakan ukuran *filter* 3x3, dengan menggunakan *stride* 1. Maksud penggunaan *stride* disini adalah untuk menentukan jumlah pergeseran *filter* terhadap matriks *input* sebanyak satu *stride*.

b) *Padding*

Padding merupakan parameter yang menentukan jumlah piksel (berisi nilai 0) yang nantinya akan ditambahkan pada setiap sisi dari matriks *input*. Tujuan penggunaan *padding* adalah untuk memnipulasi dimensi *output* dari proses konvolusi yaitu *feature map*. Agar mendapatkan hasil konvolusi yang lebih dalam lagi sehingga mendapatkan lebih banyak ekstraksi dari *inputan*.

Gambar 2.13 Penambahan *Padding*

Dalam proses CNN, terdapat banyak filter, sehingga akan menghasilkan satu set *feature map* yang berisi berbagai variasi ekstraksi yang dihasilkan. Kumpulan dari *feature map* tersebut disebut dengan *convolution layer*.



Gambar 2.14 Proses pembuatan *convolution layer* dari beberapa *feature map*
(Sumber: <https://www.megabagus.id/deep-learning>)

Terdapat beberapa hal yang perlu diperhatikan pada proses *convolutional layer*, yaitu:

- Dalam sebuah *convolutional layer*, filter yang digunakan berukuran sama untuk memudahkan proses komputasi
- Ukuran ketebalan dari suatu filter selalu mengikuti ketebalan dari gambar yang digunakan sebagai *input*
- Pada umumnya tinggi dan lebar sebuah filter berukuran ganjil. Dimana filter dengan ukuran ganjil mampu memberikan representasi yang lebih baik karena bagian kiri dan kanan seimbang
- Jumlah kernel yang digunakan dalam sebuah *convolutional layer* adalah kelipatan 2 untuk meningkatkan efisiensi

- Besaran *padding* (P) umumnya menyesuaikan agar ukuran spasial dari *output* yang dihasilkan tetap sama dengan ukuran spasial *input*

$$(P = F - 1 / 2)$$

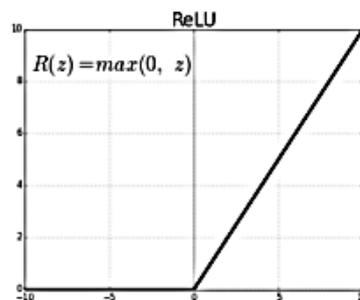
Untuk menghitung dimensi dari *feature map* kita bisa gunakan rumus seperti dibawah ini:

$$Output = \frac{W - N + 2P}{S} + 1$$

- W = Panjang/Tinggi Input
- N = Panjang/Tinggi Filter
- P = Zero Padding
- S = Stride

2) *Rectified Linear Unit (ReLU) activation*

ReLU layer berfungsi untuk menerapkan fungsi aktivasi terhadap nilai *output* hasil proses konvolusi. *Output* dari fungsi aktivasi dinyatakan sebagai 0 (nol) apabila *input*-nya negative. Namun kebalikannya jika *input*-nya positif, maka *output* nya akan sama dengan nilai *input* fungsi aktivasi itu sendiri. Berikut gambar grafik aktivasi pada *ReLU layer*.



Gambar 2.15 Grafik fungsi aktivasi *ReLU*

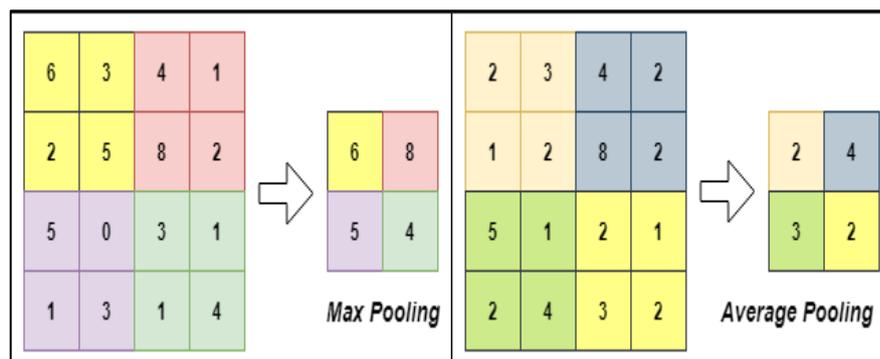
Pada fungsi ini akan memasukkan neuron – neuron berupa bilangan negative, maka tujuan fungsi aktivasi ini akan mengubah nilai tersebut menjadi nilai

0, dan jika *inputan* bernilai positif maka *output* dari *neuron* adalah nilai aktivasi itu sendiri. Fungsi aktivasi ini mempunyai kelebihan yaitu mampu mempercepat proses konvigurasi yang dilakukan dengan *Stochastic Gradient Descent* (SGD) jika dibandingkan dengan fungsi lainnya. Disisi lain, fungsi aktivasi ini juga memiliki kelemahan yaitu aktivasi ini bisa menjadi rapuh pada proses *training* dan bisa membuat unit tersebut mati.

3) *Pooling Layer*

Pooling layer adalah suatu lapisan yang berada setelah *convolutional layer*.

Seperti halnya dengan *Convolutional layer*, pada *pooling layer* akan mengurangi ukuran spasial dari *feature map*. Tujuan penggunaan *pooling layer* adalah untuk mengurangi ukuran dimensi dari *feature map* yang diperlukan dalam memproses data melalui pengurangan dimensi, dengan demikian dapat mempercepat proses perhitungan karena parameter yang perlu di update semakin sedikit serta dapat mengatasi *overfitting*. Berikut 2 gambar contoh proses *pooling layer* yaitu, *Max Pooling* dan *Average Pooling*.

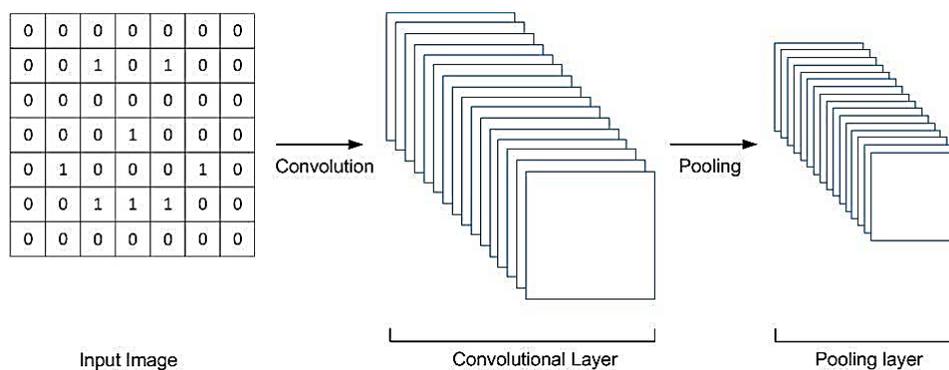


Gambar 2.16 Teknik Pooling

Pada gambar 2.8 merupakan contoh penggunaan teknik *pooling* menggunakan ukuran *filter* 2x2 dan *stride* bernilai 2. Pada proses *pooling* di atas menggunakan *input* (masukan) berukuran 4x4, masing - masing dari 4 angka yang

ada pada *input* tersebut akan diambil nilai maksimalnya atau nilai rata - rata nya. Selanjutnya membuat *output* baru dengan ukuran 2x2. Jika pada teknik *max pooling*, maka saat pergeseran filter pada area 2x2 pixel akan memilih nilai maksimum dari bagian gambar yang dicakup oleh filter. Sedangkan pada teknik *average pooling* setiap pergeseran *filter* 2x2 akan memilih rata – rata semua nilai dari bagian yang dicakup oleh filter. Melalui proses *pooling*, ukuran gambar akan menjadi lebih kecil. Dengan demikian, proses komputasi CNN akan semakin cepat.

Pada proses *pooling* ini akan menghasilkan *pooling layer* (*pooled feature map*). Dalam proses CNN, akan melakukan beberapa kali proses *pooling* sehingga banyak *pooling* yang dihasilkan. Berikut gambar proses *convolution layer* menjadi *pooling layer*.



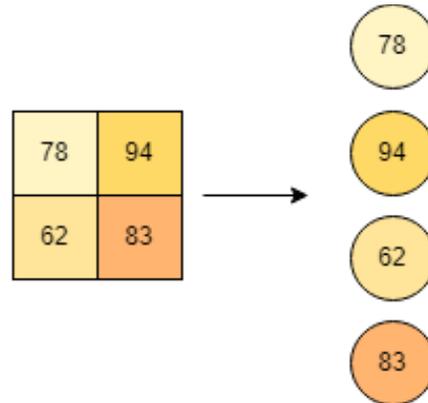
Gambar 2.17 Proses gambar dari *convolution layer* ke *pooling layer*
(Sumber: <https://www.megabagus.id/deep-learning>)

2.8.2 Classification

Proses *classification* berfungsi untuk melakukan klasifikasi terhadap setiap *neuron* yang telah diekstraksi pada proses sebelumnya yaitu proses *feature learning*. Pada bagian ini terdiri dari beberapa *layer* yang saling berkaitan satu sama lain, diantaranya *flatten* dan *fully connected layer*. Berikut penjelasan dari setiap fungsi pada bagian *classification*.

1) *Flatten*

Pada proses *flatten* akan membentuk ulang (*reshape*) *feature map* hasil dari proses *feature learning* yang masih dalam bentuk *multidimensional array* menjadi vector 1 dimensi untuk dijadikan sebagai *input* pada proses *fully connected layer*. Berikut adalah ilustrasi me-*reshape feature map* pada proses *flatten*.

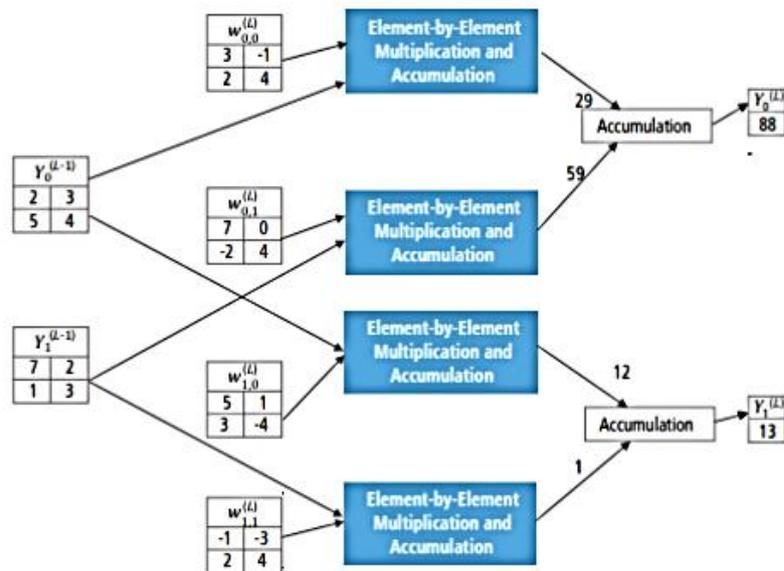


Gambar 2.18 Proses *flatten*

2) *Fully connected layer*

Fully connected layer sering digunakan pada penerapan multi layer perceptron (MLP) dengan tujuan untuk melakukan transformasi pada dimensi data agar data-data tersebut dapat diklasifikasikan secara linear. Setiap *neuron* pada *convolution layer* akan ditransformasikan menjadi satu dimensi terlebih dahulu sebelum dimasukkan ke *layer* ini. *Input* yang digunakan pada *fully connected layer* adalah data dari *feature learning* yang berbentuk vector, yang sebelumnya telah dilakukan proses *flatten*.

Perbedaan antara *fully connected layer* dengan lapisan konvolusi adalah neuron di lapisan konvolusi terhubung hanya ke daerah tertentu pada *input*, sedangkan pada *fully connected layer* memiliki neuron yang secara keseluruhan terhubung. Namun, kedua lapisan tersebut masih mengoperasikan produk dot, sehingga fungsinya tidak seluruhnya berbeda. Berikut proses *fully connected layer*:

Gambar 2.19 Proses *fully connected layer*

2.8.3 Softmax Classifier

Softmax classifier adalah generalisasi dari fungsi logistic. *Output* dari *softmax* ini dapat digunakan untuk mewakili distribusi sebuah kategori. *Softmax function* digunakan dalam berbagai macam metode klasifikasi, misalnya *multinomial logistic regression*, *multiclass linear discriminant analysis*, *naïve Bayes classifier* dan *neural network*. Secara spesifikasinya fungsi ini dapat digunakan pada metode klasifikasi *multinomial logistic regression*, *multiclass linear discriminant analysis*. Berikut rumus fungsi yang digunakan:

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}}$$

Notasi f_j menunjukkan hasil fungsi untuk setiap elemen ke- j pada vector *output* kelas. Argument z merupakan hipotesis yang diberikan oleh model pelatihan agar dapat diklasifikasikan oleh fungsi *softmax*. Fungsi ini akan memberikan hasil yang lebih intuitif dan memiliki interpretasi probabilistic yang lebih baik jika dibandingkan dengan algoritma klasifikasi lainnya. *Softmax* akan memungkinkan

kita untuk menghitung probabilitas untuk semua label. Label yang ada akan diambil sebuah vektor yang bernilai riil dan merubahnya menjadi vektor dengan nilai antara nol dan satu, yang mana apabila jika dijumlahkan semua akan bernilai satu.

2.8.4 *Cross Entropy Loss Function*

Loss function atau *cross function* merupakan fungsi yang menggambarkan kerugian yang terkait dengan semua kemungkinan yang dihasilkan oleh suatu model. *Loss function* akan bekerja ketika model pembelajaran memberikan kesalahan yang harus diperhatikan. *Loss function* yang baik adalah ketika fungsi tersebut menghasilkan nilai error yang paling rendah.

Ketika sebuah model memiliki kelas yang cukup banyak, maka perlu adanya cara untuk mengukur perbedaan antara probabilitas hasil hipotesis dan probabilitas kebenaran yang asli serta selama proses *training* dengan banyak algoritma yang dapat menyesuaikan parameter sehingga perbedaan ini dapat diminimalkan. *Crossentropy* adalah sebuah pilihan yang masuk akal. Dimana gambaran umum algoritma ini adalah meminimalkan kemungkinan log negative dari suatu dataset yang merupakan ukuran langsung dari sebuah performa prediksi suatu model. Berikut adalah fungsi yang diberikan:

$$\frac{1}{N} \sum_x (\text{target}(x) - \text{activation}(x))^2 = \frac{1}{N} \sum_x (\text{target}(x) - \max(0, \sum |x| wixi + x i x b))^2$$

2.9 *Confussion Matriks*

Penentuan baik atau tidaknya sebuah performa suatu model klasifikasi dapat dilihat dari parameter pengukuran performanya, yaitu tingkat akurasi. Untuk menghitung faktor – faktor tersebut maka diperlukan sebuah matriks yang bisa

disebut dengan *confussion matriks*. Salah satu *confussion matriks* yang sering digunakan dalam pengukuran dapat dilihat pada gambar di bawah (Fawcett, 2006).

Tabel 2.1 *Confussion Matriks*

Hipotesis Kejadian		Kejadian Sebenarnya	
		P	N
	P	True Positive	False Positive
	N	False Negative	True Negative

Berdasarkan gambar di atas, terdapat beberapa nilai dalam matriks “*True positive*” (TP), “*True Negative*” (TN), “*False Positive*” (FP), dan “*False Negative*” (FN), seluruh kemungkinan kejadian sebenarnya positif (P) dan seluruh kemungkinan kejadian sebenarnya negative (N). Nilai tersebut dapat digunakan untuk menghitung akurasi dengan persamaan seperti di bawah.

$$\text{Akurasi} = \frac{TP+TN}{P+N} \times 100\%$$

Hasil nilai akurasi ini nanti nya digunakan sebagai parameter sebagaimana keakuratan suatu model dalam melakukan klasifikasi.

2.10 *Python*

Python merupakan salah satu contoh bahasa pemrograman tingkat tinggi. Contoh lain Bahasa tingkat tinggi antara lain Pascal, Pert, C++, Java dan sebagainya. Sedangkan Bahasa pemrograman tingkat rendah merupakan bahasa mesin (*assembly*). Secara sederhana, sebuah komputer hanya mampu mengeksekusi program yang ditulis dalam bentuk bahasa mesin. Oleh karena itu, jika suatu program ditulis dalam bentuk bahasa mesin, maka program tersebut harus

diproses terlebih dahulu sebelum dijalankan dalam komputer. Hal ini merupakan salah satu kekurangan dari bahasa tingkat tinggi yang mana harus memerlukan waktu lama untuk memproses suatu program terlebih dahulu sebelum program tersebut dijalankan. Di sisi lain, bahasa tingkat tinggi juga mempunyai banyak keuntungan diantaranya, mudah dipelajari, mudah ditulis, mudah dibaca dan tentu saja juga memudahkan dalam mencari kesalahannya. Selain itu, bahasa Bahasa komputer yang dikembangkan khusus dalam membuat *source code* agar mudah dibaca. Dalam bahasa pemrograman *python* memiliki beberapa *library* yang lengkap sehingga dapat membantu serta memudahkan seorang dalam membuat suatu aplikasi yang sesuai dengan keinginan dengan *source code* yang terlihat secara sederhana.

2.11 Keras

Keras merupakan sebuah *framework* yang dibuat dengan tujuan untuk mempermudah dalam proses pembelajaran terhadap komputer. Keras adalah *API neural network* level tinggi yang ditulis dengan menggunakan bahasa pemrograman *Python*, yang dapat berjalan di atas Tensorflow, Theano atau CNTK sebagai *backend*. Dalam penelitian tentang *deep learning* dapat menggunakan Keras untuk keperluan pembuatan *prototipe* dengan mudah dan juga cepat. Selain itu, Keras juga sangat mendukung dalam proses komputasi pada algoritma *convolutional neural network* (CNN). Keras dapat berjalan dengan baik dengan menggunakan CPU maupun GPU. (Keras.io).

2.12 Tensorflow

Tensorflow merupakan sebuah perpustakaan dari perangkat lunak yang dikembangkan oleh Google Brain team dalam suatu organisasi penelitian Mesin Cerdas Google. Tujuan menggunakan *library* ini adalah untuk melakukan proses pembelajaran mesin dan penelitian jaringan syaraf dalam. Tensorflow kemudian menggabungkan aljabar komputasi dengan cara mengoptimalkn kompilasi untuk mempermudah perhitungan banyaknya ekspresi secara matematis. Permasalahan dalam melakukan proses komputasi ini adalah waktu yang dibutuhkan saat melakukan proses perhitungan.

Pada Tensorflow ini terdapat beberapa fitur utama, meliputi:

- Skalabilitas komputasi yang tinggi di seluruh mesin dan kumpulan data yang berukuran besar
- Penggunaan GPU yang transparan dapat mengotomatisasi manajemen serta mengoptimalisasi memori yang sama dengan data yang digunakan
- Tensorflow mampu menulis kode yang sama dan menjalankannya dengan baik di CPU dan GPU. Selain itu, Tensorflow juga dapat mengetahui bagian perhitungan man yang seharusnya dipindahkan ke GPU
- Pemrograman pendukung berupa jaringan syaraf tiruan dalam serta Teknik pembelajaran mesin
- Mampu mengoptimalkan, mengidentifikasi serta menghitung ekspresi matematis yang melibatkan array multidimensi secara efisien.

2.13 *Google Colaboratory*

Google colaboratory adalah sebuah *tools* yang diciptakan oleh *Google*. *Google colab* ini memberikan beberapa fasilitas kepada para pengguna yang ingin mempelajari serta mengolah data dengan menggunakan *machine learning* atau *deep learning*. *Google colab* menyediakan sebuah layanan GPU gratis sebagai *backend* proses komputasi yang mampu digunakan selama 12 jam.

Google Colaboraty dibuat ata *environment Jupyter* sehingga penggunaannya hamper mirip dengan *jupyter Notebook*, namun berbeda pada penyimpanannya saja. Media penyimpanan pada *google colab* ini menggunakan *google drive* dan berjalan pada system berbasis *cloud*. Dengan menggunakan *tools Google Colaboratory* ini dapat membantu dalm membuat suatu aplikasi yang berbasis *deep learning* dengan menggunakan berbagai *library* seperti *Tensorflow*, *Keras*, *Pytorch* serta *OpenCV* (Digimi, 2018).

BAB III

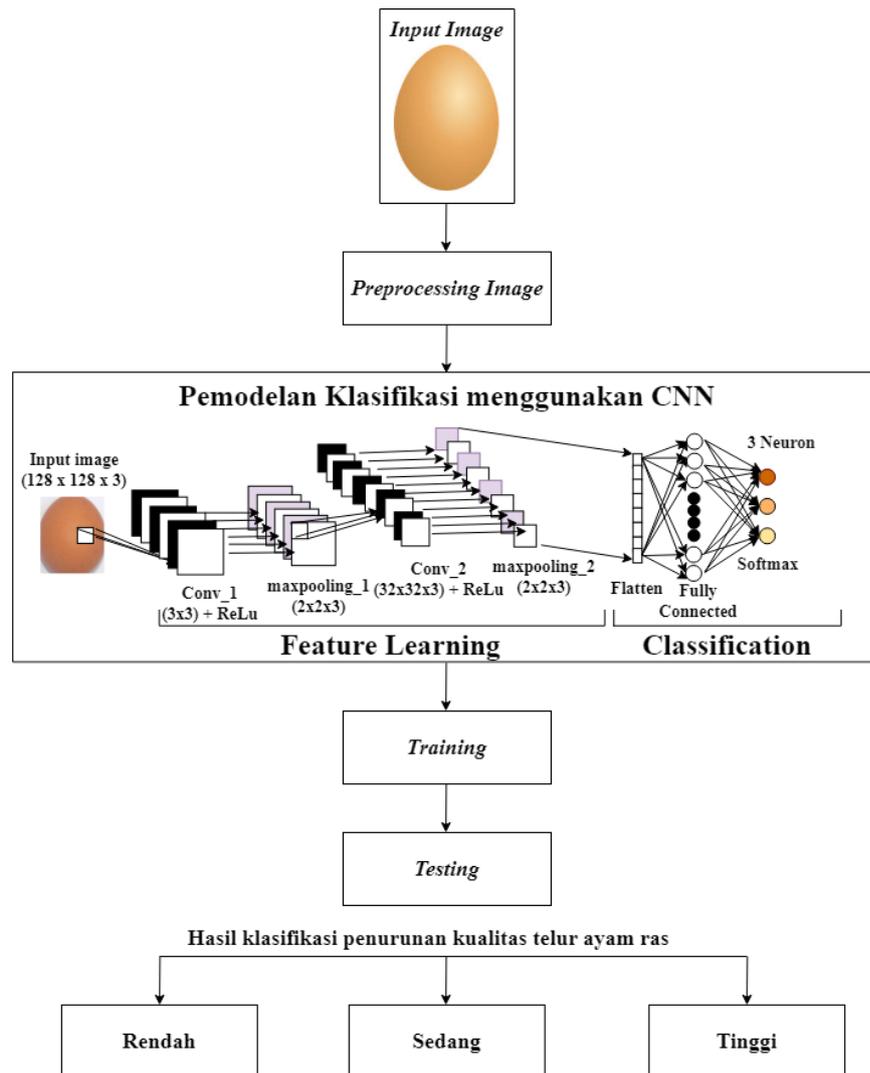
ANALISIS DAN PERANCANGAN SISTEM

3.1 Perancangan Sistem

Pada bab ini menjelaskan mengenai analisis dan rancangan sistem yang akan dibangun pada penelitian ini, sehingga dapat berjalan dengan baik dan terstruktur. Penelitian ini dirancang untuk melakukan klasifikasi penurunan kualitas telur ayam ras berdasarkan warna kerabang telur berbasis citra dengan menggunakan algoritma *Convolutional Neural Network* (CNN). Dalam membangun sistem klasifikasi penurunan kualitas telur digunakan *Google Colaboratory* untuk menulis dan mengeksekusi barisan kode *Python* di *browser*. Sistem yang akan dibangun pada penelitian ini memiliki 5 tahapan, diantaranya:

1. *Input image*
2. *Preprocessing Image*
3. Pemodelan klasifikasi menggunakan algoritma *Convolutional Neural Network* (CNN)
4. Proses *Training*
5. Proses *Testing*

Berikut desain rancangan sistem yang akan dibangun pada penelitian ini seperti gambar 3.1 di bawah:



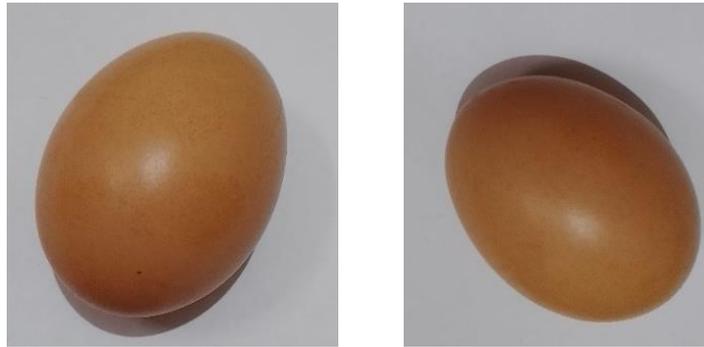
Gambar 3.1 Desain Sistem

3.1.1 Input Image

Data yang digunakan sebagai *input image* merupakan hasil dari foto telur ayam ras berdasarkan warna kerabang. Diantara warna kerabang tersebut dibagi menjadi 3 warna yaitu, warna coklat tua, warna coklat dan warna coklat muda. Proses pengambilan gambar telur ayam ras dilakukan menggunakan kamera *smartphone* dengan resolusi 8 *megapixel*. Pada proses pengambilan foto, gambar diambil dari sisi atas telur dengan menggunakan *background* atau latar belakang

warna putih sebagai alas telur. Diantara foto telur berdasarkan warna kerabang akan diidentifikasi menjadi 3 kategori berdasarkan penurunan kualitasnya, yaitu:

1. Telur warna kerabang coklat tua, menunjukkan penurunan kualitas pada telur rendah.



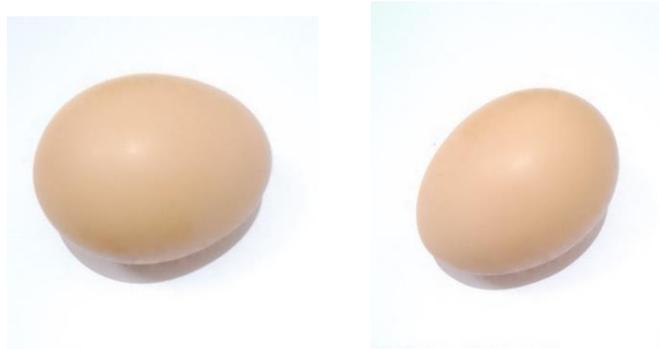
Gambar 3.2 Telur warna kerabang coklat tua

2. Telur warna kerabang coklat, menunjukkan penurunan kualitas pada telur sedang.



Gambar 3.3 Telur warna kerabang coklat

3. Telur warna kerabang coklat muda, menunjukkan penurunan kualitas pada telur semakin tinggi.



Gambar 3.4 Telur warna kerabang coklat muda

Jumlah data yang didapatkan masing – masing foto telur berdasarkan warna kerabang yaitu sebanyak 300 gambar telur. Diantaranya 100 gambar telur dengan kerabang warna coklat tua, 100 gambar telur dengan kerabang warna coklat dan 100 gambar telur dengan kerabang warna coklat muda.

Sebelum dilakukan proses *training* dan *testing*, maka dilakukan pembagian dataset untuk keperluan sebagai data latih dan data uji terlebih dahulu. Berikut *source code* proses *split* data.

```
X_train, X_test, y_train, y_test = train_test_split
(X, y, test_size=0.20, random_state=300)
```

Gambar 3.5 *source code* proses *split* data

Berdasarkan *source code* di atas, dataset dibagi menjadi 2 yaitu data *training* dan data *testing*, proporsi data *training* yang digunakan adalah 80%, 70% dan 60% dari jumlah keseluruhan data, sedangkan proporsi untuk data *testing* yang digunakan sebesar 20%, 30% dan 40%. Ukuran citra yang akan digunakan sebagai citra latih dan citra uji terdapat 3 ukuran citra yang berbeda yaitu (64x64), (128x128) dan (256x256) piksel. Berikut pembagian dataset yang digunakan terdapat pada tabel 3.1.

Tabel 3.1 Pembagian Dataset

Ukuran Gambar	Pembagian Data					
	60% : 40%		70% : 30%		80% : 20%	
	Data Latih	Data Uji	Data Latih	Data Uji	Data Latih	Data Uji
64 x 64	60	40	70	30	80	20
128 x 128	60	40	70	30	80	20
256 x 256	60	40	70	30	80	20

3.1.2 Preprocessing Image

Setelah melakukan pembagian data, tahap selanjutnya adalah melakukan *preprocessing image* untuk memproses pengolahan data yang telah dikumpulkan sebelumnya terlebih dahulu agar lebih memudahkan algoritma CNN ketika melakukan proses pelatihan, sehingga mampu mengenali ciri – ciri dari data yang di *inputkan*. *Preprocessing image* dilakukan untuk memperbanyak data yang nantinya akan digunakan sebagai data inputan. Untuk mendapatkan performa yang optimal, *deep learning* membutuhkan data dengan jumlah yang lebih banyak dan bervariasi. Karena variasi objek serta kemampuan ketika pengambilan data pada penelitian ini cukup terbatas, maka pada penelitian ini melakukan *preprocessing image* menggunakan teknik augmentasi data untuk memperbanyak variasi data tanpa menghilangkan bagian inti dari data tersebut. Dengan demikian mesin mampu belajar untuk mengenali berbagai citra dengan bentuk dan pola yang beragam. Pada proses ini dilakukan dengan memanfaatkan *library* ImageDataGenerator yang disediakan oleh *python* dengan tujuan untuk mempermudah dalam mengelolah data yang akan digunakan sebagai inputan. Teknik augmentasi data yang digunakan pada penelitian ini diantaranya *rotate*, *width & height shift range*, *rescale*, *shear*, *zoom* dan *horizontalFlip*.

```

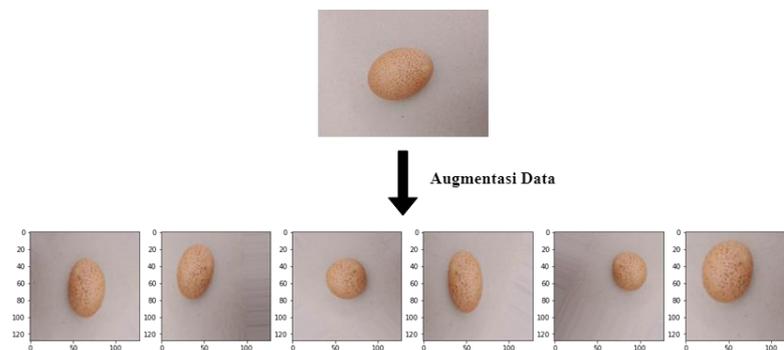
ImageDataGenerator(rotation_range=40,
                   width_shift_range=0.2,
                   height_shift_range=0.2,
                   shear_range=0.2,
                   zoom_range=0.2,
                   horizontal_flip=True

```

Gambar 3. 6 *Source code* Augmentasi data

Pada kode di atas merupakan kode untuk memanggil fungsi `ImageDataGenerator` yang digunakan saat melakukan proses augmentasi data. Diantara teknik augmentasi data yang dilakukan pada gambar antara lain melakukan *rotate* untuk memutar gambar sebesar sudut 40 derajat secara acak, selanjutnya melakukan teknik *width & height shift* untuk mengatur posisi gambar berdasarkan lebar dan tinggi pada gambar, kemudian melakukan teknik *shear* untuk menggeser gambar searah dengan jarum jam dengan pergeseran sebesar 0.2 derajat, selanjutnya melakukan teknik *zoom* untuk memperbesar gambar dengan perbesaran sebanyak 0.2 dari luas gambar asli dan terakhir melakukan teknik *horizontalFlip* untuk membalik gambar secara horizontal.

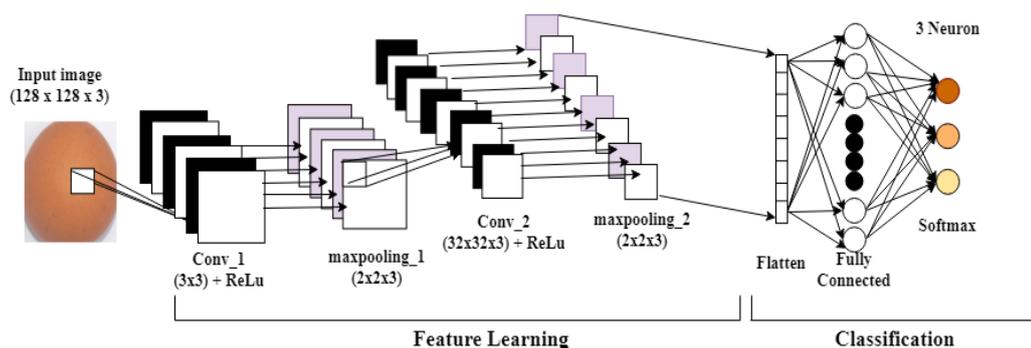
Semua fungsi pada teknik augmentasi data tersebut akan dikombinasikan dan diaplikasikan secara acak pada semua gambar sehingga menghasilkan *output* seperti berikut:



Gambar 3.7 Hasil Proses Augmentasi data

3.1.3 Pemodelan Klasifikasi Menggunakan CNN

Pada proses ini, sistem akan melatih dataset telur berdasarkan warna kerabang menggunakan algoritma *convolutional neural network* (CNN). Input gambar yang digunakan berukuran 128x128 piksel, tujuannya adalah untuk membandingkan nilai akurasi berdasarkan ukuran gambarnya. Untuk mendapatkan hasil model yang optimal, maka pada penelitian ini akan membangun arsitektur jaringan menggunakan dua lapisan konvolusi dengan fungsi aktivasi ReLu, dua *pooling layer* dan *fully connected layer*. Secara sederhana, berikut gambaran rancangan arsitektur dengan algoritma *Convolutional Neural Network* (CNN) dalam penelitian ini:



Gambar 3.8 Artitektur CNN

Berikut penjelasan arsitektur CNN di atas:

1. *Input* gambar yang digunakan merupakan hasil dari proses augmentasi dengan ukuran 128x128 piksel. Dimana maksud dari 128x128x3 adalah citra dengan ukuran piksel 128x128, sedangkan angka 3 menunjukkan bahwa citra tersebut menandakan *channel image* yang merupakan citra RGB (*red, green* dan *blue*).
2. Selanjutnya masuk pada proses konvolusi pertama. Dimana pada proses konvolusi pertama, inputan citra akan dikalikan dengan *kernel* berukuran

3x3 dengan *filter* sebanyak 32. Proses perkalian tersebut dilakukan dengan menggeser *kernel* sebanyak 1 *stride*, sehingga menghasilkan *feature map* berukuran 126x126. Setelah melakukan proses konvolusi, selanjutnya ditambahkan dengan fungsi aktivasi ReLu. Dimana fungsi ini bertujuan untuk mengubah nilai negatif menjadi nol yang ada pada sebuah matriks hasil dari proses konvolusi.

3. Pada proses *pooling* pertama akan melakukan pengurangan ukuran *feature map* yang merupakan hasil dari proses konvolusi pertama menggunakan teknik *max-pooling* dengan menggunakan *kernel* berukuran 2x2. Dimana teknik *max-pooling* akan mengambil nilai paling maksimum berdasarkan pergeseran pada kernel sebanyak nilai *stride* yaitu 2. Sehingga pada proses *pooling* pertama ini menghasilkan matriks baru berukuran 63x63 piksel.
4. Selanjutnya proses konvolusi kedua dengan inputan *feature map* hasil dari *pooling* pertama dikalikan silang dengan *kernel* berukuran 3x3 dengan *filter* sebanyak 32. Seperti halnya konvolusi pertama, proses konvolusi kedua juga dilakukan perhitungan aktivasi fungsi ReLu.
5. Proses *pooling* kedua hampir sama dengan proses *pooling* pertama, yaitu melakukan pengurangan ukuran *feature map* menggunakan teknik *max-pooling* dengan *kernel* berukuran 2x2 dengan *stride* 2 piksel. Sehingga hasil dari proses *pooling* kedua adalah *feature map* berukuran 31x31 piksel.
6. Setelah tahap *pooling* kedua, Selanjutnya proses *flatten* untuk mengubah *feature map* yang berupa 3D array menjadi vector 1D untuk dijadikan sebagai inputan pada proses *fully connected layer*.

7. Selanjutnya proses *fully connected layer* akan menghubungkan *feature map* berupa vektor hasil dari proses *flatten* ke beberapa lapisan *fully-connected*. Pada tahap ini hanya menggunakan satu *hidden layer* pada jaringan. Sebelum melakukan proses klasifikasi gambar, pada proses ini menggunakan nilai *Dropout* sebesar 0.5, tujuan proses ini yaitu untuk mengurangi terjadinya *overfitting* pada saat proses *training*.
8. Proses terakhir melakukan proses klasifikasi dengan menggunakan fungsi *softmax* untuk menghasilkan *output* berupa jumlah kelas yang sesuai dengan kategori yang telah ditentukan.

Berdasarkan penjelasan dari arsitektur jaringan di atas, maka didapatkan sebuah hasil model dari proses *training* tersebut. Berikut model CNN dengan *inputan* ukuran 128x128 yang dihasilkan:

```

Model: "sequential"
-----
Layer (type)                Output Shape              Param
-----
conv2d (Conv2D)             (None, 128, 128, 128)    3584
activation (Activation)     (None, 128, 128, 128)    0
conv2d_1 (Conv2D)          (None, 126, 126, 32)     36896
activation_1 (Activation)   (None, 126, 126, 32)     0
max_pooling2d (MaxPooling2D) (None, 63, 63, 32)       0
dropout (Dropout)          (None, 63, 63, 32)       0
conv2d_2 (Conv2D)          (None, 63, 63, 32)       9248
activation_2 (Activation)   (None, 63, 63, 32)       0
max_pooling2d_1 (MaxPooling2D) (None, 31, 31, 32)       0
dropout_1 (Dropout)        (None, 31, 31, 32)       0
flatten (Flatten)          (None, 30752)             0
dense (Dense)               (None, 512)               15745536
activation_3 (Activation)   (None, 512)               0
dropout_2 (Dropout)        (None, 512)               0
dense_1 (Dense)             (None, 3)                 1539
activation_4 (Activation)   (None, 3)                 0
-----
Total params: 15,796,803
Trainable params: 15,796,803
Non-trainable params: 0

```

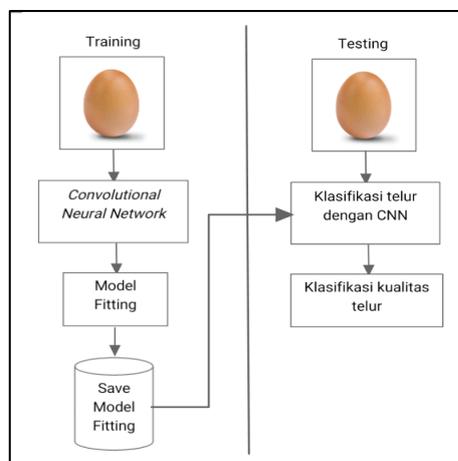
Gambar 3.9 Model CNN

Pada gambar di atas menunjukkan proses konvolusi menggunakan ukuran kernel 3×3 dengan *stride* 1. Stride disini artinya jumlah pergeseran kernel terhadap matriks *input* sebanyak 1.

Agar sistem dapat bekerja dalam mengidentifikasi penurunan kualitas telur, maka sistem perlu di *training* terlebih dahulu dengan menggunakan dataset *training* dan *testing* yang telah dikumpulkan sebelumnya.

3.2 Rancangan Percobaan

Proses selanjutnya dilakukan percobaan pada dataset yang telah dikumpulkan sebelumnya dan dibagi menjadi data latih dan uji. Dalam proses pelatihan data digunakan *test size* sebesar 0.2 dengan *random state* nya 300, seperti pada gambar 3.5 tentang *source code split data*. Tujuan dari melatih algoritma CNN ini agar mampu mengenali ciri – ciri dari setiap gambar, kemudian ditandai dengan *neuron-neuron* mana yang akan diaktifkan saat gambar diklasifikasi. Dengan demikian perlu dibuat model untuk melatih algoritma sebelum dilakukan pengujian, dengan tujuan agar algoritma sudah terlatih. Secara sederhana, berikut skema model *training* yang akan dilakukan untuk mendapatkan pola menggunakan algoritma CNN.



Gambar 3.11 *Training* dan *Testing*

Pada model di atas akan dilakukan proses *training* pada dataset sesuai dengan *training* model yang telah dibuat sebelumnya. Proses untuk melakukan pelatihan model ini disebut dengan model *fitting*. Dimana model *fitting* merupakan suatu model yang berisi parameter – parameter yang menunjukkan ciri khusus yang dimiliki pada setiap gambar. Proses model *fitting* dapat dilakukan dengan memanggil method `model.fit()`, kemudian memasukkan variabel data dan parameter iterasi yang harus dijalankan selama proses model *fitting* berlangsung.

Ketika proses model *fitting* dijalankan, maka arsitektur CNN yang sudah dibentuk langsung bekerja mengeksekusi data yang sudah disiapkan sebelumnya. Proses *training* dilakukan dengan sejumlah *epoch* yang sudah ditentukan, yang mana ketika menjalankan iterasi, sistem akan langsung otomatis menampilkan hasil *accuracy value* dan *loss value* dari data *training*. Setelah proses ini selesai, selanjutnya model yang telah terbentuk disimpan untuk dipanggil saat proses *testing*.

Adapun tahapan untuk memperoleh hasil model arsitektur CNN yang terbaik, maka dapat ditentukan dari parameter yang digunakan. Pada penelitian ini, peneliti menggunakan 3 parameter yang akan digunakan sebagai perbandingan untuk mengetahui hasil model arsitektur mana yang terbaik. Diantaranya 3 parameter sebagai pembanding tersebut adalah:

1. Jumlah data *training* dan data *testing*

Jumlah data *training* dan data *testing* yang akan digunakan sebagai perbandingan pada proses *training* ini adalah 80% : 20%, 70% : 30% dan 60% : 40%.

2. Nilai *epoch*

Nilai *epoch* yang akan digunakan sebagai perbandingan ketika dilakukan proses *training* ini adalah *epoch* 50, 75 dan 100.

3. Ukuran gambar

Ukuran gambar yang akan digunakan sebagai perbandingan pada proses *training* diantaranya ukuran 64x64, 128x128 dan 256x256.

3.3 Rancangan Pengujian

Pada pengujian ini menguji seberapa baik kinerja dari model yang sebelumnya sudah *training*. Pada proses *testing* ini sistem akan menguji *image* baru sebanyak 78 data dengan setiap kelas jenis telur berdasarkan warna sebanyak 34 data yang belum pernah digunakan oleh model tersebut, tujuannya agar dapat diketahui seberapa baik kinerja model yang telah dibuat untuk mengidentifikasi. Pada proses *testing* akan menghitung nilai *loss* dan akurasi dari model – model yang dihasilkan dari data uji dengan menggunakan model *fitting* hasil dari proses *training* sebagai pembanding untuk melakukan klasifikasi terhadap penurunan kualitas telur ayam seperti gambar inputan pada gambar 3.11 dengan 3 kategori berdasarkan warna kerabang. Dengan demikian, dapat diketahui hasil klasifikasi pada penurunan kualitas telur ayam ras berdasarkan warna kerabang.

3.4 Pengukuran Akurasi

Pada Pengukuran akurasi akan melakukan evaluasi model, sehingga dapat mengetahui seberapa akurat model yang dibangun dalam mengklasifikasi penurunan kualitas telur berdasarkan warna kerabang menggunakan metode *confusion matrix* yang memanfaatkan tabel matriks untuk menggambarkan kinerja suatu model sistem yang telah dibangun berdasarkan data uji yang diprediksi benar

atau tidak benar. Hasil prediksi sistem tersebut selanjutnya dimasukkan ke dalam tabel matriks yang akan memberikan informasi perbandingan hasil klasifikasi sistem dengan membandingkan nilai aktual data sebenarnya dengan nilai prediksi sistem. Dimana ukuran tabel matriks berdasarkan dari banyaknya kelas dalam klasifikasi sistem sehingga, pada penelitian ini terdapat 9 perbandingan yang akan dimasukkan ke dalam tabel matriks 3 x 3 berdasarkan klasifikasi penurunan kualitas telur berdasarkan warna kerabang telur yang berjumlah 3 kelas, yaitu penurunan kualitas telur semakin tinggi terdapat pada warna kerabang coklat muda (CM), sedang pada warna kerabang coklat (C) dan rendah pada warna kerabang coklat tua (CT). Misalnya terdapat 100 data *testing* yang akan diujikan pada model sistem dengan nilai benar CT = 30, C = 32, dan CM = 28 dapat dilihat pada tabel matriks di bawah:

Tabel 3. 2 *Confusion Matrix*

Matriks		Nilai Prediksi		
		Coklat Tua (CT)	Coklat (C)	Coklat Muda (CM)
Nilai Akurasi	Coklat Tua (CT)	30	2	1
	Coklat (C)	1	32	1
	Coklat Muda (CM)	0	5	28

Berdasarkan tabel 3.2 maka, untuk menghitung nilai akurasi pada metode *confusion matrix* dapat menggunakan rumus akurasi sebagai berikut:

$$\text{Akurasi} = \frac{\text{Jumlah Nilai Aktual dan Nilai Prediksi Benar}}{\text{Jumlah Data}} \times 100\%$$

Diketahui:

- TP (Jumlah nilai aktual dan nilai prediksi benar) = 30 + 32 + 28
- Jumlah Data = 100

Sehingga untuk menghitung nilai akurasi dari model klasifikasi berdasarkan tabel 3.4 sebagai berikut:

$$\text{Akurasi} = \frac{30+32+28}{100} \times 100\% = \frac{90}{100} \times 100\% = 90\%$$

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Uji Coba

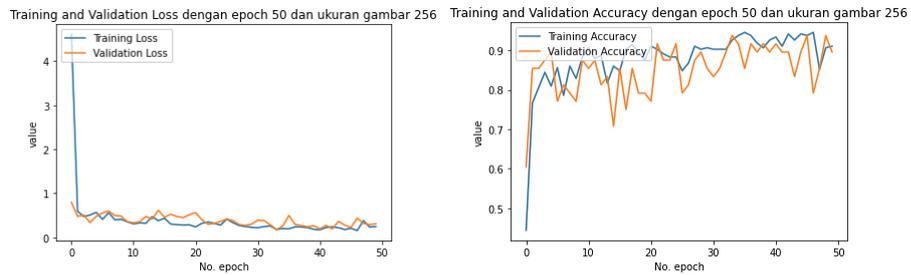
Pada bab ini akan memaparkan serta mengevaluasi hasil pengujian dari skenario uji coba yang telah dipaparkan pada sub bab sebelumnya. Dalam skenario tersebut telah didapatkan model CNN dari proses *training* yang dilakukan. Model tersebut nanti nya diterapkan ke dalam data *testing* untuk di uji kemampuannya dalam mengklasifikasi penurunan kualitas telur ayam berdasarkan warna kerabang. Berikut hasil uji coba pada sistem klasifikasi penurunan kualitas telur berdasarkan warna kerabang menggunakan algoritma CNN.

4.1.1 Hasil Proses *Training*

Bagian terpenting dari berhasilnya proses klasifikasi penurunan kualitas telur berdasar warna kerabang ini adalah baiknya hasil dari proses *training* ini. Karena baiknya hasil dari proses *training* ini yang akan memberikan dampak tinggi terhadap hasil yang didapatkan saat dilakukan proses *testing* nanti.

Pada proses *training*, parameter yang digunakan untuk mengetahui tinggi rendahnya nilai akurasi yang didapat diantaranya ukuran gambar yang berbeda yang terdapat 3 ukuran yaitu (256, 256), (128, 128) dan (64, 64) dan mengganti nilai epoch pada masing - masing ukuran gambar dengan parameter epoch1 = 50, epoch2 = 75 dan epoch3 = 100. Sehingga didapatkan hasil pada proses *training* sebagai berikut:

a. Gambar inputan (256, 256) dengan *Epoch* 50



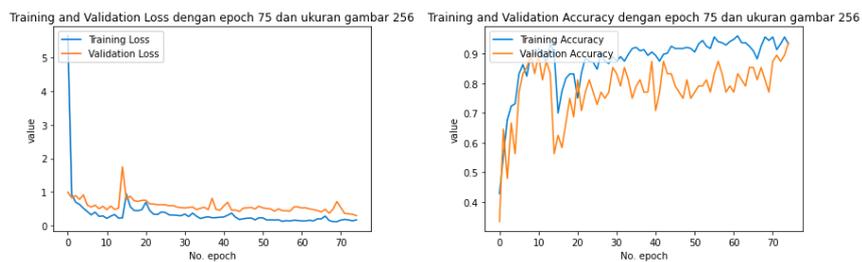
Gambar 4.1 Hasil *Training* Gambar Ukuran 256 dan *Epoch* 50

Epoch 50/50

17/17 [=====] - 99s 6s/step - loss:
0.2460 - accuracy: 0.9105 - val_loss: 0.3065 - val_accuracy:
0.8958

Waktu yang dibutuhkan: 83.3814 Menit

b. Gambar inputan (256, 256) dengan *Epoch* 75



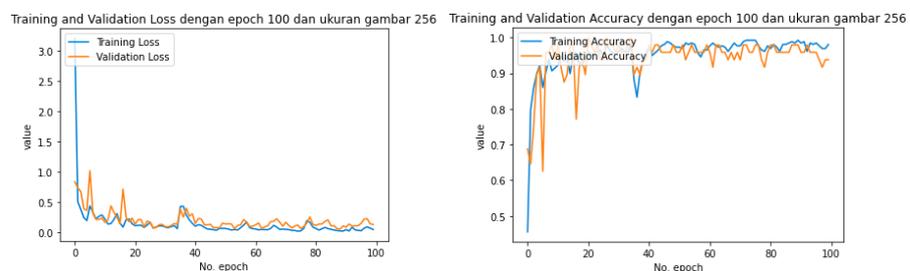
Gambar 4.2 Hasil *Training* Gambar Ukuran 256 dan *Epoch* 75

Epoch 75/75

17/17 [=====] - 100s 6s/step - loss:
0.1760 - accuracy: 0.9339 - val_loss: 0.3016 - val_accuracy:
0.9375

Waktu yang dibutuhkan: 125.9955 Menit

c. Gambar inputan (256, 256) dengan *Epoch* 100



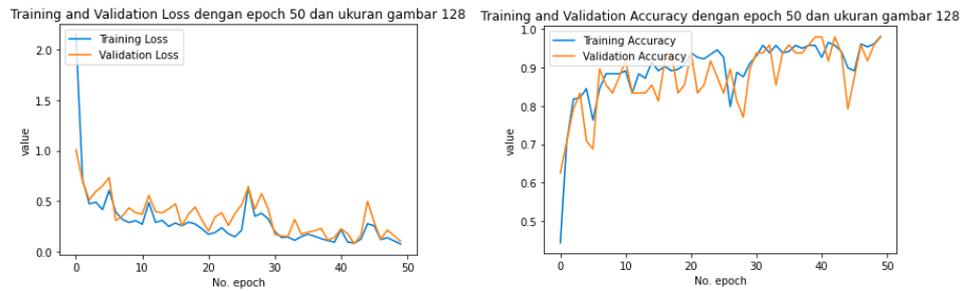
Gambar 4.3 Hasil *Training* Gambar Ukuran 256 dan *Epoch* 100

Epoch 100/100

17/17 [=====] - 98s 6s/step - loss:
0.0481 - accuracy: 0.9805 - val_loss: 0.1301 - val_accuracy:
0.9375

Waktu yang dibutuhkan: 164.3843 Menit

d. Gambar inputan (128, 128) dengan *Epoch* 50



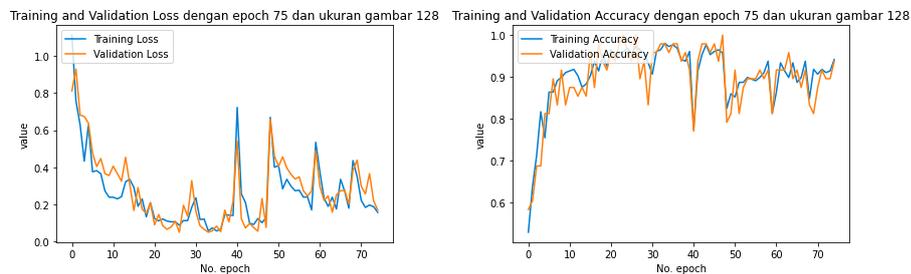
Gambar 4.4 Hasil *Training* Gambar Ukuran 128 dan *Epoch* 50

Epoch 50/50

17/17 [=====] - 49s 3s/step - loss:
0.0750 - accuracy: 0.9805 - val_loss: 0.1021 - val_accuracy:
0.9792

Waktu yang dibutuhkan: 42.3757 Menit

e. Gambar inputan (128, 128) dengan *Epoch* 75



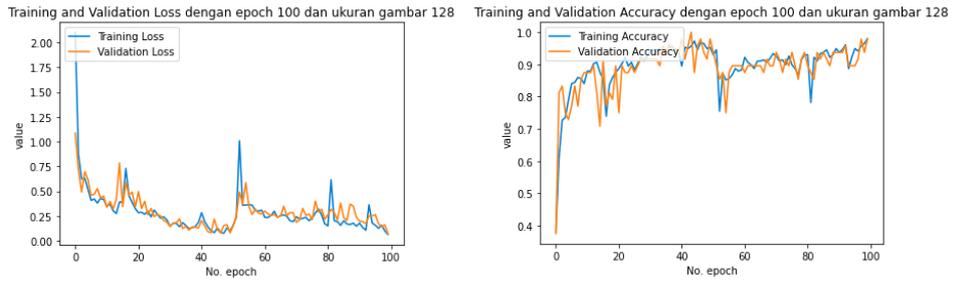
Gambar 4.5 Hasil *Training* Gambar Ukuran 128 dan *Epoch* 75

Epoch 75/75

17/17 [=====] - 50s 3s/step - loss:
0.1565 - accuracy: 0.9416 - val_loss: 0.1691 - val_accuracy:
0.9375

Waktu yang dibutuhkan: 63.8148 Menit

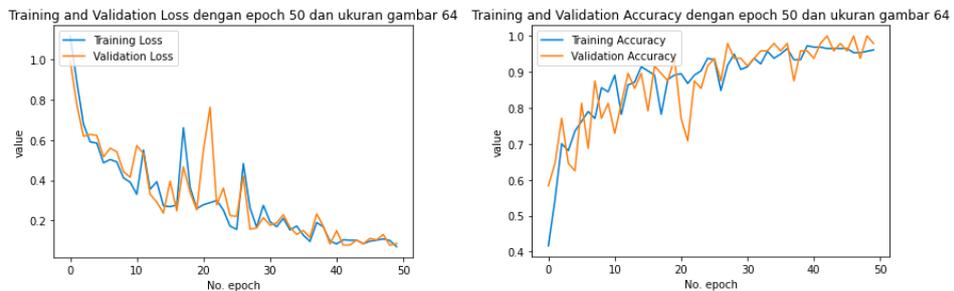
f. Gambar inputan (128, 128) dengan *Epoch* 100



Gambar 4.6 Hasil *Training* Gambar Ukuran 128 dan *Epoch* 100

Epoch 100/100
 17/17 [=====] - 50s 3s/step - loss:
 0.0644 - accuracy: 0.9805 - val_loss: 0.0725 - val_accuracy:
 0.9792
 Waktu yang dibutuhkan: 82.3399 Menit

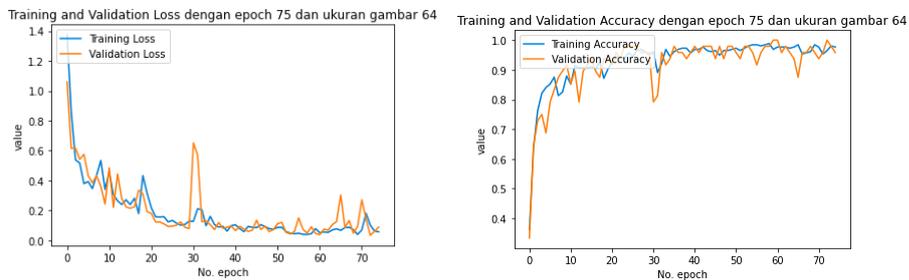
g. Gambar inputan (64, 64) dengan *Epoch* 50



Gambar 4.7 Hasil *Training* Gambar Ukuran 64 dan *Epoch* 50

Epoch 50/50
 17/17 [=====] - 38s 2s/step - loss:
 0.0704 - accuracy: 0.9611 - val_loss: 0.0865 - val_accuracy:
 0.9792
 Waktu yang dibutuhkan: 32.8590 Menit

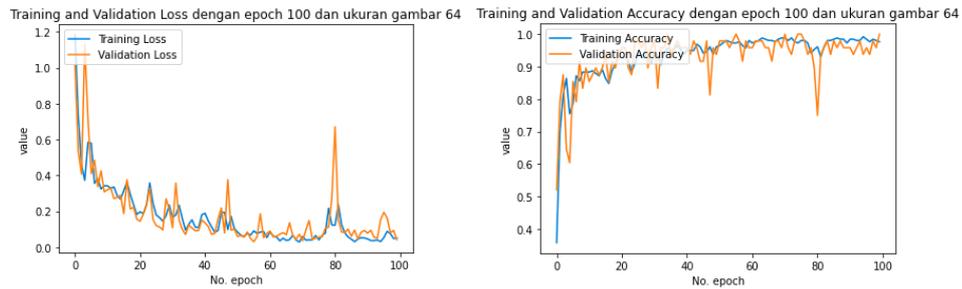
h. Gambar inputan (64, 64) dengan *Epoch* 75



Gambar 4.8 Hasil *Training* Gambar Ukuran 64 dan *Epoch* 75

Epoch 75/75
 17/17 [=====] - 38s 2s/step - loss:
 0.0568 - accuracy: 0.9767 - val_loss: 0.0885 - val_accuracy:
 0.9583
 Waktu yang dibutuhkan: 49.3792 Menit

a. Gambar inputan (64, 64) dengan *Epoch* 100



Gambar 4.9 Hasil *Training* Gambar Ukuran 64 dan *Epoch* 100

Epoch 100/100

17/17 [=====] - 37s 2s/step - loss:
0.0507 - accuracy: 0.9767 - val_loss: 0.0414 - val_accuracy:
1.0000

Waktu yang dibutuhkan: 62.7821 Menit

Untuk memudahkan dalam membacanya maka hasil diatas dirangkum dalam bentuk tabel seperti berikut:

Tabel 4.1 Pengaruh Ukuran Gambar dan Besar Nilai *Epoch*

Jumlah Epoch	Akurasi	Loss	Waktu Yang Dibutuhkan
Ukuran Gambar (64, 64)			
50	97%	0.05	32.8590 Menit
75	97.67%	0.05	49.3792 Menit
100	97.67%	0.05	62.7821 Menit
Ukuran Gambar (128, 128)			
50	98%	0.07	42.3757 Menit
75	98%	0.1	63.8148 Menit
100	94.16%	0.15	82.3399 Menit
Ukuran Gambar (256, 256)			
50	91%	0.24	83.3814 Menit
75	93.39%	0.17	125.9955 Menit
100	98%	0.04	164.3843 Menit

Berdasarkan tabel diatas dapat disimpulkan bahwa ukuran gambar dan besarnya nilai *epoch* sangat mempengaruhi pada hasil akurasi dan komputasi. Jika semakin besar ukuran gambar maka semakin lama komputasi nya. Pada proses *training* dapat dilihat tingkat akurasi paling tinggi dan loss paling rendah yaitu terdapat pada parameter ukuran gambar 128x128 dengan *epoch* 100. Hal ini dikarenakan jika ukuran gambar besar maka akan terdapat banyak fitur yang akan diambil begitu sebaliknya. Sedangkan *epoch* dalam sebuah jaringan CNN akan berpengaruh pada nilai akurasi, dikarenakan banyaknya batch yang akan diproses. Hal ini dapat dilihat ketika ukuran gambar yang sama dengan menggunakan nilai *epoch* yang berbeda maka terdapat perubahan di hasil akurasinya.

4.1.2 Hasil Proses *Testing*

Proses *testing* menggunakan dataset *testing* sebanyak 60 data, diantaranya setiap kelas berdasarkan warna kerabang telur terdapat 20 dataset telur dengan warna keraban coklat tua, 20 dataset telur dengan warna keraban coklat dan 20 dataset telur dengan warna keraban coklat muda. Saat melakukan proses *testing* akan menguji semua data *testing* satu per satu. Sehingga didapatkan hasil dari pengujian yang dilakukan bisa dilihat pada tabel di bawah.

Tabel 4.2 Hasil uji coba klasifikasi telur kerabang coklat tua

No.	Gambar	Data Asal	Prediksi Sistem	Hasil Prediksi	Kualitas
1.	0.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
2.	1.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
3.	2.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
4.	3.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
5.	4.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
6.	5.jpg	CT	CT	Benar	Penurunan Kualitas Rendah

7.	6.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
8.	7.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
9.	8.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
10.	9.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
11.	10.jpg	CT	C	Salah	-
12.	11.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
13.	12.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
14.	13.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
15.	14.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
16.	15.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
17.	16.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
18.	17.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
19.	18.jpg	CT	C	Salah	-
20.	19.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
21.	20.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
22.	21.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
23.	22.jpg	CT	CT	Benar	Penurunan Kualitas Rendah
24.	23.jpg	CT	CT	Benar	Penurunan Kualitas Rendah

Tabel 4.3 Hasil uji coba klasifikasi telur kerabang coklat

No.	Gambar	Data Asal	Prediksi Sistem	Hasil Prediksi	Kualitas
1.	0.jpg	C	C	Benar	Penurunan Kualitas Sedang
2.	1.jpg	C	C	Benar	Penurunan Kualitas Sedang
3.	2.jpg	C	C	Benar	Penurunan Kualitas Sedang
4.	3.jpg	C	C	Benar	Penurunan Kualitas Sedang
5.	4.jpg	C	C	Benar	Penurunan Kualitas Sedang
6.	5.jpg	C	C	Benar	Penurunan Kualitas Sedang
7.	6.jpg	C	C	Benar	Penurunan Kualitas Sedang
8.	7.jpg	C	C	Benar	Penurunan Kualitas Sedang
9.	8.jpg	C	C	Benar	Penurunan Kualitas Sedang
10.	9.jpg	C	C	Benar	Penurunan Kualitas Sedang
11.	10.jpg	C	CM	Salah	-

12.	11.jpg	C	C	Benar	Penurunan Kualitas Sedang
13.	12.jpg	C	C	Benar	Penurunan Kualitas Sedang
14.	13.jpg	C	CM	Salah	-
15.	14.jpg	C	C	Benar	Penurunan Kualitas Sedang
16.	15.jpg	C	C	Benar	Penurunan Kualitas Sedang
17.	16.jpg	C	C	Benar	Penurunan Kualitas Sedang
18.	17.jpg	C	CT	Salah	-
19.	18.jpg	C	C	Benar	Penurunan Kualitas Sedang
20.	19.jpg	C	C	Benar	Penurunan Kualitas Sedang
21.	20.jpg	C	C	Benar	Penurunan Kualitas Sedang
22.	21.jpg	C	C	Benar	Penurunan Kualitas Sedang
23.	22.jpg	C	C	Benar	Penurunan Kualitas Sedang
24.	23.jpg	C	C	Benar	Penurunan Kualitas Sedang

Tabel 4.4 Hasil uji coba klasifikasi telur kerabang coklat muda

No.	Gambar	Data Asal	Prediksi Sistem	Hasil Prediksi	Kualitas
1.	0.jpg	CM	C	Salah	-
2.	1.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
3.	2.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
4.	3.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
5.	4.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
6.	5.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
7.	6.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
8.	7.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
9.	8.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
10.	9.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
11.	10.jpg	CM	C	Salah	-
12.	11.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
13.	12.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
14.	13.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
15.	14.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
16.	15.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi

17.	16.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
18.	17.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
19.	18.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
20.	19.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
21.	20.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
22.	21.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi
23.	22.jpg	CM	C	Salah	-
24.	23.jpg	CM	CM	Benar	Penurunan Kualitas Tinggi

Berdasarkan gambar grafik di atas, dapat diketahui bahwa terdapat 2 data *image* yang salah pada kategori telur kerabang coklat tua, 3 data *image* yang salah pada kategori telur kerabang coklat dan 3 data *image* yang salah pada kategori telur kerabang coklat muda. Dalam mengidentifikasi penurunan kualitas telur berdasar warna kerabang, maka dari tabel di atas kemudian dimasukkan ke dalam tabel prediksi seperti di bawah.

Tabel 4.5 Hasil prediksi proses *testing*

Matriks		Prediksi		
		Coklat Tua	Coklat	Coklat Muda
<i>Actual Class</i>	Coklat Tua	18	2	0
	Coklat	3	17	0
	Coklat Muda	0	3	17

Berdasarkan tabel hasil prediksi di atas menunjukkan hasil yang baik. Dimana pada kelas coklat tua terdapat prediksi yang salah 2 data dari 20 data. Sedangkan pada kelas coklat dan coklat muda terdapat 3 data yang diprediksi salah. Selanjutnya untuk melakukan perhitungan akurasi menggunakan *confusion matrix* seperti berikut.

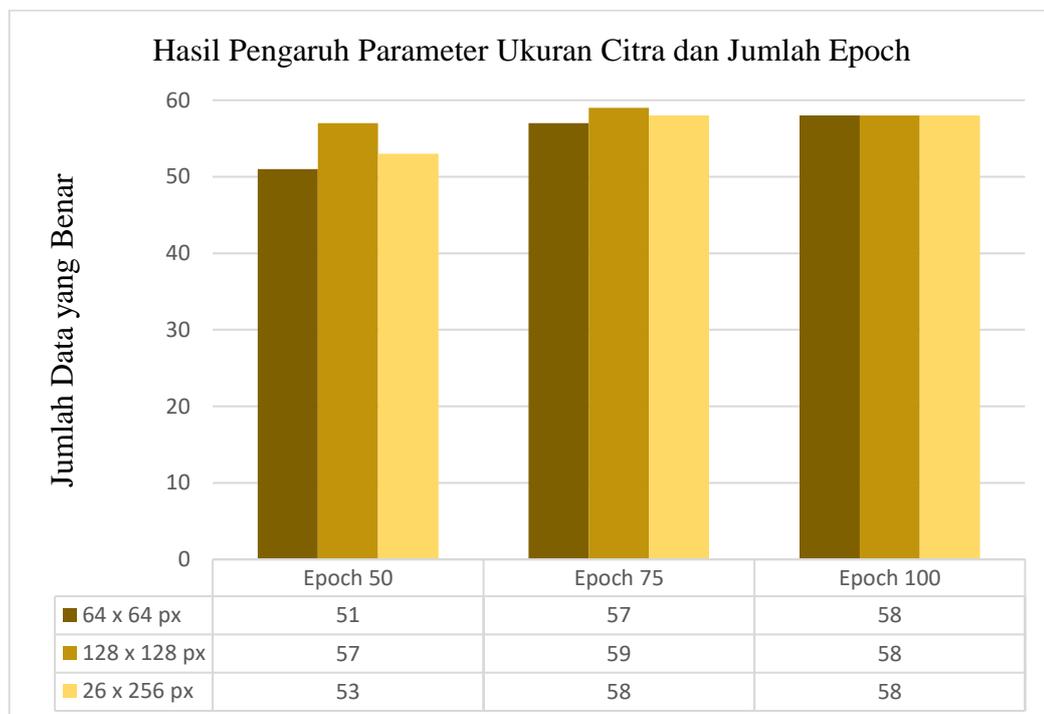
$$\text{Akurasi} = \frac{\text{Jumlah prediksi benar}}{\text{Jumlah Data}} \times 100\% \quad (4.1)$$

$$\text{Akurasi} = \frac{52}{60} \times 100\% = 88,89\% \quad (4.2)$$

Berdasarkan hasil dari perhitungan menggunakan *confussion matrix* didapatkan nilai akurasi klasifikasi penurunan kualitas telur dengan algoritma CNN sebesar 88,89%.

4.1.3 Pengaruh Parameter terhadap Akurasi

Diantara hal yang mempengaruhi tinggi rendahnya akurasi saat proses pembentukan model CNN adalah parameter *learning*. Parameter *learning* yang dimaksud adalah pengaruh jumlah *epoch* dan pengaruh ukuran inputan citra. Tujuan dari penentuan parameter model ini agar mengetahui model mana yang paling terbaik dengan memperhatikan nilai parameternya. Berikut grafik hasil pengujian berdasarkan parameter ukuran citra (64x64, 128x128, 256x256) dan jumlah *epoch* (50, 75, 100).



Gambar 4. 10 Gambar grafik Hasil Perbandingan Parameter

Berdasarkan grafik di atas menunjukkan bahwa pengujian dengan parameter ukuran citra 128 x 128 piksel dengan jumlah *epoch* 75 menghasilkan jumlah pengujian data yang benar paling banyak. Selanjutnya berikut penjelasan untuk mengetahui pengaruh parameter yang digunakan terhadap hasil akurasi dapat dilihat di bawah:

1) Pengaruh Jumlah *Epoch*

Dalam melakukan proses *training*, ketika seluruh dataset sudah melalui proses *training* pada *neural network*, kemudian dikembalikan ke awal dalam satu putaran itulah disebut dengan *epoch*. Dimana jumlah *epoch* inilah akan mempengaruhi nilai akurasi pada saat proses *training*. Berikut tabel pengaruh jumlah *epoch* terhadap akurasi.

Tabel 4.6 Pengaruh *epoch*

<i>Epoch</i>	Akurasi	<i>Loss value</i>	Waktu
50	95.3%	0.12	1 jam 58 menit
75	95%	0.12	2 jam 39 menit
100	97,8%	0.06	3 jam 9 menit

Berdasarkan tabel di atas, didapatkan akurasi yang paling baik terdapat pada *epoch* 100 yaitu 97.8%. Jika dilihat dari tabel tersebut, ketika *epoch* ditambah 75, maka akurasi menurun dan nilai *loss* juga bertambah.

2) Pengaruh Ukuran Citra

Selain jumlah *epoch* yang mempengaruhi nilai akurasi, terdapat parameter ukuran dari citra juga dapat mempengaruhi nilai akurasi yang dijadikan inputan saat proses *training*. Karena semakin besar ukuran pada citra, semakin banyak nilai piksel yang perlu dilatih. Hal ini juga menyebabkan meningkatnya komputasi sehingga membuat waktu pemrosesan semakin lama. Berikut tabel pengaruh ukuran citra terhadap akurasi.

Tabel 4.7 Pengaruh ukuran citra

Ukuran Citra	Akurasi	Loss value	Waktu
64 x 64	96%	0.5	49.3401 menit
128 x 128	97.4%	0.1	62.8434 menit
256 x 256	94.2%	0.15	1 jam 24 menit

Berdasarkan tabel di atas, didapatkan akurasi yang paling baik terdapat pada ukuran citra 128x128 yaitu 97.4%.

3) Pengaruh Jumlah Data

Pengaruh parameter selanjutnya berdasarkan perbandingan jumlah dataset yang digunakan sebagai data *training* dan data *testing*. Berikut tabel hasil pengaruh perbandingan jumlah dataset.

Tabel 4.8 Pengaruh Jumlah Data

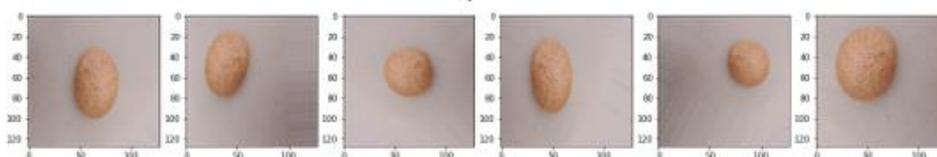
Jumlah Data	Akurasi	Loss value	Waktu
60% : 40%	95.3%	0.12	1 jam 58 menit
70% : 30%	95%	0.12	2 jam 39 menit
80% : 20%	97.8%	0.06	3 jam 9 menit

Berdasarkan tabel di atas, didapatkan akurasi yang paling baik terdapat pada perbandingan jumlah dataset yaitu 80% sebagai data *training* dan 20% sebagai data *testing* dengan hasil tingkat akurasi sebesar 97.8%.

4.2 Integrasi

Berdasarkan hasil proses uji coba sistem untuk klasifikasi penurunan kualitas telur ayam ras berdasarkan warna kerabang menggunakan algoritma *Convolutional Neural Network* (CNN) menunjukkan hasil akurasi sebesar 88,89%. Dimana nilai akurasi menunjukkan seberapa dekat telur ayam ras diidentifikasi dengan nilai benar berdasarkan kategori yang ditentukan. Dapat diketahui, dari 72 dataset telur yang diuji, sistem mampu mengklasifikasi penurunan kualitas telur ayam tersebut sebesar 64 dataset telur.

Terdapat beberapa faktor yang menunjukkan terjadinya kesalahan dalam mengklasifikasi telur ayam ras berdasarkan warna kerabang, sehingga nilai tingkat akurasi sistem belum maksimal di antaranya, ukuran dataset *training* masih terlalu kecil dan tidak mengandung sampel data yang cukup untuk secara akurat dalam mempresentasikan semua kemungkinan nilai data masukan sehingga menyebabkan terjadinya *overfitting*. *Overfitting* bisaanya akan menangkap data *noise* yang seharusnya diabaikan. Suatu model yang mengalami *overfitting* memiliki nilai *low loss* dan akurasi yang rendah. Sementara suatu model yang dinyatakan baik adalah model yang mampu menjelaskan data tanpa menghiraukan *noise* data tersebut dan memiliki nilai *loss* rendah dan nilai akurasi yang tinggi. Jika suatu model mengalami *overfitting* akan sangat mempengaruhi optimalnya hasil akurasi saat dilakukan proses uji coba. Oleh karena itu, pada penelitian ini peneliti menambahkan proses *augmentasi data* pada tahap *preprocessing* untuk memperbanyak variasi data dengan tujuan untuk menghindari terjadinya *overfitting*. Berikut hasil *preprocessing* menggunakan teknik *augmentasi data*.

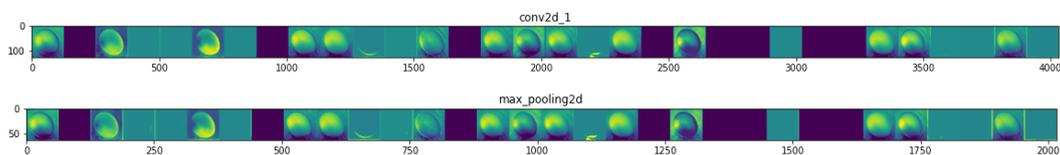


Gambar 4.11 Hasil *Preprocessing* menggunakan teknik *augmentasi*

Hasil dari *preprocessing* nanti nya digunakan sebagai inputan dengan 3 ukuran yang berbeda seperti yang telah dijelaskan pada bab sebelumnya. Peneliti menemukan dengan penggunaan *input shape* berukuran 128 x 128 piksel, ukuran filter 3x3 dengan *epoch* 75 menghasilkan nilai akurasi yang sangat baik sebesar 94%. Hal ini yang menunjukkan pengaruh ukuran citra dengan jumlah *epoch* sangat berpengaruh ketika dilakukan proses *training*. Penggunaan ukuran filter 3 x 3 pada

penelitian ini berdasarkan penelitian sebelumnya bahwa untuk penentuan ukuran ketebalan suatu filter selalu mengikuti ketebalan dari gambar yang digunakan sebagai *input*, dan pada umumnya tinggi dan lebar sebuah filter berukuran ganjil. Karena dengan ukuran ganjil mampu memberikan representasi yang lebih baik karena jumlah bagian kiri dan kanan seimbang.

Kemampuan algoritma CNN dalam melakukan klasifikasi gambar sebenarnya sudah cukup baik. Cara kerja CNN agar bisa memahami dan melakukan tindakan lebih lanjut agar model mampu bekerja lebih baik adalah melalui visualisasi terhadap *feature map*. Tujuan dari memvisualisasikan *feature map* adalah agar mampu memahami apa saja fitur dari *input* yang terdeteksi oleh model. Pada umumnya, semakin dalam *layer* maka fitur yang didapat semakin umum. Jadi pada *layer* pertama setelah *input*, fitur yang didapat sangat detail. Berikut hasil dari visualisasi *feature map* yang dilakukan pada penelitian ini.



Gambar 4.12 Hasil Visualisasi *feature map*

Berdasarkan gambar di atas, dapat dilihat bahwa tingkat kesalahan yang lebih kecil berbanding lurus dengan fitur – fitur yang lebih tajam yang berpengaruh pada hasil klasifikasi di akhir. Hal ini menunjukkan bahwa pentingnya untuk belajar dan mempelajari sesuatu secara mendalam. Ketika algoritma tidak mampu mempelajari suatu objek dengan baik, maka hasil yang didapatkan kurang baik. Sama halnya dengan manusia, ketika tidak mau belajar dan mempelajari sesuatu yang baru maka kemampuan kita akan *stuck* disitu saja.

Hasil akurasi yang didapatkan merupakan bentuk dari aplikasi pengklasifikasian terhadap kualitas pada citra telur berdasarkan warna kerabang telur. Dalam hal ini terdapat 2 mu'amalah dalam islam yang memiliki peranan sangat penting, karena muamalah berisi tentang aturan – aturan dan hokum yang sesuai dengan syariat islam yang mengatur tentang urusan dunia dan di akhirat kelak. Diantaranya 2 muamalah tersebut yaitu:

1. Muamalah Ma'a Allah

Pada aplikasi ini dalam melakukan pengklasifikasian terhadap kualitas telur sangatlah penting, karena untuk menjaga kesehatan tubuh pada dasarnya merupakan suatu kewajiban bagi setiap muslim yaitu dengan ikhtiar mengkonsumsi makanan dan minuman yang halal dan *thayyib*, sebagaimana telah dijelaskan dalam Al-Qur'an Surat Al - Baqarah ayat 168 yang berbunyi:

يَا أَيُّهَا النَّاسُ كُلُوا مِمَّا فِي الْأَرْضِ حَلَالًا طَيِّبًا وَلَا تَتَّبِعُوا خُطُوَاتِ الشَّيْطَانِ إِنَّهُ لَكُمْ عَدُوٌّ مُّبِينٌ

Artinya: "*Wahai manusia! Makanlah dari (makanan) yang halal dan thayyib (baik) yang terdapat di bumi, dan janganlah kamu mengikuti langkah - langkah setan. Sungguh setan itu musuh yang nyata bagimu.*" (QS. Al-Baqarah: 168).

Menurut penjelasan ayat diatas, pada dasarnya menjaga kesehatan tubuh dengan mengkonsumsi makanan dan minuman yang halal dan *thayyib* adalah sebuah kewajiban bagi setiap muslim, apalagi di era pandemi Covid-19 saat ini. Menurut Islam, makanan yang halal dan *thayyib* adalah makanan dan minuman yang dianjurkan untuk dikonsumsi. Halal menurut pemahaman fiqih adalah halal dari segi zatnya dan segi prosesnya. Sedangkan makanan yang disebut *thayyib* adalah apabila makanan tersebut dalam keadaan baik dan aman, sehingga jika dikonsumsi dapat memberi manfaat bagi tubuh manusia.

2. Muamalah Ma'a Naas

Seiring berkembangnya zaman dapat mempengaruhi umat manusia untuk mempelajari ilmu sains dan teknologi yang kemudian dapat dimanfaatkan dalam berbagai aspek kehidupan. Menurut ajaran islam, belajar merupakan sebuah kewajiban untuk setiap umat manusia, seperti halnya telah tertulis dalam ayat Al-Qur'an surat Al 'Alaq ayat 1 – 5 yang berbunyi:

اِقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ (۱) خَلَقَ الْإِنْسَانَ مِنْ عَلَقٍ (۲) اِقْرَأْ وَرَبُّكَ الْأَكْرَمُ (۳) الَّذِي عَلَّمَ بِالْقَلَمِ
(۴) عَلَّمَ الْإِنْسَانَ مَا لَمْ يَعْلَمُ (۵)

Artinya: “*Bacalah dengan (menyebut) nama TuhanMu Yang Menciptakan, Dia telah menciptakan manusia dari segumpal darah. Bacalah, dan TuhanMulah Yang Maha Pemurah, Yang mengajar (manusia) dengan perantara kalam. Dia mengajarkan kepada manusia apa yang tidak diketahuinya.*” (QS. Al-‘Alaq: 1-5).

Berdasarkan penjelasan ayat diatas yaitu dianjurkan bagi umat manusia untuk mempelajari tentang banyak hal dengan membaca. Bacaan yang dimaksud pada ayat tersebut adalah bacaan dengan berbagai objek, salah satunya mengenai sains dan teknologi. Selain itu, belajar juga merupakan salah satu ikhtiar bagi setiap manusia untuk mencapai tujuan sesuai yang diinginkan. Seperti halnya manusia, sebuah sistem klasifikasi juga perlu adanya melakukan pembelajaran melalui proses training dan pembelajaran fitur dari sebuah citra, sebagaimana telah tertulis dalam Al-qur'an surat Ar-Ra'd ayat 11 yang berbunyi:

لَهُ مَعْقِبَاتٌ مِنْ بَيْنِ يَدَيْهِ وَمِنْ خَلْفِهِ ۖ يَحْفَظُونَهُ ۗ مِنْ أَمْرِ اللَّهِ إِنَّ اللَّهَ لَا يُعَيِّرُ مَا بِقَوْمٍ حَتَّىٰ يُعَيِّرُوهُمَا
بِأَنفُسِهِمْ ۗ وَإِذَا أَرَادَ اللَّهُ بِقَوْمٍ سُوءًا فَلَا مَرَدَّ لَهُ ۗ وَمَا لَهُمْ مِنْ دُونِهِ مِن وَّالٍ

Artinya: “*Baginya (manusia) ada malaikat – malaikat yang selalu menjaganya bergiliran, dari depan dan belakangnya. Mereka menjaganya atas perintah Allah SWT. Sesungguhnya Allah tidak akan mengubah keadaan suatu kaum sebelum mereka mengubah keadaan yang ada pada diri mereka sendiri. Dan apabila Allah menghendaki keburukan terhadap suatu kaum, maka tidak ada yang dapat menolaknya dan tidak ada pelindung bagi mereka selain Dia.*” (QS. Ar-Ra'd: 11).

Berdasarkan ayat tersebut telah dijelaskan bahwa hasil yang diperoleh sangat berhubungan dengan usaha yang dilakukannya. Semakin banyak usaha yang dilakukan maka hasil yang diperoleh akan optimal, seperti halnya suatu sistem yang mempelajari banyak data, maka hasil yang didapatkan juga semakin maksimal.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan untuk mengidentifikasi penurunan kualitas telur ayam ras berdasarkan warna kerabang telur menggunakan algoritma *Convolutional Neural Network* (CNN), maka dapat diperoleh kesimpulan bahwa penggunaan algoritma *Convolutional Neural Network* (CNN) tersebut cukup baik dalam melakukan klasifikasi gambar telur ayam ras berdasarkan warna kerabang, diantaranya:

- 1) Pada tahap hasil uji model *training* menggunakan inputan berukuran 128 x128 piksel dengan nilai *epoch* sebesar 75 akan diujikan dengan dataset *testing* yang berjumlah 72 data. Hasil akurasi yang didapatkan sebesar 88.89 % dengan jumlah kesalahan sebesar 8 dari jumlah dataset yang diujikan.
- 2) Selanjutnya dari model awal pada kesimpulan pertama dalam penelitian ini, dikembangkan lagi dengan mengganti ukuran gambar menjadi 3 ukuran diantaranya (64x64, 128x128 dan 256x256) piksel, dengan masing – masing menggunakan parameter *epoch* (50, 75 dan 100). Hasil yang didapatkan menyimpulkan bahwa pengaruh nilai parameter tersebut terhadap model CNN yang dibangun sangat mempengaruhi hasil akurasi dan waktu yang dibutuhkan algoritma CNN dalam proses *training* dan mengklasifikasi kualitas telur berdasarkan warna kerabang.

5.2 Saran

Dalam pengembangan sistem klasifikasi kualitas telur ayam berdasarkan warna kerabang menggunakan algoritma CNN pada penelitian ini, maka perlu diperhatikan beberapa hal diantaranya:

- 1) Memperbaiki kualitas foto telur yang diambil dan memperbanyak lagi jumlah dataset yang digunakan agar algoritma CNN mampu bekerja dengan maksimal, sehingga menghasilkan nilai akurasi yang lebih baik lagi.
- 2) Perlu dikembangkan lagi dengan memodifikasi arsitektur jaringan CNN yang digunakan, agar dapat dibandingkan hasil model yang terbaik berdasarkan pengaruh dari arsitektur yang dibangun.

DAFTAR PUSTAKA

- Arif, Y. M., & Nurhayati, H. (2022). Learning Material Selection for Metaverse-Based Mathematics Pedagogy Media Using Multi-Criteria Recommender System. *International Journal of Intelligent Engineering and Systems*, 15(6), 541–551. <https://doi.org/10.22266/ijies2022.1231.48>
- Arif, Y. M., Nurhayati, H., Kurniawan, F., Nugroho, S. M. S., & Hariadi, M. (2020). Blockchain-Based Data Sharing for Decentralized Tourism Destinations Recommendation System. *International Journal of Intelligent Engineering and Systems*, 13(6), 472–486. <https://doi.org/10.22266/ijies2020.1231.42>
- Arrofiqoh, E. N., & Harintaka, H. (2018). Implementasi Metode Convolutional Neural Network Untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi. *Geomatika*, 24(2), 61. <https://doi.org/10.24895/jig.2018.24-2.810>
- Bun, K., Informatika, J. T., Industri, F. T., Gunadarma, U., & Barat, J. (2018). *IDENTIFIKASI TELUR RETAK MENGGUNAKAN METODE JARINGAN SYARAF TIRUAN BERDASARKAN TEKSTUR TELUR*. 183–192.
- Geng, L., Liu, H., Xiao, Z., Yan, T., Zhang, F., & Li, Y. (2020). Hatching egg classification based on CNN with channel weighting and joint supervision. *Multimedia Tools and Applications*, 79(21–22), 14389–14404. <https://doi.org/10.1007/s11042-018-6784-9>
- <https://www.clarifai.com/blog/classification-vs-detection-vs-segmentation-models-the-differences-between-them-and-how-each-impact-your-results> (diakses pada tanggal 29 November 2021)
- <https://medium.com/@samuelsena/pengenalan-deep-learning-part-7-convolutional-neural-network-cnn-b003b477dc94> (diakses pada tanggal 7 Oktober 2021)
- <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53> (diakses pada tanggal 6 februari 2022)
- IBRAHIM, N., SA'IDAH, S., HIDAYAT, B., & DARANA, S. (2022). Klasifikasi Grade Telur Ayam Negeri secara non- Invasive menggunakan Convolutional Neural Network. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 10(2), 297. <https://doi.org/10.26760/elkomika.v10i2.297>
- Ibrahim, R., Zin, Z. M., Nadzri, N., Shamsudin, M. Z., & Zainudin, M. Z. (2012). Egg's grade classification and dirt inspection using image processing techniques. *Lecture Notes in Engineering and Computer Science*, 2198, 1179–1182.
- Kelleher, J.D. 2019. Deep Learning. MIT press.

- Lee, C. Y., Gallagher, P. W., & Tu, Z. (2016). Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree. *Proceedings of the 19th tinju472*.
- Maimunah, & Rokhman, T. (2018). Klasifikasi Penurunan Kualitas Telur Ayam Ras Berdasarkan Warna Kerabang Menggunakan Support Vector Machine. *Informatics for Educators and Professionals*, 3(1), 43–52.
- Park, Y. S., & Lek, S. (2016). Artificial Neural Networks: Multilayer Perceptron for Ecological Modeling. In *Developments in Environmental Modelling* (Vol. 28). Elsevier. <https://doi.org/10.1016/B978-0-444-63623-2.00007-4>
- Patil, A., & Rane, M. (2021). Convolutional Neural Networks: An Overview and Its Applications in Pattern Recognition. *Smart Innovation, Systems and Technologies*, 195, 21–30. https://doi.org/10.1007/978-981-15-7078-0_3
- Shimizu, R., Yanagawa, S., Shimizu, T., Hamada, M., & Kuroda, T. (2018). Convolutional neural network for industrial egg classification. *Proceedings - International SoC Design Conference 2017, ISOCC 2017*, 67–68. <https://doi.org/10.1109/ISOCC.2017.8368830>
- Thipakorn, J., Waranusast, R., & Riyamongkol, P. (2017). Egg weight prediction and egg size classification using image processing and machine learning. *ECTI-CON 2017 - 2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, 477–480. <https://doi.org/10.1109/ECTICon.2017.8096278>
- Wijaya, T. A., & Prayudi, Y. (2010). Implementasi Visi Komputer Dan Segmentasi Citra. *Snati* 2010, 2010(Snati), 1–5. <http://journal.uui.ac.id/index.php/Snati/article/view/1949/1724>
- Xu, J., Zhang, Y., & Miao, D. (2020). Three-way confusion matrix for classification: A measure driven view. *Information Sciences*, 507(xxxx), 772–794. <https://doi.org/10.1016/j.ins.2019.06.064>
- Yang, H. M., Wang, Z. Y., & Lu, J. (2009). Study on the relationship between eggshell colors and egg quality as well as shell ultrastructure in Yangzhou chicken. *African Journal of Biotechnology*, 8(12), 2898–2902.