

**SISTEM PENGENALAN DAN TRANSLITERASI AKSARA LAMPUNG  
BERBASIS PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

**(Skripsi)**

**Oleh**

**DESI RAHMA UTAMI  
NPM 1815031043**



**FAKULTAS TEKNIK  
JURUSAN TEKNIK ELEKTRO  
UNIVERSITAS LAMPUNG  
2024**

## ABSTRAK

### SISTEM PENGENALAN DAN TRANSLITERASI AKSARA LAMPUNG BERBASIS PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* (CNN)

Oleh

**DESI RAHMA UTAMI**

Era digital mempunyai potensi kebermanfaatan namun juga kerugian dalam proses pengembangannya. Pada bidang sosial budaya, salah satu pengaruh, ialah menimbulkan perubahan pada pola interaksi antar masyarakat terkhusus dalam penggunaan bahasa dan aksara daerah yang dinilai kuno dan tertinggal. Salah satu daerah yang mempunyai aksara ialah provinsi Lampung. Menyikapi hal tersebut, perlu adanya teknologi informasi yang mampu menghilangkan stigma kuno dan tertinggal tanpa menghilangkan nilai luhur dari suatu budaya. Salah satu fokus ilmu yang dinilai mampu melakukan penerapan teknologi yang mampu melakukan proses mengenali dan mentransliterasi data citra aksara Lampung dalam kata, ialah *Convolution Neural Network* (CNN). Sistem pengenalan dan transliterasi aksara Lampung dengan model pembelajaran CNN ini memiliki tingkat akurasi pengujian sebesar 80 %.

Penelitian ini menggunakan perangkat lunak *Code editor Visual Studio Code* dengan Bahasa pemrograman *Python 3.9.4 64 bit*, 1732 masukan data citra yang terbagi dalam 4 kelas (*output*) dengan masing-masing 423 citra untuk membangun sistem dan 10 citra untuk menguji sistem pada masing-masing katanya, dengan fungsi optimasi *Root Mean Square Propagation (RMSProp)*, jenis *pooling* berupa *max pooling*, *dropout* 0.01, *features extraction* (pengenalan) sebanyak 4 layer dan *classification* (transliterasi) sebanyak 3 layer.

**Kata kunci** : Aksara Lampung, Pengenalan, Transliterasi, CNN, Data Citra

## **ABSTRACT**

### **SISTEM PENGENALAN DAN TRANSLITERASI AKSARA LAMPUNG BERBASIS PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

**Oleh**

**DESI RAHMA UTAMI**

The digital era has potential benefits but also losses in the development process. In the socio-cultural field, one of the influences is causing changes in interaction patterns between communities, especially in the use of regional languages and scripts which are considered ancient and underdeveloped. One area that has script is Lampung province. In response to this, there is a need for information technology that is able to eliminate ancient and backward stigmas without eliminating the noble values of a culture. One of the scientific focuses that is considered capable of implementing technology that is capable of carrying out the process of recognizing and transliterating Lampung script image data into words is Convolution Neural Network (CNN). The Lampung script recognition and transliteration system with the CNN learning model has a testing accuracy rate of 80%.

This study uses Visual Studio Code editor software with Python 3.9.4 64 bit programming language, 1732 image data inputs divided into 4 classes (output) with 423 images each to build the system and 10 images to test the system on each word, with the Root Mean Square Propagation (RMSProp) optimization function, pooling type in the form of max pooling, dropout 0.01, features extraction (recognition) as many as 4 layers and classification (transliteration) as many as 3 layers.

**Keyword** : Lampung Script, Recognition, Transliteration, CNN, Image Data

**SISTEM PENGENALAN DAN TRANSLITERASI AKSARA LAMPUNG  
BERBASIS PENERAPAN *CONVOLUTIONAL NEURAL NETWORK* (CNN)**

Oleh

**DESI RAHMA UTAMI**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Mencapai Gelar  
SARJANA TEKNIK**

Pada

**Jurusan Teknik Elektro  
Fakultas Teknik**



**FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2024**

**Judul Skripsi** : **SISTEM PENGENALAN DAN  
TRANSLITERASI AKSARA LAMPUNG  
BERBASIS PENERAPAN  
CONVOLUTIONAL NEURAL NETWORK  
(CNN)**

**Nama Mahasiswa** : **Desi Rahma Utami**

**Nomor Pokok Mahasiswa** : **1815031043**

**Jurusan** : **Teknik Elektro**

**Fakultas** : **Teknik**



**Umi Murdika, S.T., M.T.**  
NIP. 197202062005012002

**Syaiful Alam, S.T., M.T.**  
NIP. 196904161998031004

## 2. Mengetahui

**Ketua Jurusan  
Teknik Elektro**

**Herlinawati, S.T., M.T.**  
NIP. 197103141999032001

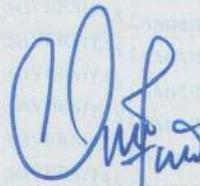
**Ketua Program Studi  
Teknik Elektro**

**Sunadi, S.T., M.T.**  
NIP. 197311042000031001

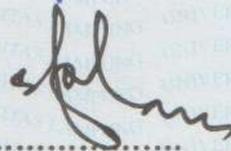
**MENGESAHKAN**

**1. Tim Penguji**

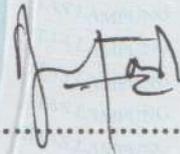
**Ketua : Umi Murdika, S.T., M.T.**



**Sekretaris : Syaiful Alam, S.T., M.T.**



**Penguji Utama : Sumadi, S.T., M.T.**



**2. Dekan Fakultas Teknik**



**Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. )**  
NIP 19750928 2001 12 1 002



**Tanggal Lulus Ujian Skripsi : 29 Juli 2024**

## SURAT PERNYATAAN KEASLIAN HASIL KARYA

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya dengan judul "Sistem Pengenalan dan Transliterasi Aksara Lampung Berbasis Penerapan *Convolutional Neural Network* (CNN)" merupakan hasil karya saya sendiri dan bukan hasil karya orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila pernyataan saya tidak benar dan dikemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia dikenai sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 20 Agustus 2024



Desi Rahma Utami  
NPM. 1815031043

## RIWAYAT HIDUP



Penulis dilahirkan di Pedamaran pada tanggal 06 Desember 2000, sebagai anak pertama dari tiga bersaudara. Ayah bernama Deka Meriadi dan Ibu bernama Evi Susanti (Alm). Penulis menyelesaikan pendidikan pertama mulai dari Sekolah dasar (SD) di SDN 11 Kota Jambi pada tahun 2012, Sekolah Menengah Pertama (SMP) di SMP PGRI Pedamaran pada tahun 2015, dan Sekolah Menengah Atas (SMA) di SMAN 1 Pedamaran pada tahun 2018.

Pada tahun 2018, penulis terdaftar sebagai mahasiswa jurusan Teknik Elektro Fakultas Teknik melalui jalur SBMPTN. Pada tahun 2020, Penulis bersama tim mendapatkan pendanaan Program Kreativitas Mahasiswa (PKM) jenis PKMKC dengan judul “Aplikasi Kehidupan : Pengembangan Karakter dan Impian Berbasis Android”. Pada rentang waktu 09 Agustus – 10 September 2021, penulis melaksanakan Kerja Praktik di Workshop Sintelis (Sinyal, Telekomunikasi dan Listrik) PT Kereta Api Indonesia (Persero) Divre IV Tanjung Karang. Penyelesaian Kerja Praktik tersebut menghasilkan sebuah laporan dengan judul “Analisis Penggunaan *Interface Block Module* (IBM) dalam Sistem Komunikasi Blok di PT. Kereta Api Indonesia (Persero) Devisi Regional IV Tanjung Karang”.

Selama menjadi mahasiswa, penulis aktif mengikuti beberapa organisasi kemahasiswaan, antara lain:

1. Menjadi Anggota Himpunan Mahasiswa Teknik Elektro (HIMATRO) Devisi Kerohanian periode 2019 dan 2020.
2. Menjadi Sekretaris Biro Kesekretariatan Forum Silaturahmi dan Studi Islam Fakultas Teknik (FOSSI-FT) periode 2019.
3. Menjadi Staff Ahli Kementerian Kajian dan Strategis (Kastrat) BEM U KBM Unila dan tergabung kedalam tim kajian pemilihan Rektor periode 2019
4. Menjadi Sekretaris Panitia Khusus (Pansus) Pemilihan Raya (PEMIRA) Fakultas Teknik Periode 2019
5. Menjadi Sekretaris Bidang Perempuan Kesatuan Aksi Mahasiswa Muslim Indonesia Komisariat Lampung (KAMMILA) periode 2019/2020.
6. Menjadi Wakil Ketua Umum Forum Silaturahmi dan Studi Islam Fakultas Teknik (FOSSI-FT) periode 2020..
7. Menjadi Wakil Ketua 1 Dewan Perwakilan Mahasiswa Fakultas Teknik (DPM-FT) periode 2021.
8. Menjadi Bendahara Umum Kesatuan Aksi Mahasiswa Muslim Indonesia Komisariat Lampung (KAMMILA) periode 2021.

## SANWACANA

### *Bismillahirrohmannirrohim.*

Puji syukur penulis ucapkan pada Allah Subhanahu wata'ala atas segala limpahan rahmat dan nikmat yang diberikan-Nya sehingga penulis dapat menyelesaikan skripsi ini. Shalawat beserta salam senantiasa dicurahkan kepada Nabi Muhammad Shallallahu Alaihi Wassalaam. Suri teladan terbaik hingga akhir zaman.

Skripsi dengan judul “Sistem Pengenalan dan Transliterasi Aksara Lampung Berbasis Penerapan *Convolutional Neural Network* (CNN)” ini merupakan salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Jurusan Teknik Elektro Fakultas Teknik Universitas Lampung.

Selama mengerjakan Skripsi ini, penulis mendapatkan bantuan pemikiran maupun dorongan moril dari berbagai pihak. Oleh karena itu dalam kesempatan ini, penulis ingin sampaikan rasa terimakasih kepada :

1. Ibu Prof. Dr. Ir. Lusmeilia Afriani, D.E.A., I.P.M. selaku Rektor Universitas Lampung.
2. Bapak Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. selaku Dekan Fakultas Teknik Universitas Lampung
3. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
4. Bapak Ir. Meizano Ardhi Muhammad, S.T., M.T. selaku Sekretaris Jurusan Teknik Elektro Universitas Lampung.
5. Bapak Sumadi, S.T., M.T. selaku Ketua Program Studi Teknik Elektro Universitas Lampung dan selaku penguj utama. Terima kasih atas seluruh

masukan dan arahan sehingga skripsi ini menjadi lebih baik dan mempersiapkan diri menjadi seorang Sarjana Teknik.

6. Ibu Umi Murdika, S.T., M.T. selaku Pembimbing Utama yang telah memberikan bimbingan, arahan, motivasi dan kesabaran dalam membimbing penulis dalam penyelesaian skripsi ini dan mempersiapkan diri menjadi seorang Sarjana Teknik.
7. Bapak Syaiful Alam, S.T., M.T. selaku Pembimbing Pendamping yang telah memberikan bimbingan, arahan, dan motivasi dalam membimbing penulis dalam penyelesaian skripsi ini dan mempersiapkan diri menjadi seorang Sarjana Teknik.
8. Ibu Yetti Yuniarti, S.T., M.T. selaku Dosen Pembimbing Akademik (PA) yang telah memberikan bimbingan, nasehat dan motivasi bagi penulis untuk kembali semangat dalam menyelesaikan studi dan mempersiapkan diri menjadi seorang Sarjana Teknik.
9. Segenap dosen di Jurusan Teknik Elektro yang telah memberikan ilmu dan pengalaman yang bermanfaat dan luas bagi penulis.
10. Segenap staff di Jurusan Teknik Elektro dan Fakultas Teknik yang telah membantu penulis baik dalam hal administrasi dan hal-hal lainnya.
11. Ibu Rusnon sebagai nenek yang telah memberikan doa, kasih sayang, dan dukungannya sampai saat ini.
12. Bapak Deka Meriadi, Ibu Karni Yanti, Bapak Herman dan Ibu Asmi selaku orang tua yang telah memberikan doa, kasih sayang, dan dukungannya sampai saat ini.
13. Adikku David Andriansyah dan Dea Putri Evika Sari yang telah memberikan doa dan dukungan sampai saat ini.
14. Ustadz Amir, Tete Risky, dan Mbak Ayu yang telah memonitor, memberikan arahan dan motivasi dalam perkembangan diri penulis baik dalam akademik maupun non akademik.
15. Diah, Elci, Mommy, Annisa, Tuti, Dira, Salma, Firra, Mile, dan si bontot Ami yang telah banyak memberikan bantuan, motivasi dan dukungan bagi penulis selama proses studi dan penyelesaian skripsi ini.

16. Nadia dan Bunga yang telah memberikan bantuan, motivasi dan dukungan bagi penulis selama proses pembelajaran terutama dalam pemikiran dan analisis kasus.
17. Rekan-rekan Mahasiswa Teknik Elektro Angkatan 2018 Unila yang selalu memberikan dukungan dan semangat.
18. Semua pihak yang tidak dapat disebut satu persatu yang telah membantu serta mendukung penulis dari awal kuliah sampai dengan terselesainya Skripsi ini.

Penulis menyadari bahwa Skripsi ini masih jauh dari sempurna dikarenakan terbatasnya pengetahuan dan pengalaman yang dimiliki penulis, oleh karena itu penulis mengharapkan kritik dan saran yang membangun dari berbagai pihak. Semoga Skripsi ini dapat berguna dan bermanfaat bagi para pembaca dan semua pihak khususnya dalam bidang Teknik Elektro.

Bandar Lampung,      Agustus 2024  
Penulis,

Desi Rahma Utami  
NPM. 1815031043

## DAFTAR ISI

	Halaman
<b>ABSTRAK</b> .....	ii
<b>ABSTRACT</b> .....	iii
<b>HALAMAN JUDUL</b> .....	iv
<b>LEMBAR PERSETUJUAN</b> .....	v
<b>LEMBAR PENGESAHAN</b> .....	vi
<b>LEMBAR PERNYATAAN</b> .....	vii
<b>RIWAYAT HIDUP</b> .....	viii
<b>SANWACANA</b> .....	x
<b>DAFTAR TABEL</b> .....	xv
<b>DAFTAR GAMBAR</b> .....	xvii
<b>BAB I. PENDAHULUAN</b> .....	1
1.1 Latar Belakang .....	1
1.2 Tujuan Penelitian .....	3
1.3 Manfaat Penelitian .....	3
1.4 Rumusan Masalah .....	4
1.5 Batasan Masalah .....	4
1.6 Sistematika Penulisan .....	4
<b>BAB II. TINJAUAN PUSTAKA</b> .....	6
2.1 <i>State of The Art</i> .....	6
2.2 Aksara Lampung .....	7
2.2.1 Tipografi dan Anatomi Aksara Lampung .....	9
2.3 Citra .....	14
2.4 Pengolahan Citra Digital .....	16
2.5 Tahapan Pengolahan Citra .....	17
2.5.1. Akuisisi Data Citra .....	17
2.5.2. <i>Preprocessing</i> Data .....	18
2.6 <i>Convolution Neural Network</i> (CNN) .....	20
2.6.1. Gambaran Umum Arsitektur CNN .....	20
2.6.2. <i>Convolutional Layer</i> .....	22
2.6.3. <i>Rectifier Linear Unit (ReLU) Layer</i> .....	23
2.6.4. <i>Pooling Layer</i> .....	25
2.6.5. <i>Fully Connected Layer</i> (Layer Klasifikasi) .....	27
2.6.6. Fungsi Aktivasi .....	30

2.7 Tahapan pembuatan sistem .....	33
2.7.1. Membangun sistem .....	33
2.7.2. Menguji dan mengevaluasi Sistem .....	41
<b>BAB III. METODE PENELITIAN</b> .....	44
3.1 Waktu dan Tempat penelitian .....	44
3.2 Alat dan Bahan .....	44
3.3 Tahapan dan Metode Penelitian .....	44
3.3.1 Akuisisi <i>Dataset</i> .....	45
3.3.2. <i>Pre-processing Data</i> .....	49
3.3.3 Pembuatan Model Arsitektur CNN .....	51
3.3.4. Melatih Sistem .....	92
3.3.5. Pengujian Sistem .....	93
3.3.6 Analisis dan Pembahasan .....	93
3.4 Diagram Alir Penelitian .....	93
3.5 Diagram Alir Sistem Perancangan .....	95
<b>BAB IV HASIL DAN PEMBAHASAN</b> .....	99
4.1. Akuisisi Data Citra .....	99
4.2. Pre-Processing Data Citra .....	102
4.2.1. Pemisahan <i>Dataset</i> Citra (Normalisasi Kuantitas Citra) .....	102
4.2.2. <i>Augmentasi Dataset</i> Citra (Normalisasi dan modifikasi citra)..	105
4.3. Percobaan pembuatan model arsitektur CNN .....	108
4.3.1. Percobaan fungsi optimasi .....	109
4.3.2. Percobaan jenis <i>pooling</i> dan nilai <i>dropout</i> (perbandingan jumlah <i>epoch</i> ) .....	111
4.4. Pembangunan dan Evaluasi model arsitektur CNN .....	116
4.4.1. <i>Features Extraction</i> .....	118
4.4.2. <i>Classification (Neural Network)</i> .....	119
4.5. Perhitungan pembuatan Arsitektur Model CNN .....	120
4.6. Membangun Sistem Model CNN .....	123
4.7. Pengujian dan Evaluasi Sistem .....	128
4.7.2. Evaluasi Pengujian Sistem .....	141
<b>BAB V SIMPULAN DAN SARAN</b> .....	144
5.1. Simpulan .....	144
5.2. Saran .....	144
<b>DAFTAR PUSTAKA</b> .....	146
<b>LAMPIRAN</b> .....	148
LAMPIRAN A .....	149
LAMPIRAN B .....	150

## DAFTAR TABEL

Tabel		Halaman
1.	Anak Huruf dan Tanda Baca Aksara Lampung [5] .....	8
2.	Tipografi Aksara Lampung [6] .....	9
3.	Anatomi Aksara Lampung [6] .....	12
4.	Citra Digital .....	15
5.	<i>Hyperparameter Convolution Layer</i> .....	23
6.	<i>Hyperparameter Fully Connected Layer</i> [14] .....	28
7.	Pengelompokan Dataset Citra.....	47
8.	Pembagian Dataset Citra.....	50
9.	<i>Hyperparameter</i> tahapan pengenalan ( <i>feature extraction</i> ) .....	52
10.	<i>Hyperparameter</i> tahapan transliterasi ( <i>classification</i> ).....	90
11.	Proses Akuisisi Data .....	99
12.	Augmentasi Data Citra.....	107
13.	Representasi Hasil <i>Augmentasi</i> Citra.....	108
14.	Perbandingan akurasi dan <i>loss</i> percobaan penerapan fungsi optimasi.....	111
15.	Perbandingan <i>metric error</i> jumlah <i>epoch</i> , <i>Average Pooling</i> .....	113
16.	Perbandingan <i>metric error</i> jumlah <i>epoch</i> , <i>Max Pooling</i> .....	115
17.	Perbandingan hasil akhir terbaik <i>metric error</i> .....	116
18.	Perhitungan <i>output</i> pembuatan arsitektur model CNN .....	121

19.	Pengujian Sistem Pengenalan dan Transliterasi .....	128
20.	Hasil Distribusi Keberhasilan Pengujian Sistem .....	138
21.	Durasi Tahapan Pengujian Sistem .....	139
22.	Perhitungan metrik <i>error</i> pengujian sistem .....	141

## DAFTAR GAMBAR

Gambar	Halaman
1. Induk huruf Aksara Lampung [5] .....	8
2. Arsitektur <i>Convolution Neural Network</i> (CNN).....	21
3. Contoh <i>convolution layer</i> matrik 8x8 .....	22
4. Contoh matriks ReLU .....	24
5. Grafik ReLU .....	24
6. Contoh matriks pooling.....	26
7. Arsitektur <i>Artificial Neural Network</i> (ANN).....	29
8. Proses <i>plattening</i> matriks citra.....	30
9. Grafik Fungsi Aktivasi <i>Softmax</i> .....	31
10. Grafik Fungsi Aktivasi <i>Sigmoid</i> .....	32
11. Representasi Citra 8x8 kata ‘Abadi’ per Channel.....	45
12. Nilai intensitas citra 8x8 kata ‘Abadi’ <i>channel Red</i> (R).....	46
13. Nilai intensitas citra 8x8 kata ‘Abadi’ <i>channel Green</i> (G).....	46
14. Nilai intensitas citra 8x8 kata ‘Abadi’ <i>channel Blue</i> (B).....	46
15. Data citra mentah (awal) kelas 01 ‘Abadi’ .....	47
16. Data citra mentah (awal) kelas 02 ‘Abu’ .....	48
17. Data citra mentah (awal) kelas 03 ‘Acara’ .....	48
18. Data citra mentah (awal) kelas 04 ‘Ada’ .....	48
19. Data citra uji kelas 01 ‘Abadi’ .....	50
20. Data citra uji kelas 02 ‘Abu’ .....	50
21. Data citra uji kelas 03 ‘Acara’ .....	50
22. Data citra uji kelas 04 ‘Ada’ .....	50
23. Representasi kernel ukuran 3x3 .....	52
24. Diagram alir penelitian .....	94

25.	Diagram Alir Pembangunan (pelatihan dan validasi) Sistem .....	96
26.	Diagram Alir Pengujian Sistem .....	98
27.	Citra Mentah Aksara Kelas (a) Abadi (b) Abu (c) Acara (d) Ada...	100
28.	Perintah penginputan direktori dataset kedalam <i>software</i> .....	101
29.	Representasi perintah pemisahan data .....	103
30.	Direktori data citra latih kelas 01 'Abadi' .....	104
31.	Direktori data citra validasi kelas 01 'Abadi' .....	104
32.	Perintah persiapan modifikasi <i>Augmentasi data</i> citra.....	105
33.	Perintah Implementasi <i>Augmentasi Dataset</i> citra latih dan citra validasi .....	106
34.	Diagram Alir <i>Augmentasi data</i> .....	107
35.	Representasi Perintah Percobaan fungsi optimasi 'Adagrad' .....	109
36.	Grafik Akurasi dan Loss percobaan teknik optimasi.....	110
37.	Representasi percobaan perbandingan jumlah <i>epoch</i> , jenis <i>pooling</i> dan nilai <i>dropout</i> .....	114
38.	Representasi Model Arsitektur <i>Convolution Neural Network</i> (CNN) .....	117
39.	Arsitektur model CNN pada proses <i>features extraction</i> .....	118
40.	Arsitektur CNN pada proses klasifikasi ( <i>neural network</i> ).....	119
41.	Rangkuman model Arsitektur CNN.....	120
42.	Diagram Alir Arsitektur Model CNN Aksara Lampung .....	123
43.	Persiapan pelatihan model pembelajaran CNN .....	124
44.	Pelatihan model pembelajaran CNN.....	125
45.	Durasi Pelatihan .....	125
46.	Grafik nilai akurasi pembangunan (pelatihan dan validasi) model CNN .....	126
47.	Grafik nilai kerugian ( <i>loss</i> ) pembangunan (pelatihan dan validasi) model CNN .....	127

## **BAB I. PENDAHULUAN**

### **1.1 Latar Belakang**

Era digital mempunyai potensi kebermanfaatan namun juga kerugian dalam proses pengembangannya. Pengaruh maju mundurnya dan bermanfaat tidaknya era digital di tengah kehidupan berbangsa dan bernegara bergantung pada bagaimana pengoptimalan dan penjagaan, serta pelestarian dan pengembangan teknologi, tanpa perlu mengurangi atau bahkan menghilangkan nilai-nilai luhur dan jati diri bangsa (Budaya). Salah satu fokus ilmu yang dinilai sangat efektif dalam melakukan pengembangan budaya di era digital saat ini ialah dengan melakukan pengolahan dan pengoptimalan teknologi informasi.

Pada bidang penerapan teknologi dan informasi, tantangan nyata dalam perjalanan era digital semakin kompleks, seperti memetakan dan mengatasi permasalahan yang ada, menaikkan tingkatan efektivitas, merancang kreativitas, inovasi, dan membangun efisiensi dalam melakukan pekerjaan [1]. Pada bidang sosial budaya, salah satu pengaruh teknologi di era digital terutama pada bidang teknologi informasi saat ini, ialah menimbulkan perubahan pada pola interaksi antar masyarakat terutama dalam masyarakat ekonomi golongan menengah ke atas, terkhusus dalam hal penggunaan bahasa dan aksara daerah yang dinilai kuno dan tertinggal [1]. Salah satu daerah yang mempunyai aksara ialah provinsi Lampung.

Menyikapi hal tersebut, maka perlu adanya perhatian khusus terhadap perkembangan teknologi dan sosial-budaya. Perlu adanya teknologi informasi yang mampu menghilangkan stigma kuno dan tertinggal tanpa menghilangkan nilai luhur dari suatu budaya, terkhusus bahasa dan aksara yang merupakan sarana komunikasi suatu bangsa.

Salah satu fokus ilmu yang dinilai mampu melakukan perkembangan teknologi dan sosial budaya tersebut ialah penerapan teknologi citra yang menerapkan metode pembelajaran yang mampu melakukan proses mengenali dan mentransliterasi data (citra aksara Lampung), yaitu *Convolution Neural Network* (CNN).

Adapun penelitian terdahulu yang menerapkan pengenalan citra berbasis CNN diantaranya;

Sazqiah,N. dkk [2], dalam upaya mengenali aksara Lampung menggunakan metode *Convolution Neural Network* (CNN) yang menghasilkan akurasi sebesar 100%

Silalahi, dkk [3], dalam upaya mengimplementasikan *Convolution Neural Network* (CNN) untuk klasifikasi kata pada citra teks yang menghasilkan model yang melakukan proses pengujian klasifikasi dengan keberhasilan pengklasifikasian sebanyak 1.727 citra dengan benar.

Mulyanto, dkk [4], dalam upaya mengenali aksara Lampung menggunakan penerapan *Optical Character Recognition* (OCR) dengan metode *Convolution Neural Network* (CNN) yang menghasilkan tingkat klasifikasi dan pengenalan karakter (segmentasi) yang lebih akurat.

Penelitian ini akan membuat suatu sistem yang dapat melakukan pengenalan dan transliterasi data karakter (aksara Lampung) menggunakan metode *Artificial Intelligent*, yaitu *Convolution Neural Network* (CNN). Perbedaan penelitian ini terhadap penelitian sebelumnya [2]-[4] ditunjukkan pada objek penelitian yang digunakan, yaitu berupa data citra aksara Lampung dalam bentuk kata.

Penelitian ini diharapkan dapat berkembang dan menghasilkan aplikasi lanjutan yang terintegrasi kedalam perangkat digital ataupun perangkat elektronik, serta mampu mengenali dan mentransliterasi tulisan aksara lampung dalam klasifikasi yang lebih banyak lagi.

## **1.2 Tujuan Penelitian**

Tujuan dari penelitian ini adalah membangun sistem identifikasi pengenalan dan transliterasi kata aksara Lampung berbasis *Convolution Neural Network* (CNN).

## **1.3 Manfaat Penelitian**

Adapun manfaat yang diharapkan pada penelitian ini adalah:

1. Dapat menjadi salah satu sarana alternatif dalam mempertahankan dan melestarikan kembali budaya aksara lampung.
2. Upaya menerapkan teknologi dalam bidang pengolahan citra.
3. Hasil penelitian dapat dijadikan bahan rujukan menciptakan ataupun mengembangkan teknologi terbaru, baik berupa aplikasi ataupun alat dalam upaya melestarikan dan mempertahankan budaya Lampung pada khususnya dan Indonesia pada umumnya

#### **1.4 Rumusan Masalah**

Rumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana merancang dan membangun sistem yang mampu mengenali suatu karakter (Aksara Lampung) ?
2. Bagaimana merancang dan membangun sistem yang mampu menyeleksi dan mengelola data digital yang berasal dari suatu karakter secara efisien?

#### **1.5 Batasan Masalah**

Beberapa hal yang membatasi masalah dalam pembahasan tugas akhir ini adalah sebagai berikut:

- a. Studi kasus pada penelitian ini hanya mengelola dan menyeleksi data sebanyak  $\pm 1732$  citra. Terbagi kedalam 4 kelas (*output*) dengan masing-masing  $\pm 423$  citra untuk membangun sistem dan masing-masing  $\pm 10$  citra untuk menguji sistem pada masing-masing katanya
- b. Sistem ini digunakan hanya untuk melakukan pengenalan dan transliterasi terhadap empat jenis kata dari aksara Lampung menjadi Bahasa Indonesia.

#### **1.6 Sistematika Penulisan**

Dalam memudahkan kepenulisan dan pemahaman mengenai materi dalam tugas akhir ini, tugas akhir ini dibagi ke dalam 5 bab, yaitu :

#### **BAB I. PENDAHULUAN**

Memaparkan latar belakang, tujuan penelitian, manfaat penelitian, rumusan masalah, batasan masalah, dan sistematika penulisan.

## **BAB II. TINJAUAN PUSTAKA**

Memaparkan tentang landasan dasar teori yang melandasi perancangan dan pembuatan sistem pengenalan dan transliterasi aksara lampung menggunakan metode penerapan *Convolutional Neural Network* (CNN).

## **BAB III. METODE PENELITIAN**

Memaparkan waktu dan tempat pelaksanaan penelitian, alat dan bahan yang diperlukan, tahapan penelitian yang dilakukan, serta diagram alir metode yang direncanakan.

## **BAB IV. HASIL DAN PEMBAHASAN**

Memaparkan hasil penelitian, pembahasan, dan perhitungan kinerja metode sistem model yang direncanakan.

## **BAB V. SIMPULAN DAN SARAN**

Memaparkan simpulan yang didapatkan melalui hasil penelitian, dan saran - saran untuk tahapan pengembangan lebih lanjut.

## BAB II. TINJAUAN PUSTAKA

### 2.1 *State of The Art*

*State of The Art* merupakan pernyataan yang menyatakan bahwa pemecahan konflik yang digunakan tidak sama dengan penelitian yang sudah berkembang atau sedang diterapkan oleh peneliti lain. Dalam hal ini dijelaskan perbandingan dengan penelitian sebelumnya dan dijadikan sebagai referensi dalam penyusunan tugas akhir.

Referensi penelitian untuk mendukung penulisan makalah tugas akhir ini antara lain:

Penelitian pertama [2] tentang pengenalan aksara Lampung menggunakan metode *Convolution Neural Network* (CNN). Penelitian ini berhasil melakukan pengenalan aksara Lampung dengan akurasi 100%

Penelitian kedua [3] tentang Implementasi *Convolution Neural Network* untuk klasifikasi kata pada citra teks. Penelitian ini berhasil melakukan pengujian untuk melakukan klasifikasi yang lebih banyak dari penelitian sebelumnya, yaitu 1.727 citra kata dengan benar.

Penelitian ketiga [4] tentang Penerapan *Convolutional Neural Network* (CNN) pada Pengenalan Aksara Lampung berbasis *Optical Character Recognition*

(OCR). Penelitian ini berhasil melakukan klasifikasi dan pengenalan karakter aksara (segmentasi) yang lebih akurat.

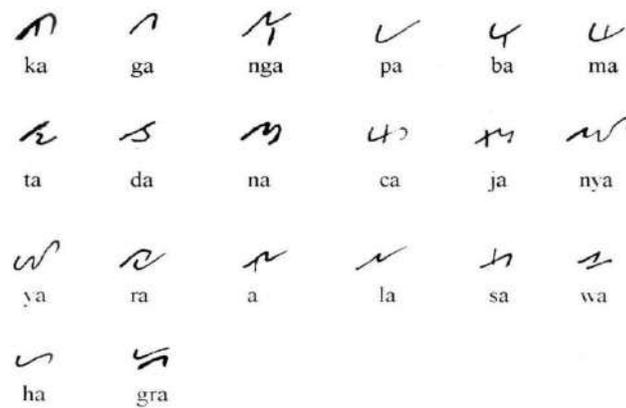
Penelitian tugas akhir (skripsi) ini memiliki perbedaan dengan referensi tersebut di atas. Perbedaan terletak dalam objek penelitian yang digunakan. Objek penelitian yang akan diolah pada penelitian ini berupa citra aksara Lampung dalam bentuk kata, bukan per huruf atau suku kata seperti penelitian sebelumnya.

Dengan objek penelitian berupa citra aksara dalam bentuk kata, proses pengenalan dan transliterasi citra aksara Lampung akan menghasilkan *output* yang lebih kompleks dengan pemodelan yang lebih sederhana. Selain itu, data yang telah diolah dapat ditingkatkan kembali menjadi *database* tersendiri yang dapat digunakan untuk penelitian pengembangan selanjutnya.

## **2.2 Aksara Lampung**

Tulisan Lampung (Aksara Lampung) sering disebut oleh suku asli Lampung dengan nama ‘Surat Lampung’ (*Had Lampung*). Aksara Lampung terdiri dari tiga bagian, yaitu; Induk huruf (*kelabai surat*), tanda bunyi/ Anak huruf (*benah surat*), dan tanda-tanda baca [5].

Aksara Lampung memiliki induk huruf sebanyak 20 huruf (seperti yang ditunjukkan oleh Gambar 1 di bawah ini) dan anak huruf berjumlah 12 huruf (seperti yang ditunjukkan oleh Tabel 1 di bawah ini).



Gambar 1 Induk huruf Aksara Lampung [5]

Terlihat pada Gambar 1 di atas bahwa Induk aksara Lampung adalah berupa suku kata yang terdiri atas 20 huruf, yaitu Ka, Ga, Nga, Pa, Ba, Ma, Ta, Da, Na, Ca, Ja, Nya, Ya, Ra, A, La, Sa, Wa, Ha, dan Gra.

Tabel 1 Anak Huruf dan Tanda Baca Aksara Lampung [5]

No	Nama	Bunyi	Aksara	Key board	Contoh
1	datas	an	---	A	= panda
2	ulan	i	---	i	= pipi
3	ulan	e	---	e	= menara
4	bicek	e	---	E	= cabe
5	tekelubang	ang	---	@	= cabang
6	rejenjung	ar	---	R	= sabar
7	tekelungau	au	---	W	= limau
8	bitan	o	---	o	= bolong
9	bitan	u	---	u	= busur
10	tekelingai	ai	---	I	= bangkai
11	keleniah	ah	---	x	= susah
12	nengen		---/	M	= tutup
13	tanda koma		,	,	= saya, dia
14	tanda seru		!	!	= awas !
15	tanda tanya		?	?	= bisa?
16	tanda titik		.	.	= sudah.

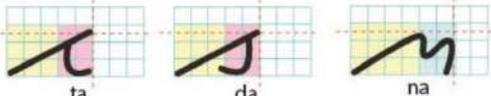
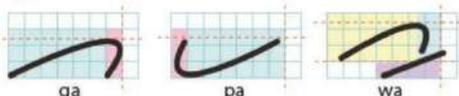
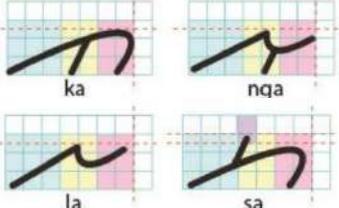
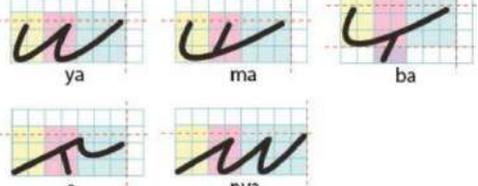
Dalam Aksara Lampung terdapat tanda baca seperti halnya aksara Arab. Misalnya, tanda yang letaknya di atas huruf tanda fathah, sedangkan tanda baca di bawah huruf kasrah [5]. Dikarenakan setiap hurufnya mengandung bunyi ‘a’, masyarakat Lampung menyebutnya dengan tulisan Basaja [5].

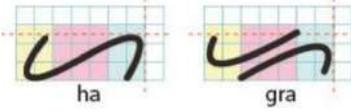
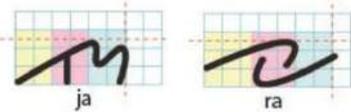
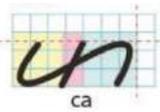
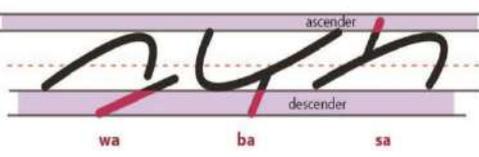
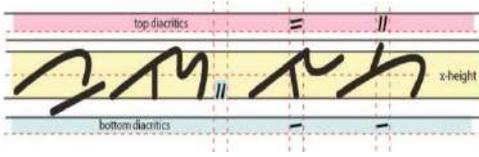
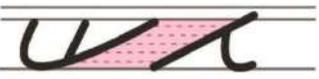
### 2.2.1 Tipografi dan Anatomi Aksara Lampung

Pengukuran tipografi merupakan langkah awal dalam mendesain dan mengenali sebuah huruf. Tanpa pengukuran tipografi yang jelas, huruf yang dihasilkan menjadi tidak jelas dan akan berakibat pada transliterasi yang keliru. Parameter karakter dan struktur, ukuran huruf, spasi huruf, dan *grid* adalah beberapa hal yang harus dipertimbangkan ketika melakukan transformasi ataupun pengenalan aksara Lampung [6]. Adapun Tipografi Aksara Lampung diperlihatkan oleh Tabel 2 di bawah ini.

Tabel 2 Tipografi Aksara Lampung [6]

No	Karakter dan Parameter	Keterangan Deskripsi Bentuk	Tujuan
1	Garis	<p>Garis lengkung (Cekung dan Cembung)</p>  <p>Garis campuran lengkung atas-tegak</p> 	Pembentukan dasar

No	Karakter dan Parameter	Keterangan Deskripsi Bentuk	Tujuan
1	Garis	<p>Garis sampungan lengkung bawah-tegak</p> 	Pembentukan dasar
		<p>Garis lengkung rangkap</p> 	
		<p>Garis lengkung rangkap dengan garis tegak</p> 	
2	Rasio Ukuran	<p>3:2/4:2</p> 	Penentu jarak harmonis antar huruf
		<p>6:1</p> 	
		<p>3:2:2</p> 	
		<p>2:2:3</p> 	

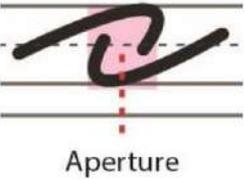
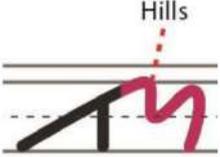
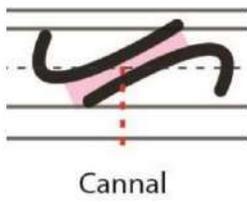
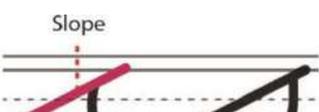
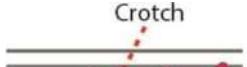
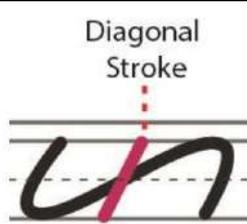
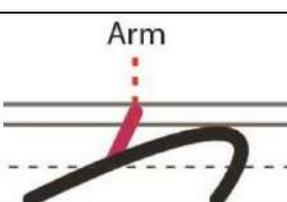
No	Karakter dan Parameter	Keterangan Deskripsi Bentuk	Tujuan
2	Rasio Huruf	<p>2:3:2</p>  <p>ha gra</p> <p>2:2:2</p>  <p>ja ra</p> <p>3:1:3</p>  <p>ca</p>	Penentu jarak harmonis antar huruf
3	Proporsi Huruf	<p>Penempatan induk huruf [Kecuali huruf induk 'wa', 'ba' dan 'sa' tidak memiliki descender dan ascender]</p>  <p>ascender</p> <p>descender</p> <p>wa ba sa</p> <p>Peletakan tanda bunyi [Di atas (<i>top</i>), Tengah(<i>middle</i>), Di bawah(<i>bottom</i>)]</p>  <p>top diacritics</p> <p>middle</p> <p>bottom diacritics</p> <p>x-height</p>	Penentu kesatuan dan kemudahan pembacaan
4	Spasi Huruf	<p>Pertemuan karakter garis cembung/cekung dan sebaliknya</p>  <p>Pertemuan Kombinasi</p> 	Penentu kebutuhan ruang spasial

No	Karakter dan Parameter	Keterangan Deskripsi Bentuk	Tujuan
5	Grid	<p>Skala umum penempatan huruf induk dan tanda bunyi [1:3:2]</p> <p>a. base line for top diacritics b. top line for middle diacritics c. base line for bottom diacritics</p>	Penentu kombinasi penempatan
		<p>Penggunaan tanda bunyi tunggal dan ganda [3:2/4:2 (ujung kanan dan ujung kiri), 3:2:2, (ketiga dari kiri), dan 6:1(kedua dari kiri)]</p>	

Selain melalui pengukuran tipografi, karakter aksara Lampung dapat diidentifikasi melalui pembentuk anatominya. Setidaknya ada 12 bagian bagian yang dapat diidentifikasi dalam aksara lampung, yaitu seperti yang diperlihatkan oleh Tabel 3 di bawah ini.

Tabel 3 Anatomi Aksara Lampung [6]

No	Nama	Bentuk	Keterangan
1	<i>Hook</i>		Bagian puncak dari garis lengkung utama yang berbentuk cembung.
2	<i>Leg</i>		Bagian dari huruf yang memanjang ke bawah, melekat pada garis huruf utama dan menggantung di ujung lainnya.
3	<i>Valley</i>		Bagian patahan garis utama huruf yang berbentuk cekung.

No	Nama	Bentuk	Keterangan
4	<i>Aperture</i>		Sebuah ruang negatif antar huruf yang membentuk area terbuka membulat.
5	<i>Hills</i>		Bagian patahan garis utama huruf yang bergerigi dan membentuk cembungan.
6	<i>Cannal</i>		Sebuah ruang negatif tipis berbentuk memanjang antara dua garis utama huruf.
7	<i>Slope</i>		Garis utama huruf pembentuk seluruh karakter aksara Lampung.
8	<i>Terminal</i>		Garis perpanjangan kaki ( <i>leg</i> ) yang berbentuk melengkung.
9	<i>Crotch</i>		Sebuah sudut ruang negatif di antara garis cekungan tajam.
10	<i>Double Valley</i>		Garis cekung dengan sudut tajam yang berulang.
11	<i>Diagonal Stroke</i>		Garis diagonal yang memotong garis utama.
12	<i>Arm</i>		Bagian dari huruf yang memanjang ke atas, melekat pada garis huruf utama dan menggantung di ujung lainnya.

### 2.3 Citra

Citra adalah gambaran visual mengenai sesuatu atau kumpulan beberapa objek. Objek-objek tersebut diantaranya; Gambar, Foto, Hasil Rontgen, serta citra satelit, dan masih banyak lagi bentuk-bentuk objek citra lainnya [7]. Berdasarkan sifat, sinyal dan sistem, citra dikelompokkan kedalam dua macam, yaitu citra Analog dan citra Digital.

Citra analog ialah citra yang memiliki sifat sinyal yang kontinu atau analog seperti foto, tulisan, lukisan, hasil CT scan, pemandangan alam, plat nomor kendaraan, dan lain sebagainya. Citra analog dihasilkan oleh perangkat yang bersifat analog juga, seperti kamera foto analog, video kamera analog, CT scan, *webcam*, sensor *ultrasound* dan lain sebagainya [8]. Citra analog merupakan citra yang tidak mampu dibaca dan diproses dalam komputer secara langsung. Supaya citra analog ini mampu dibaca dan diproses oleh komputer, citra analog ini harus mengalami proses perubahan bentuk dari analog ke digital terlebih dulu.[8]

Citra digital ialah citra yang memiliki sifat sinyal yang diskret atau digital seperti foto, gambar dan video dalam *handphone*, monitor laptop, televisi, serta dalam media penyimpanan, seperti *memory card*, *flashdisk*, dan lainnya. Citra digital merupakan citra yang dapat dibaca dan diproses secara langsung dalam komputer. Citra digital dapat digunakan dalam penerapan segmentasi, penghitungan, deteksi, pelacakan dan pengenalan.[8]

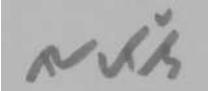
Secara matematis, citra digital dapat digambarkan sebagai susunan angka dalam kolom dan baris tertentu atau matriks yang terdiri dari M kolom dan N baris  $[M \times N]$ . Matriks  $M \times N$  tersebut memiliki elemen terkecil yang merupakan

bagian-bagian dari baris dan kolom. Elemen terkecil tersebut dikenal dengan sebutan piksel (*picture element*). Piksel terdiri dari dua metrik, yaitu intensitas (warna) dan koordinat. Seperti yang terlihat oleh persamaan di bawah ini :

$$f(x, y) = \begin{bmatrix} f(0,0) & \dots & f(0, M - 1) \\ \dots & \dots & \dots \\ f(N - 1, 0) & \dots & f(N - 1, M - 1) \end{bmatrix} \dots\dots\dots (2.1)$$

Berdasarkan piksel yang menyusunnya, citra digital dapat dipecah ke dalam tiga bagian, yaitu citra biner (citra monokrom), citra skala keabuan (*grayscale*), dan citra warna (RGB). Penjelasan lebih lanjut dapat dilihat pada Tabel 4 di bawah ini.

Tabel 4 Citra Digital

No	Jenis Citra	Piksel yang menyusun	
		Secara intensitas (warna)	Secara sistem
1	Citra Biner 	Memiliki hanya dua nilai intensitas (1 bit) [1 untuk putih, 0 untuk hitam]	Diperoleh melalui proses Binerisasi bagian gambar skala keabuan ( <i>grayscale</i> ), menggunakan nilai ambang batas [ <i>Thesholding Value</i> ]
2	Citra berskala keabuan ( <i>grayscale</i> ) 	Memiliki rentan nilai intensitas dari 0-225 (8 bit)	Kombinasi dari pembagian warna hitam dan putih.
3	Citra warna(RGB) 	Setiap piksel membutuhkan 24 bit untuk mengkodekan 3 komponen RGB	Kombinasi tiga model warna adiktif utama, yaitu <i>Red</i> (R), <i>Green</i> (G). <i>Blue</i> (B).

Berdasarkan grafik penyebaran nilai-nilai intensitas (warna) piksel dari suatu citra (Histogram), terdapat beberapa karakteristik penyusun citra, yaitu kontras,

ketajaman, resolusi dan derau. Kontras adalah perbedaan pencahayaan dan warna. Karakteristik kontras dapat dipengaruhi oleh pencahayaan dalam proses akurasi atau juga bisa karena kondisi fisik objek yang memang sudah tidak jelas [9].

Ketajaman berkaitan dengan proses menghitamkan batas antara dua bagian dalam citra yang berdekatan. Resolusi adalah jumlah piksel dalam suatu citra (menggambarkan tingkat kerapatan intensitas (warna)). Derau adalah bentuk gangguan yang tidak memiliki arti apapun dan dapat mengurangi informasi yang sebenarnya [9].

## 2.4 Pengolahan Citra Digital

Pengolahan citra digital adalah ilmu yang mempelajari pembahasan yang berkaitan dengan upaya memperbaiki ataupun mengelola kualitas citra (perubahan warna, pengaturan skala kontras, restorasi citra), transformasi citra (rotasi, translasi, geometrik dan pengaturan skala), pengolahan citra ciri (*feature images*) yang optimal dalam melakukan proses analisis, dan masih banyak lagi [10].

Secara umum, bagian yang diperlukan dalam pemrosesan citra digital dapat dibagi menjadi empat keperluan, yaitu *digitizer*, komputer, piranti tampilan peraga, dan media penyimpanan. *Digitizer (digital image acquisition system)*, merupakan perangkat yang diperlukan untuk mengambil citra digital, menjelajah citra dan mengkonversinya.

Hasil dari *digitizer* adalah matriks yang menyatakan nilai intensitas (warna) cahaya pada suatu titik. Contoh *digitizer* ialah kamera digital dan *scanner*. Komputer, diperlukan sebagai sistem pemroses citra. Piranti tampilan peraga,

diperlukan untuk mengkonversi nilai matriks intensitas (warna) yang menggambarkan citra ke tampilan yang dapat diartikan oleh mata manusia. Contoh piranti tampilan ialah monitor dan *printer*. Media penyimpanan, diperlukan untuk menyimpan hasil secara permanen agar dapat diproses lagi pada waktu yang lain [8].

## 2.5 Tahapan Pengolahan Citra

### 2.5.1. Akuisisi Data Citra

Akuisisi data citra adalah proses perekaman atau pengambilan data dari objek nyata ke dalam bentuk objek digital (data numerik), sehingga data dapat dibaca dan diolah (manipulasi) oleh komputer [9]. Seringkali hasil yang diperoleh melalui proses akuisisi data memiliki kualitas yang buruk, misalnya citra mengalami derau (*noise*) pada saat pengiriman melalui saluran transmisi, citra terlalu terang/gelap, citra kurang tajam, kabur, dan sebagainya [11]. Maka dari itu, perlu dilakukan perbaikan kualitas citra terlebih dahulu sebelum melakukan tahapan pengenalan pola citra atau klasifikasi citra menggunakan metode *Artificial Intilligent*.

Perbaikan kualitas citra yang dapat dilakukan diantaranya perbaikan kontras gelap atau terang, perbaikan tepian objek (*edge enhancement*), penajaman citra (*sharpening*), pemberian warna semu (*pseudocoloring*), dan penipisan derau (*noise filtering*). Proses perbaikan kualitas citra umumnya dilakukan dalam tahapan yang umumnya disebut dengan *preprocessing* data.

### 2.5.2. *Preprocessing* Data

*Preprocessing* merupakan tahap awal dalam pengolahan data citra sebelum data citra diolah dengan metode *Artificial Intelligent*. Tahapan *preprocessing* ini berfokus pada perbaikan kualitas citra dan penyetaraan atau penyeragaman citra (normalisasi) dan modifikasi citra (*augmentasi data*).

#### 2.5.2.1. Normalisasi Data Citra

Normalisasi data citra merupakan penerapan konsep ilmu pengolahan citra digital yang lebih berfokus kepada upaya penyeragaman *dataset* citra. Normalisasi yang dilakukan dapat berupa normalisasi piksel (ukuran), normalisasi intensitas (warna) dan normalisasi kuantitas (jumlah) citra. Normalisasi data citra dapat dilakukan melalui proses khusus atau tersendiri dan/atau dapat juga dilakukan bersamaan dalam proses *preprocessing* lainnya. Normalisasi piksel (ukuran) dan intensitas (warna) umumnya dilakukan bersamaan dalam tahap *augmentasi* data citra. Sementara, normalisasi kuantitas (jumlah) citra umumnya dilakukan bersamaan dalam proses pemisahan *dataset* citra

#### 2.5.2.2. Pemisahan *Dataset* Citra

Pemisahan *dataset* citra merupakan tahapan memisahkan dan mengelompokkan *dataset* citra ke dalam kategori tertentu, sesuai dengan kebutuhan algoritma model pembelajaran. Pemisahan *dataset* citra umumnya dilakukan dalam algoritma model pembelajaran kategori *Supervised Learning*. Kategori yang digunakan

dalam tahapan ini, umumnya dibagi kedalam tiga kategori, yaitu data latih, data validasi dan data uji.

### 2.5.2.3. *Augmentasi Data Citra*

*Augmentasi data* citra merupakan penerapan konsep ilmu pengolahan citra digital yang lebih berfokus kepada upaya perubahan/modifikasi citra atau transformasi citra (rotasi, translasi, geometri, pengaturan skala, dan masih banyak lagi). Selain untuk modifikasi, *augmentasi data* citra umumnya juga digunakan untuk memperbanyak data citra.

Tujuan utama proses *augmentasi data* umumnya adalah meningkatkan akurasi dalam penerapan model *artificial intelligent* terutama *Convolution Neural Network* (CNN). Teknik *augmentasi data* yang dapat digunakan untuk data citra diantaranya adalah transformasi geometri, transformasi spasi warna, mengubah fitur inti, penghapusan acak dan pengadukan gambar.

Transformasi geometri merupakan tahapan modifikasi citra yang berfokus kepada perubahan bentuk geometri citra secara acak, seperti membalik citra, memotong citra, meregangkan dan memperbesar citra. Transformasi spasi warna merupakan tahapan modifikasi citra yang berfokus kepada perubahan nilai intensitas (warna) piksel citra secara acak, seperti mengubah warna RGB, mengubah kontras dan mengubah kecerahan.

*Tools python library* *Augmentasi data* citra yang dapat digunakan diantaranya *opencv*, *pillow*, dan *imgaug*.

## 2.6 Convolution Neural Network (CNN)

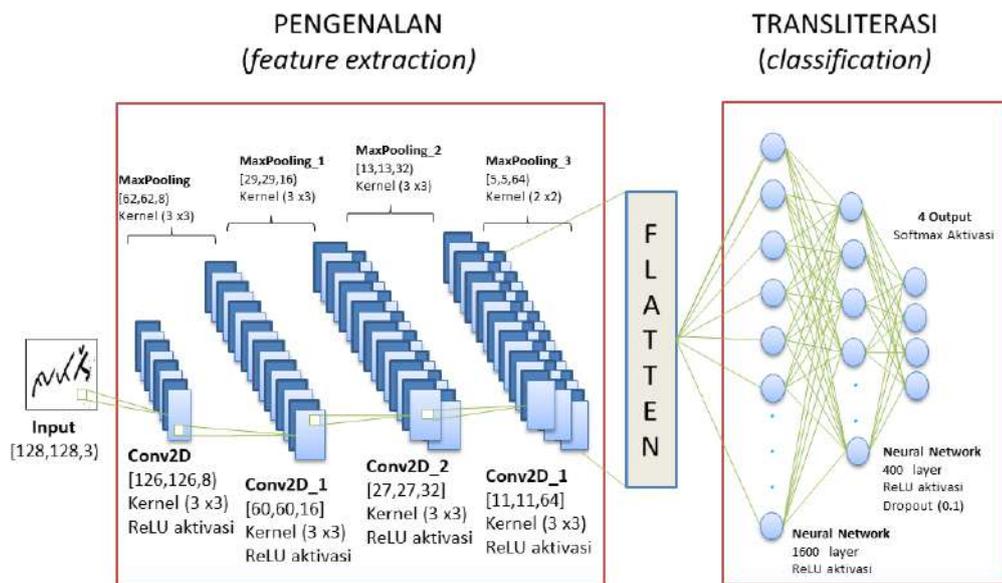
*Convolutional Neural Network* (CNN) merupakan salah satu metode kecerdasan jaringan sistem syaraf (*deep learning*) yang seringkali digunakan pada penerapan bidang ilmu *computer vision* terkhusus dalam proses mengklasifikasi citra dan mengenali objek.

Berdasarkan pengembangannya, *Convolution Neural Network* (CNN) merupakan pengembangan metode *artificial intelligent* sebelumnya, yaitu Jaringan Sistem Syaraf (JST) atau seringkali juga disebut dengan *artificial neural network* (ANN). CNN adalah variasi *Multilayer Perceptron* yang terinspirasi dari jaringan syaraf manusia.

Metode CNN menggabungkan dua konsep ilmu sekaligus, yaitu konvolusi dan *neural network*. Metode konvolusi berperan dalam proses pengenalan ciri (*features extraction*), sementara *neural network* berperan dalam proses transliterasi (klasifikasi).

### 2.6.1. Gambaran Umum Arsitektur CNN

Arsitektur CNN terdiri atas beragam jenis. Namun, secara umum arsitektur model CNN terdiri dari *convolutional layer*, *Rectifier Linear Unit* (ReLU) *layer*, *pooling layer*, dan *fully connected layer* (layer klasifikasi). Seperti diperlihatkan oleh Gambar 2 di bawah ini.



Gambar 2 Arsitektur *Convolution Neural Network* (CNN)

Terlihat pada Gambar 2 di atas bahwa arsitektur *convolution neural network* (CNN) tersebut secara garis besar terdiri dari dua bagian yaitu pengenalan (*features extraction*) dan transliterasi (*classification*). Pengenalan (*features extraction*) memiliki empat bagian *convolution layer* dan *pooling layer*. Transliterasi (*classification*) memiliki tiga bagian *fully connected layer* (*neural network*) [12].

Secara matematis format data citra (*Input/Output*) pada arsitektur ditunjukkan dalam persamaan berikut:

$$F_c = N \times H \times W \times C \quad (2.2)$$

Dimana :

$F_c$  = Format Citra CNN

$N$  = Banyaknya Dataset Citra

$H$  = *Height*, Besarnya nilai baris piksel (tinggi) citra

$W$  = *Weight*, Besarnya nilai kolom piksel(bobot) citra

$C$  = *Channel*, Banyaknya channel

Adapun perhitungan perubahan nilai/bentuk *output* data citra CNN disetiap prosesnya dilakukan dengan persamaan sebagai berikut :

$$O = \frac{W-N+2P}{S} + 1 \quad (2.3)$$

Dimana :

$O = Output$

$W = \text{panjang/tinggi Input}$

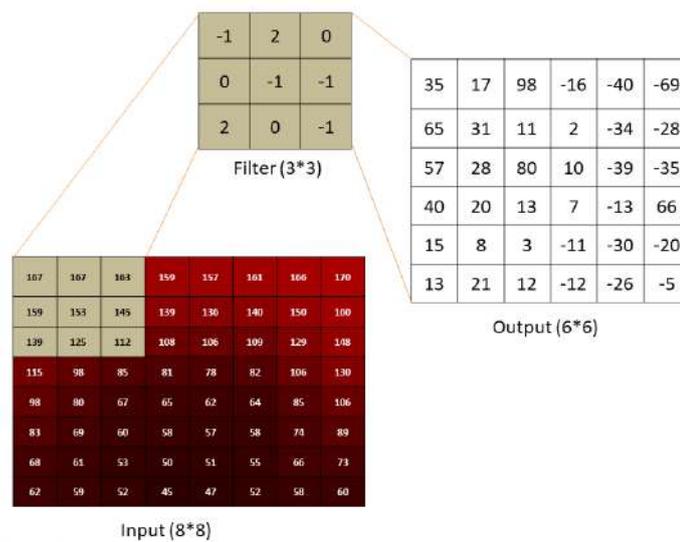
$N = \text{Panjang/tinggi Filter(kernel\_size)}$

$P = \text{Padding}$

$S = \text{Stride}$

### 2.6.2. Convolutional Layer

*Convolutional layer* merupakan layer yang mempunyai filter atau *kernel* sebanyak  $k$ . Filter berdimensi  $n \times n$ .  $n$  (baris/kolom) seringkali lebih kecil dari ukuran dimensi input citra. Contoh proses matriks *convolutional layer* ini diperlihatkan oleh Gambar 3 di bawah ini.



Gambar 3 Contoh *convolution layer* matrik 8x8

Gambar 3 di atas menunjukkan representasi *convolution layer* dengan input 8x8 piksel, filter 3x3 piksel, pergeseran sebanyak 1 piksel dan jenis *padding* berupa *zero padding*. Melalui ketentuan tersebut menghasilkan *output* berupa citra 6x6. *Convolutional layer* memiliki *hyperparameter* dan *parameter*. *Hyperparameter* pada layer ini memiliki 3 *parameter*, yaitu *Depth*, *Stride*, dan *Zero-Padding*. Masing-masing parameter menjadi acuan untuk dapat menentukan jumlah dan ukuran hasil ekstraksi layer, seperti yang diperlihatkan oleh Tabel 5 di bawah ini.

Tabel 5 *Hyperparameter Convolution Layer*

No	<i>Hyperparameter</i>	Keterangan
1	<i>Depth</i>	Kedalaman layer atau jumlah layer konvolusi.
2	<i>Stride</i>	Jumlah pergeseran filter pada proses konvolusi.
3	<i>Zero-Padding</i>	Jenis penambahan nilai intensitas nol di daerah sekitar input gambar.

Parameter bagian layer ini merupakan himpunan *learnable filter* atau suatu kernel yang nilainya berupa bobot. Melalui penggambaran filter, dalam menerapkan proses konvolusi (*convolution*) akan memperoleh bagian yang mencolok dari citra masukan dari *convolutional layer* disebut *feature map* sebanyak  $k$  dengan dimensi  $m \times n + 1$ . Hasil dari proses konvolusi kemudian harus dinormalisasi dengan fungsi aktivasi berupa *Rectifier Linear Unit* (ReLU) untuk menghapuskan bagian yang tidak dibutuhkan.

### 2.6.3. *Rectifier Linear Unit (ReLU) Layer*

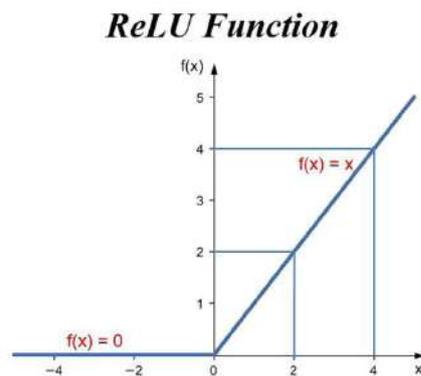
*ReLU* adalah layer tambahan yang merupakan fungsi matematis, memungkinkan pelatihan yang lebih cepat dan efektif dengan memetakan nilai

negatif ke nol dan mempertahankan nilai positif. Proses ini dilakukan karena pada umumnya, nilai intensitas (warna) dalam setiap piksel bernilai 0 atau positif. Berkisar 0 – 1 untuk citra biner dan 0-255 untuk citra *grayscale* dan citra warna (RGB). Contoh proses aktivasi ReLU ini diperlihatkan oleh Gambar 4 di bawah ini.

35	17	98	-16	-40	-69	35	17	98	0	0	0
65	31	11	2	-34	-28	65	31	11	2	0	0
57	28	80	10	-39	-35	57	28	80	10	0	0
40	20	13	7	-13	66	40	20	13	7	0	66
15	8	3	-11	-30	-20	15	8	3	0	0	0
13	21	12	-12	-26	-5	13	21	12	0	0	0

Gambar 4 Contoh matriks ReLU

Terlihat pada Gambar 4 di atas bahwa *output* nilai negatif pada *input* (Gambar kiri) setelah diberikan fungsi aktivasi *ReLU* berubah menjadi bernilai nol (Gambar kanan). Secara Grafik, diperlihatkan pada Gambar 5 di bawah ini.



Gambar 5 Grafik ReLU

Pada dasarnya ReLU adalah operasi per-piksel dengan cara mengganti nilai negatif piksel di dalam *feature map* menjadi nol. Secara matematis dirumuskan sebagai berikut :

$$f(x_i) = \begin{cases} x_i, & x_i > \epsilon \\ 0, & x_i \leq \epsilon \end{cases} \dots\dots\dots (2.4)$$

Fungsi ReLU digunakan pada tahap *forward pass* CNN adalah  $y = \max(\epsilon, x)$ . Sedangkan untuk tahap *backward pass* digunakan fungsi turunannya sebagai berikut :

$$f'(x_i) = \begin{cases} 1, & x_i > \epsilon \\ 0, & x_i \leq \epsilon \end{cases} \dots\dots\dots (2.5)$$

Dimana :

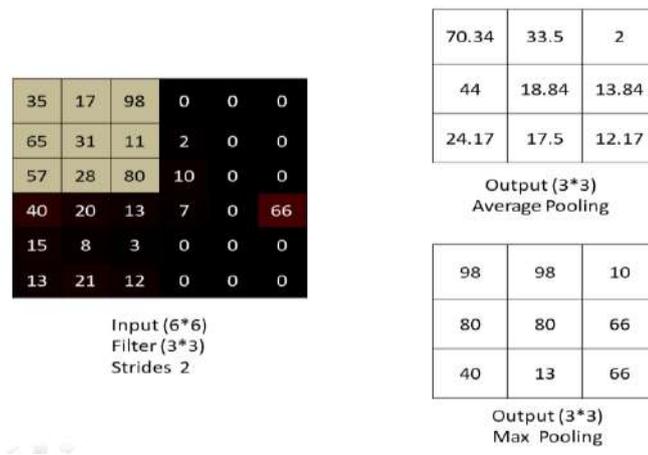
$x_i$  = Input aktivasi non-linear pada channel ke-i.

$\epsilon$  = Nilai normalisasi, bernilai 0

Walaupun ReLU tidak terdiferensiasi di  $\epsilon$ . Tetapi mempunyai *subdifferensial* pada selang (0,1).

#### 2.6.4. Pooling Layer

*Pooling layer* akan mereduksi ukuran spasial dan jumlah parameter dalam jaringan serta mempercepat komputasi dan mengontrol terjadinya *overfitting*. Terdapat dua jenis *pooling* yang sering digunakan, yaitu *Max Pooling* dan *Average Pooling*. Contoh proses *pooling* layer ini diperlihatkan oleh Gambar 6 di bawah ini.



Gambar 6 Contoh matriks pooling

Gambar 6 di atas menunjukkan representasi dua jenis *Pooling* dengan *input* berupa citra 6x6 piksel, filter 3x3 piksel, dan pergeseran sebanyak 2 piksel. Melalui ketentuan tersebut menghasilkan *output* berupa citra 3x3. *Max Pooling* merupakan proses reduksi yang akan melakukan pengambilan terhadap nilai terbesar dari *kernel* yang berdimensi p.q untuk memperoleh keluaran pola fitur (z) disetiap pergeserannya, secara matematis dapat dirumuskan sebagai berikut :

$$u_{i,j}^{(k)} = \max_{p,q \in P_{i,j}} z_{p,q}^{(k)} \dots\dots\dots (2.6)$$

Dimana :

$u_{i,j}^{(k)}$  = *Pooling*, proses reduksi *kernel* berdimensi i,j

$\max_{p,q \in P_{i,j}}$  = Fungsi *pooling*, mengambil nilai terbesar dari *kernel* berdimensi p.q terhadap nilai kernel pada dimensi i,j

$z_{p,q}^{(k)}$  = Fitur, objek reduksi *kernel* berdimensi p.q

*Average Pooling* merupakan proses reduksi yang akan melakukan pengambilan terhadap nilai rata-rata dari *kernel* yang berdimensi p.q pada blok spasial

$H.H$  untuk memperoleh keluaran pola fitur ( $z$ ) disetiap pergeserannya, secara matematis dapat dirumuskan sebagai berikut :

$$u_{i,j}^{(k)} = \frac{1}{H^2} \sum_{(p,q) \in P_{i,j}} z_{p,q}^{(k)} \dots\dots\dots (2.7)$$

$u_{i,j}^{(k)}$  = *Pooling*, proses reduksi *kernel* berdimensi  $i,j$

$\frac{1}{H^2} \sum_{(p,q) \in P_{i,j}}$  = Fungsi *pooling*, mengambil nilai rata dari *kernel* berdimensi  $p,q$  terhadap nilai *kernel* pada dimensi  $i, j$  pada blok spasial  $H.H$

$z_{p,q}^{(k)}$  = Fitur, objek reduksi *kernel* berdimensi  $p,q$

*Pooling* dapat diterapkan dalam berbagai *kernel size* dan pergeseran sesuai dengan ukuran dimensi citra yang dibutuhkan [13]. Dalam penerapannya, tiap *feature map* yang diperoleh dari proses *convolution*, haruslah diiringi dengan menerapkan proses *pooling*, menentukan *kernel size* dan menentukan pergeseran *kernel* (biasanya 1 atau 2), menggeser *kernel* terhadap gambar yang sedang mengalami proses filter, dan terakhir mengambil nilai rata-rata atau terbesar dari setiap proses [13] .

#### 2.6.5. Fully Connected Layer (Layer Klasifikasi)

Secara keseluruhan *fully connected layer* merupakan proses jaringan komputasi (sistem *node* dan interkoneksi antar *node*) yang terinspirasi oleh jaringan saraf biologis, yang merupakan jaringan kompleks *neural* di otak manusia. Dengan demikian, secara umum terdapat dua *hyperparameter* yang dapat dijadikan acuan dalam membangun *fully connected layer* ini, yaitu sistem *node* dan interkoneksi antar *node*. Setiap *node* yang terdapat dalam setiap sistem *node* akan memiliki

nilai yang biasa disebut dengan bobot dan antar *node* akan terkoneksi satu sama lain karena suatu nilai yang biasa disebut dengan *bias*. Dimana nilai *bias* diperoleh melalui beberapa penerapan fungsi matematis.

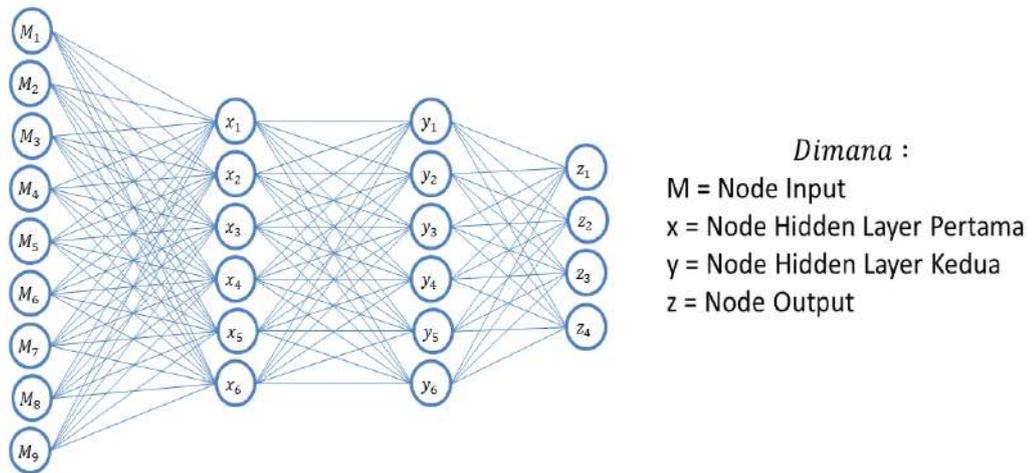
Setiap *hyperparameter* memiliki beberapa parameter yang perlu dipertimbangkan supaya dapat menghasilkan keluaran prediksi/klasifikasi yang tepat, seperti dijelaskan oleh Tabel 6 di bawah ini

Tabel 6 *Hyperparameter Fully Connected Layer* [14]

No	<i>Hyperparameter</i>	Parameter	Keterangan
1	Sistem <i>Node</i>	<i>Input Layer</i>	Lapisan yang menyimpan bobot masukan, tidak terjadi proses komputasi, yang terjadi adalah proses pengiriman sinyal ke lapisan <i>hidden</i> .
2		<i>Hidden Layer</i>	Lapisan yang merupakan tempat terjadi proses komputasi terhadap bobot dan <i>bias</i> .
3		<i>Output Layer</i>	Lapisan yang menyimpan bobot hasil (prediksi/klasifikasi) proses komputasi pada layer <i>hidden</i> .
4	Interkoneksi Antar <i>Node</i>	Nilai <i>Dropout</i>	Fungsi matematika yang digunakan untuk menentukan persentase pembuangan nilai yang dianggap paling buruk dari sejumlah nilai yang dihasilkan pada 1 tahap pembelajaran
5		Fungsi Aktivasi	Fungsi matematika yang berada pada <i>hidden layer</i> yang berfungsi untuk mengubah ANN menjadi bersifat <i>non linear</i> .
6		Teknik Optimasi	Digunakan untuk memilih nilai yang dianggap paling optimal dari sejumlah nilai pada proses pembelajaran.

No	Hyperparameter	Parameter	Keterangan
7	Interkoneksi Antar Node	<i>Batch_size</i>	Ketentuan nilai yang berisikan ketentuan dari banyaknya jumlah sampel yang akan di latih
8		<i>Epoch</i>	Ketentuan nilai yang berisikan banyaknya (berapa kali) data melakukan <i>feed forward</i> dan <i>backward</i> .

Arsitektur *fully connected layer* pada umumnya seperti diperlihatkan oleh Gambar 7 di bawah ini.



Gambar 7 Arsitektur *Artificial Neural Network* (ANN)

Merujuk pada Gambar 7 di atas, maka perhitungan hasil disetiap neural (*node*) diperlihatkan oleh persamaan berikut :

$$X_{(1...n)} = \sum_1^n M x w_x \dots\dots\dots (2.8)$$

$$Y_{(1...n)} = \sum_1^n X x w_y \dots\dots\dots (2.9)$$

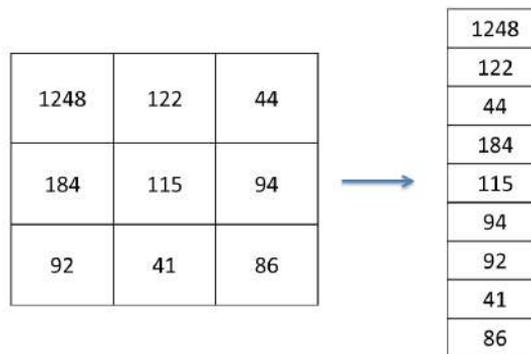
$$Z_{(1...n)} = \sum_1^n Y x w_z \dots\dots\dots (2.10)$$

Dimana :

$w$  = Nilai Bobot

$x$  = Perkalian Silang (*Cross Product*)

Input yang digunakan dalam *fully connected layer* ini dihasilkan melalui proses sebelumnya yang kemudian diratakan (*flattening*), seperti diperlihatkan oleh Gambar 8 di bawah ini.



Gambar 8 Proses *plattening* matriks citra

*Fully connected layer* akan menghasilkan output berupa vektor berukuran  $N$ . Dimana  $N$  merupakan jumlah kelas (prediksi) yang disesuaikan dengan kategori dataset yang digunakan (misalnya angka, nama hewan, nama benda).

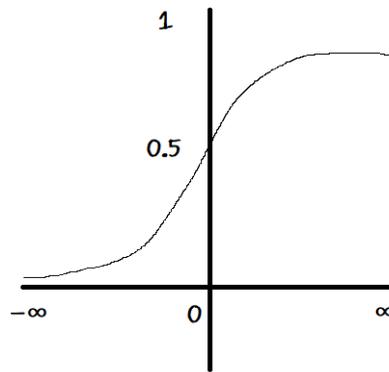
#### 2.6.6. Fungsi Aktivasi

Fungsi aktivasi berfungsi untuk mengendalikan bagaimana sinyal mengalir dari satu layer ke layer berikutnya, sedemikian rupa mengikuti cara kerja *neuron-neuron*. Sinyal *output* yang terkait dengan referensi sebelumnya akan mengaktifkan banyak neuron yang lain, sehingga sinyal dapat mengalami propogasi secara efisien [15]. Dalam kasus klasifikasi yang umumnya memiliki

*output* berupa *multiclass* ( lebih dari satu kelas) atau *multilabel* (lebih dari satu label), fungsi aktivasi yang umumnya sering digunakan ialah *Softmax* dan *Sigmoid*.

#### 2.6.6.1. *Softmax*

Fungsi *Softmax* adalah fungsi aktivasi untuk menghitung probabilitas dari sebuah kejadian. Fungsi *Softmax* menghasilkan nilai keseluruhan kelas antara 0 sampai 1. Secara grafik, diperlihatkan pada Gambar 9 di bawah ini.



Gambar 9 Grafik Fungsi Aktivasi *Softmax*

Fungsi *Softmax* umumnya digunakan dalam model multi-kelas dimana fungsi ini mengembalikan probabilitas masing-masing kelas, dengan kelas target yang memiliki probabilitas tertinggi [15]. Fungsi aktivasi ini dapat dihitung menggunakan persamaan berikut :

$$f(x)_i = \frac{e^{x_i}}{\sum_{j=1}^N e^{z_j}} \dots\dots\dots (2.11)$$

Dimana :

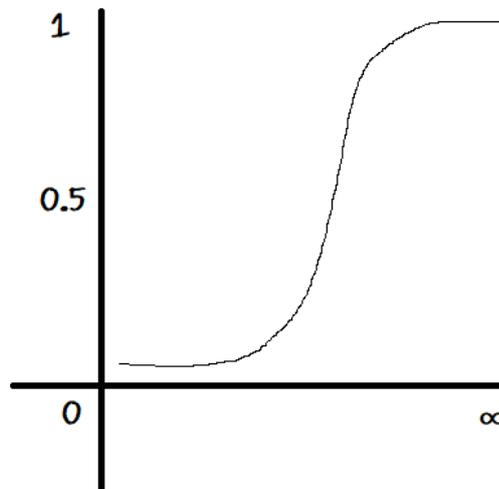
$x$  = vektor keluaran mentah dari jaringan syaraf

$e$  = Eksponensial, nilai  $e \approx 2,718$

$z$  = Vektor keluaran mentah dari jaringan saraf

### 2.6.6.2. Sigmoid

Fungsi *Sigmoid* adalah fungsi aktivasi untuk menghitung probabilitas dari sebuah kejadian. Fungsi *Sigmoid* menghasilkan nilai disetiap kelas antara 0 sampai 1. Secara grafik, diperlihatkan pada Gambar 10 di bawah ini.



Gambar 10 Grafik Fungsi Aktivasi *Sigmoid*

Fungsi *Sigmoid* umumnya digunakan dalam model multi-label dimana fungsi ini mengembalikan probabilitas masing-masing kelas, biasanya berupa persentase, dengan kelas target yang memiliki probabilitas tertinggi. Besar kemungkinan output lebih dari satu. Fungsi aktivasi ini dapat dihitung menggunakan persamaan berikut :

$$f(x) = \frac{1}{1+e^{-x}} \dots\dots\dots (2.12)$$

Dimana :

x = vektor keluaran mentah dari jaringan syaraf

e = Eksponensial, nilai  $e \approx 2,718$

## 2.7 Tahapan pembuatan sistem

### 2.7.1. Membangun sistem

Membangun sistem merupakan tahap eksekusi pembuatan sistem yang berfokus pada keberjalanan pembangunan (pelatihan dan validasi) sistem, melakukan perhitungan dan uji coba untuk menemukan atribut arsitekur model CNN yang telah dipaparkan pada Subbab 2.6 sebelumnya, beserta metrik pembangunan sistem berupa teknik optimasi dan fungsi kerugian (*loss function*) yang terbaik.

#### 2.7.1.1. Teknik Optimasi

Teknik Optimasi merupakan teknik yang digunakan untuk memperbaiki parameter bobot. Teknik optimasi ini bertujuan untuk meminimalisir angka *loss* yang diterima. Terdapat beberapa teknik optimasi yang seringkali digunakan, seperti diantaranya [15]:

##### 1. *Stochastic Gradient Descent* (SGD)

*Stochastic Gradient Descent* biasa disebut sebagai *incremental gradient descent*. Metode ini mencari suatu bobot baru dengan metode pengambilan salah satu data (secara acak) dari semua data pelatihan, setelah itu SGD melakukan Analisa dari setiap data yang diambil [15]. SGD beroperasi berdasarkan prinsip-prinsip yang terdefinisi dengan baik yang mengatur proses optimasi berulangnya. Adapun mekanisme SGD dalam memperbaiki parameter model dan menavigasi lanskap pengoptimalan adalah sebagai berikut :

#### Langkah 1 : Menginisialisasi parameter

- SDG dimulai dengan menginisialisasi parameter model secara acak atau menggunakan nilai yang telah ditentukan sebelumnya.
- Parameter ini mewakili bobot dan bias model yang akan disesuaikan selama proses optimasi.

#### Langkah 2 : Optimasi Iteratif.

- Pada setiap iterasi (atau *epoch*), SDG menghitung gradien fungsi kerugian terkait parameter model.
- SDG mengambil sampel kumpulan kecil titik data secara acak.
- Pengambilan sampel *stokastik* ini memperkenalkan keacakan ke dalam proses pengoptimalan, memungkinkan SGD keluar dari minimum lokal dan menjelajahi ruang solusi secara lebih dinamis.

#### Langkah 3 : Menghitung gradien.

- Setelah kumpulan kecil titik data dipilih, SGD menghitung gradien fungsi kerugian terkait parameter model.
- Gradien mewakili arah dan besarnya kenaikan paling curam dari fungsi kerugian.

#### Langkah 4 : Memperbarui parameter

- Dengan menggunakan gradien yang dihitung, SDG memperbarui parameter model ke arah yang meminimalisir fungsi kerugian.

- Besarnya pembaruan parameter ditentukan oleh kecepatan pembelajaran, yaitu *hyperparameter* yang mengontrol ukuran langkah proses pengoptimalan.
- Parameter yang diperbarui dihitung menggunakan persamaan berikut :

$$\theta_{i+1} = \theta_i - \eta \cdot \nabla J(\theta_i) \quad (2.13)$$

Dimana :

$\theta_{i+1}$  = Parameter bobot pada iterasi setelah iterasi ke-i (bobot baru)

$\theta_i$  = Parameter bobot pada iterasi i (bobot lama)

$\eta$  = Kecepatan pembelajaran (*Learning Rate*)

$\nabla J(\theta_i)$  = Gradien fungsi kerugian terhadap parameter bobot pada iterasi i

## 2. Adaptive Gradient Algorithm (Adagrad)

*Adagrad* merupakan suatu metode *adaptive learning rate*. Pada metode ini *learning rate* akan dijadikan parameter bobot. *Adagrad* menyesuaikan kecepatan pembelajaran untuk melakukan adaptasi *learning rate* berdasarkan gradien sebelumnya [15]. Prinsip utama AdaGrad adalah menskalakan kecepatan pembelajaran untuk setiap parameter berdasarkan total gradien kuadrat yang diamati selama proses pelatihan sistem. Adapun langkah-langkah optimasi Adagrad adalah sebagai berikut :

Langkah 1 : Menginisialisasi variabel

- Menginisialisasi parameter  $\theta$  dan konstanta kecil  $\varepsilon$  untuk menghindari pembagian dengan nol.
- Menginisialisasi jumlah variabel gradien kuadrat  $G$  dengan nol, yang bentuknya sama dengan  $\theta$ .

Langkah 2 : Menghitung gradient fungsi kerugian.

$$g_{i,t} = \nabla J(\theta_{i,t}) \quad (2.14)$$

Langkah 3 : Mengakumulasi (memperbarui jumlah) gradien kuadrat.

$$G_{i,t} = \sum_t (g_{i,t})^2 \quad (2.15)$$

Langkah 4 : Memperbarui setiap parameter menggunakan kecepatan pembelajaran adaptif.

$$\theta_{i+1} = \theta_i - \frac{\eta}{(\sqrt{G_{i,t} + \epsilon})} g_{i,t} \quad (2.16)$$

Dimana :

- $g_{i,t}$  = Gradien fungsi kerugian terhadap parameter bobot dari i ke t,
- $J(\theta_{i,t})$  = Fungsi Kerugian terhadap parameter bobot dari i ke t,
- $\sum_t$  = Penjumlahan dari awal pelatihan sampai sekarang (iterasi ke-t),
- $\theta_{i+1}$  = Parameter bobot pada iterasi setelah iterasi ke-i (bobot baru),
- $\theta_i$  = Parameter bobot pada iterasi ke-i (bobot lama),
- $\eta$  = Kecepatan pembelajaran (*Learning Rate*),
- $\epsilon$  = Konstanta kecil untuk mencegah pembagian dengan nol, biasanya diatur ke  $10^{-8}$

### 3. *Root Mean Square Propagation* (RMSProp)

Metode ini merupakan penyempurnaan dari kekurangan yang ada pada *Adagrad* dengan memakai *moving average* dari *gradient* kuadrat. Metode *RMSProp* memakai besarnya *gradient descent* terbaru guna menormalkan *gradient*. *Learning rate* pada metode ini diselaraskan dengan cara otomatis serta memakai *learning rate* yang berlainan untuk setiap parameter. *RMSProp* memisah *learning rate* dengan rata-rata keseluruhan eksponensial dari *gradient descent*

[15]. Algoritma *RMSProp* memperbarui parameter dengan langkah-langkah dan persamaan-persamaan berikut :

Langkah 1 : Menghitung Gradien :

$$g_{i,t} = \nabla J(\theta_{i,t}) \quad (2.17)$$

Langkah 2 : Mengakumulasi gradien kuadrat.

$$E[G_i]_t = \beta E[G_i]_{t-1} + (1 - \beta) g_{i,t}^2 \quad (2.18)$$

Langkah 3 : Menghitung kecepatan pembelajaran adaptif :

$$\eta_t = \frac{\eta}{\sqrt{(E[G_i]_t + \epsilon)}} \quad (2.19)$$

Langkah 4 : Memperbarui parameter

$$\theta_{t+1} = \theta_t - \eta_t g_{i,t} \quad (2.20)$$

Dimana :

$g_{i,t}$  = Gradien fungsi kerugian terhadap parameter bobot dari i ke t,

$J(\theta_{i,t})$  = Fungsi kerugian terhadap parameter bobot dari i ke t,

$E[G_i]_t$  = *Weighted Average*, Akumulasi gradient yang diperbaharui

$E[G_i]_{t-1}$  = *Weighted Average*, Akumulasi gradient yang diperbaharui

$\beta$  = Laju peluruhan, biasanya diatur ke 0.9

$\eta_t$  = Kecepatan Pembelajaran (*Learning Rate*) Adaptif

$\eta$  = Kecepatan pembelajaran (*Learning Rate*) awal

$\epsilon$  = Konstanta kecil untuk mencegah pembagian dengan nol,  
biasanya diatur ke  $10^{-8}$

$\theta_{t+1}$  = Parameter bobot baru

$\theta_t$  = Parameter bobot lama

Langkah-langkah ini diulang untuk setiap parameter dalam jaringan hingga konvergensi atau hingga jumlah iterasi maksimum tercapai [15].

#### 4. Adaptive Movement Estimation (Adam)

*Adam* merupakan pengembangan dari *Adagrad* serta kombinasi antara *RMSProp* dan *Stochastic Gradient Descent (SGD)* dengan momentum. Metode ini sangat efisien secara komputasi dan dalam prosesnya sedikit memori yang dibutuhkan. Adam optimizer salah satu algoritma *gradient descent* yang sering digunakan. Algoritma *Adam* memperbarui parameter dengan langkah-langkah dan persamaan-persamaan berikut :

Langkah 1 : Menghitung gradien

$$g_{i,t} = \nabla J(\theta_{i,t}) \quad (2.21)$$

Langkah 2 : Memperbarui bias momen pertama

$$m_{i,t} = \beta_1 m_{t-1} + (1 - \beta_1) g_{i,t} \quad (2.22)$$

Langkah 3 : Memperbarui bias momen kedua

$$v_{i,t} = \beta_2 v_{t-1} + (1 - \beta_2) (g_{i,t})^2 \quad (2.23)$$

Langkah 4 : Menghitung koreksi bias momen pertama

$$\hat{m}_{i,t} = \frac{m_{i,t}}{1 - \beta_1^t} \quad (2.24)$$

Langkah 5 : Menghitung koreksi bias momen kedua

$$\tilde{v}_{i,t} = \frac{v_{i,t}}{1 - \beta_2^t} \quad (2.25)$$

Langkah 6 : Memperbarui parameter

$$\theta_{i,t} = \theta_{i,t-1} \frac{\eta}{\sqrt{\tilde{v}_{i,t} + \epsilon}} \hat{m}_{i,t} \quad (2.26)$$

Dimana :

$g_{i,t}$  = Gradien fungsi kerugian terhadap parameter bobot dari i ke t,

$J(\theta_{i,t})$  = Fungsi kerugian terhadap parameter bobot dari i ke t,

$m_{i,t}$  = Momen estimasi pertama, *Running Average* dari gradient

$\beta_1$  = *Eksponensial delay rates* momen pertama, biasanya diatur 0.9

$m_{t-1}$  = Momen estimasi pertama yang sebelumnya

$v_{i,t}$  = Momen estimasi kedua, *Running Average* dari gradient kuadrat

- $\beta_2$  = Eksponensial delay rates momen kedua, biasanya diatur 0.999  
 $v_{t-1}$  = Momen estimasi kedua yang sebelumnya  
 $\hat{m}_{i,t}$  = koreksi bias momen estimasi pertama  
 $\tilde{v}_{i,t}$  = koreksi bias momen estimasi pertama  
 $\theta_{t+1}$  = Parameter bobot baru  
 $\theta_t$  = Parameter bobot lama  
 $\eta$  = Kecepatan pembelajaran (*Learning Rate*) awal

### 2.7.1.2. Fungsi Kerugian (*loss function*)

*Loss function* adalah fungsi yang mengukur seberapa bagus kinerja yang dihasilkan oleh model dalam prediksi terhadap target. Fungsi *loss* yang baik adalah fungsi yang menghasilkan *error* yang diharapkan paling rendah. Apabila model memiliki kelas yang cukup banyak perlu adanya cara untuk mengukur perbedaan antara probabilitas hasil hipotesis dan probabilitas hasil kebenaran asli. Berikut jenis *loss function* :

#### 1. *Mean Absolute Error* (MAE)

*Mean Absolute Error* (MAE), juga disebut *L1 Loss*, menghitung rata-rata jumlah perbedaan mutlak antara nilai aktual dan nilai prediksi. *Loss* ini memeriksa ukuran kesalahan dalam satu set nilai yang diprediksi, tanpa melihat arah positif atau negatifnya. Jika nilai absolut dari kesalahan tidak digunakan, maka nilai negatif dapat membatalkan nilai positif. MAE digunakan ketika distribusi variabel target memiliki *outlier*, seperti nilai-nilai kecil atau besar yang sangat jauh dari nilai rata-rata. Rumus persamaan *mean absolute error* adalah sebagai berikut :

$$\text{loss}(x, y) = |x - y| \quad (2.27)$$

Dimana :

x = Nilai asli

y = Nilai prediksi

## 2. Mean Squared Error (MSE)

*Mean Squared Error* (MSE), juga disebut *L2 Loss*, menghitung rata-rata perbedaan kuadrat antara nilai aktual dan nilai prediksi. Untuk meningkatkan akurasi model, *L2 loss* yang diperoleh haruslah sekecil mungkin atau mencapai nilai sempurna adalah 0,0. MSE digunakan saat melakukan regresi, dikondisikan pada input, terdistribusi normal, dan ingin kesalahan besar secara signifikan (kuadrat) lebih diberikan perhitungan daripada kesalahan kecil. Rumus persamaan *mean squared error* adalah sebagai berikut :

$$\text{loss}(x, y) = (x, y)^2 \quad (2.28)$$

Dimana :

x = Nilai asli

y = Nilai prediksi

## 3. Cross-Entropy

*Loss function* ini menghitung perbedaan antara dua distribusi probabilitas untuk serangkaian kejadian atau variabel acak yang disediakan. Ini digunakan untuk menghitung skor yang merangkum perbedaan rata-rata antara nilai yang diprediksi dan nilai sebenarnya. Untuk meningkatkan akurasi model, nilai *Cross entropy* yang diperoleh haruslah sekecil mungkin dan nilai sempurna adalah 0.

Fungsi *Cross-Entropy* memiliki variasi yang luas, jenis yang paling umum adalah *Binary Cross-Entropy* (BCE). *BCE Loss* terutama digunakan untuk model klasifikasi biner; yaitu, model hanya memiliki 2 kelas. Persamaan matematika dari *cross entropy* sebagai berikut

$$\text{loss}(x, y) = \sum x \cdot \log(y) \quad (2.29)$$

Dimana :

x = Nilai asli

y = Nilai prediksi

### 2.7.2. Menguji dan mengevaluasi Sistem

Menguji sistem merupakan tahap evaluasi pembuatan sistem yang berfokus pada keberhasilan pembangunan sistem (klasifikasi), melakukan pengujian, analisis dan evaluasi menggunakan beserta metrik kesalahan (*error*) sistem, *confusion matrix* berupa *accuracy*, *precision*, *recall*, dan *f1-score*

*Confusion matrix* adalah tabel yang menjelaskan hasil klasifikasi dari jumlah data uji yang benar dan jumlah data uji yang salah [15]. Juga merupakan salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode klasifikasi. Pada dasarnya, *Confusion Matrix* berisi informasi yang membandingkan hasil klasifikasi yang dibuat oleh sistem dengan hasil klasifikasi yang seharusnya [15].

*Accuracy* menggambarkan seberapa akurat model dalam mengklasifikasi dengan benar, secara matematis dirumuskan dengan persamaan sebagai berikut :

$$A = \frac{TP}{D} \quad (2.30)$$

Dimana :

A = Akurasi

$T_P$  = *True Positif*, Banyaknya data yang berhasil mengklasifikasi data dengan benar sesuai dengan kelasnya.

D = Jumlah Keseluruhan Data Asli

*Precision* menggambarkan keberhasilan klasifikasi antara data yang diminta (asli) dengan hasil prediksi yang diberikan oleh model, secara matematis dirumuskan dengan persamaan sebagai berikut :

$$P = \frac{T_P}{D_P} \quad (2.31)$$

Dimana :

P = Presisi

$T_P$  = *True Positif*, Banyaknya data yang berhasil mengklasifikasi data dengan benar sesuai dengan kelasnya.

$D_P$  = Jumlah Data Hasil Prediksi

*Recall* atau *sensitivity* menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi, secara matematis dirumuskan dengan persamaan sebagai berikut :

$$R = \frac{T_P}{D_r} \quad (2.32)$$

Dimana :

R = *Recall*

$T_P$  = *True Positif*, Banyaknya data yang berhasil mengklasifikasi data dengan benar sesuai dengan kelasnya.

$D_r$  = Jumlah Keseluruhan Pemanggilan (Data Asli/Kelas)

*F1-Score* menggambarkan perbandingan rata-rata *precision* dan *recall*, secara matematis dirumuskan dengan persamaan sebagai berikut :

$$F1 = \frac{2RP}{R+P} \quad (2.33)$$

Dimana :

F1 = *F1-Score*

R = *Recall*

P = *Precision*

## **BAB III. METODE PENELITIAN**

### **3.1 Waktu dan Tempat penelitian**

Penelitian ini dijalankan selama 8 bulan mulai pada bulan Oktober 2023 sampai Mei 2024 yang berlokasi di Universitas Lampung.

### **3.2 Alat dan Bahan**

Alat dan bahan yang digunakan pada penelitian tugas akhir ini adalah sebagai berikut :

1. Satu unit laptop
2. Perangkat lunak *Visual Studio Code*.
3. Bahasa pemrograman *Python 3.9.4. 64 bit*
4. Data-data gambar aksara lampung

### **3.3 Tahapan dan Metode Penelitian**

Adapun tahapan dan metode yang diterapkan dalam penelitian ini secara garis besar terdiri atas enam bagian, yaitu tahapan akuisisi data, *pre-processing data*, pembuatan model arsitektur CNN, pelatihan sistem, pengujian sistem dan analisis dan pembahasan.

### 3.3.1 Akuisisi *Dataset*.

Akuisisi *dataset* merupakan tahapan mempersiapkan *dataset* citra mentah (awal) dan *dataset* citra uji, berupa citra tulisan tangan aksara lampung dengan total keseluruhan data citra sebanyak 1732 citra. 1692 citra aksara untuk pembangunan model (data mentah, latih dan validasi) dan 40 citra aksara untuk pengujian model (data latih). Terdapat dua tahapan yang akan dilakukan, yaitu mengumpulkan data dan mengelompokkan data kedalam kelasnya (folder)

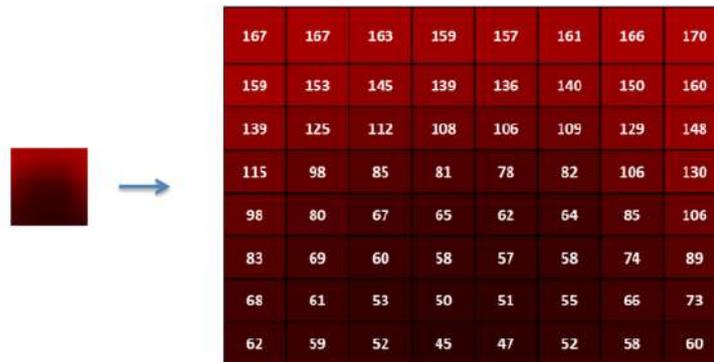
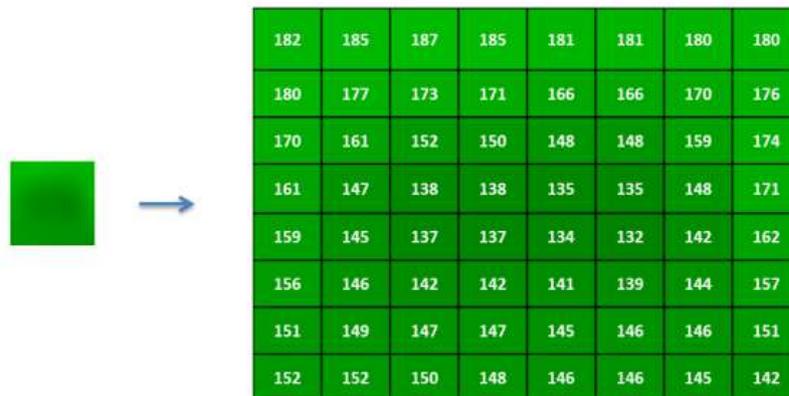
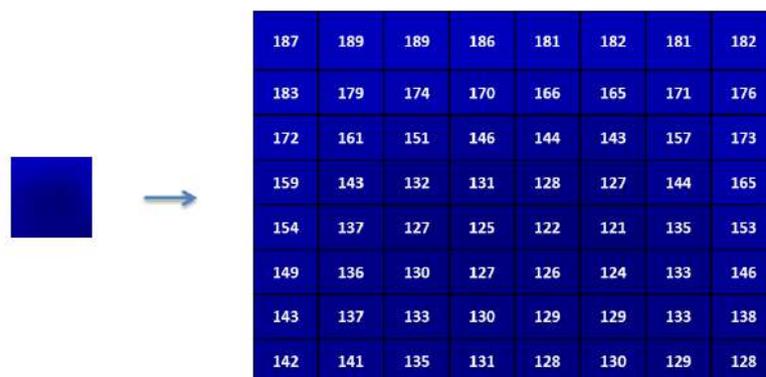
#### 3.3.1.1. Mengumpulkan Data Citra

Data yang akan digunakan dalam penelitian ini berupa variasi citra tulisan tangan empat kata aksara lampung yang ditulis secara manual menggunakan perantara kertas HVS putih 70 gram dan media tulis yang beragam. Variasi tulisan tangan aksara lampung kemudian difoto secara manual menggunakan kamera *handphone* dan menghasilkan citra warna (RGB). Adapun representasi salah satu bagian citra 8x8 disetiap *channel*-nya, diperlihatkan oleh Gambar 11 di bawah ini.



Gambar 11 Representasi Citra 8x8 kata 'Abadi' per Channel

Setiap *channel* merepresentasikan nilai intensitas (warna) disetiap pikselnya, sebagai diperlihatkan oleh Gambar 12, Gambar 13 dan Gambar 14 di bawah ini.

Gambar 12 Nilai intensitas citra 8x8 kata 'Abadi' *channel Red* (R)Gambar 13 Nilai intensitas citra 8x8 kata 'Abadi' *channel Green* (G)Gambar 14 Nilai intensitas citra 8x8 kata 'Abadi' *channel Blue* (B)

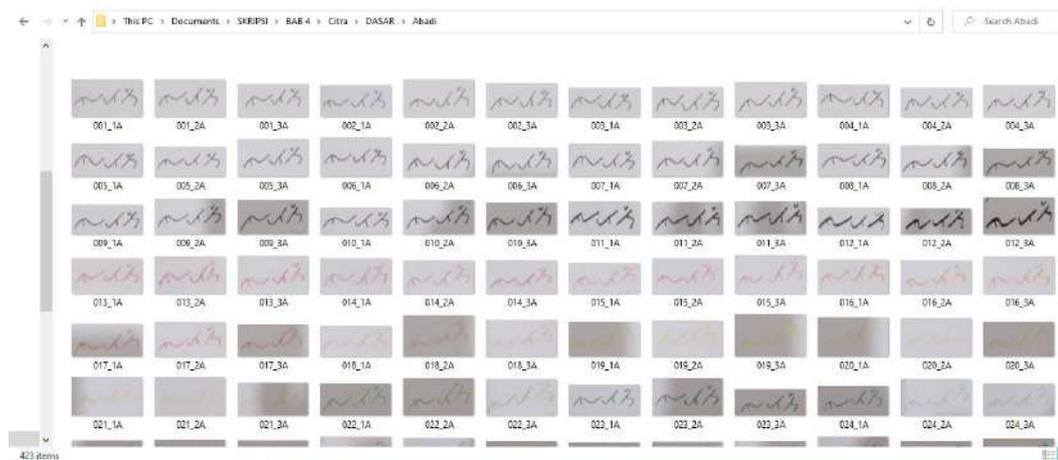
### 3.3.1.2. Mengelompokkan *Dataset* Citra

Data yang telah terkumpul kemudian dipotong, dikelompokkan dan dimasukkan ke dalam empat folder sesuai dengan jenis kata (*output*) secara manual. Sehingga menjadi *dataset* citra mentah (awal) aksara lampung yang kemudian dapat diolah lebih lanjut ke dalam program selanjutnya. Adapun penjelasan terkait pengelompokan dataset citra mentah aksara lampung diperlihatkan oleh Tabel 7 di bawah ini.

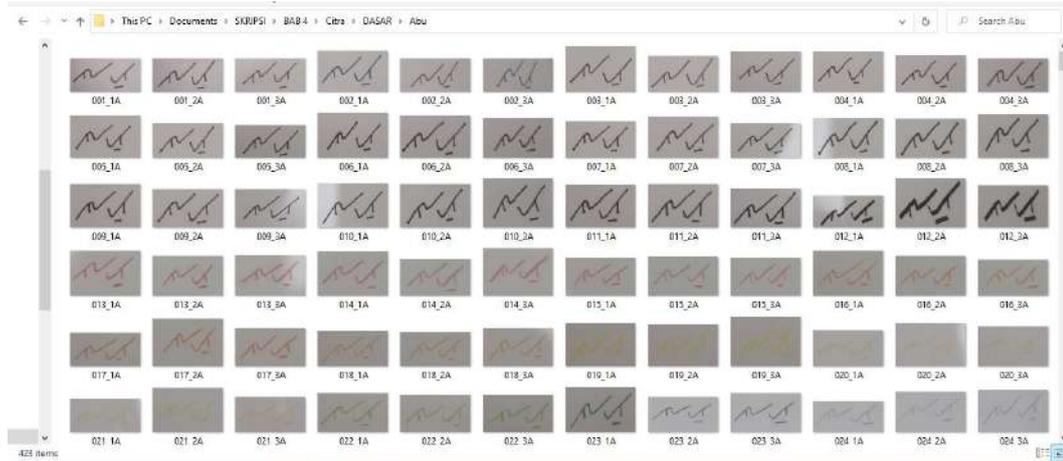
Tabel 7 Pengelompokan Dataset Citra

No	Nama folder kelas dataset citra Jenis aksara (data mentah)	Jumlah data citra
1	Abadi	423
2	Abu	423
3	Acara	423
4	Ada	423
<b>Total Dataset Citra</b>		1692

Adapun tampilan beberapa data citra mentah tiap kelas diperlihatkan oleh Gambar 15, Gambar 16, Gambar 17, dan Gambar 18 di bawah ini.



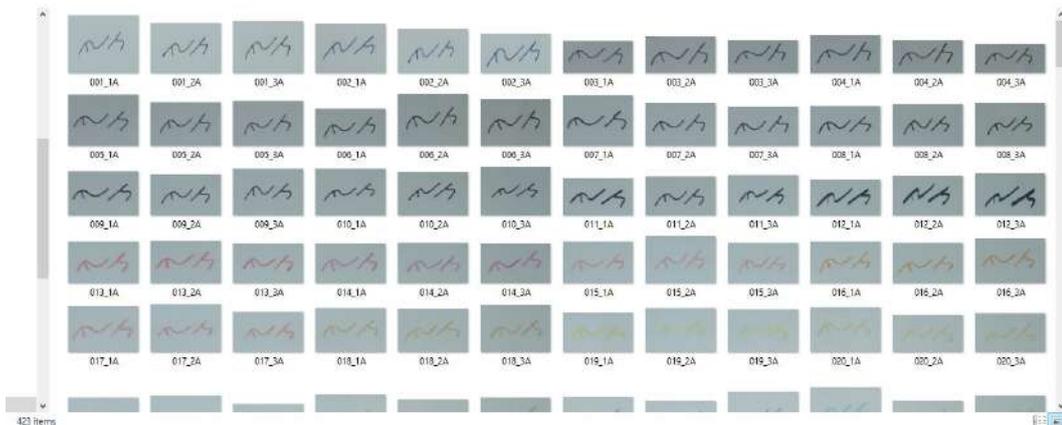
Gambar 15 Data citra mentah (awal) kelas 01 'Abadi'



Gambar 16 Data citra mentah (awal) kelas 02 'Abu'



Gambar 17 Data citra mentah (awal) kelas 03 'Acara'



Gambar 18 Data citra mentah (awal) kelas 04 'Ada'

### 3.3.2. *Pre-processing Data.*

*Pre-processing data* merupakan tahapan mempersiapkan data citra sebelum data citra diproses oleh model pembelajaran. Terdapat dua tahapan yang diimplementasikan dalam penelitian ini, yaitu normalisasi (tahapan penyetaraan) dan memodifikasi citra (*augmentasi*).

#### 3.3.2.1. Normalisasi *Dataset* Citra

*Dataset* citra mentah aksara lampung yang telah dikelompokkan sebelumnya, kemudian dimasukkan dan dibaca oleh program menggunakan *software visual studio code*, dengan bahasa pemrograman *python 3.9.4 64 bit*. Setelah berhasil terbaca oleh program, kemudian dilakukan normalisasi. Terdapat dua tahapan normalisasi, yaitu : normalisasi kuantitas (jumlah) dan normalisasi resolusi (ukuran) piksel.

Normalisasi kuantitas (jumlah) merupakan tahapan menyetarakan dan menyesuaikan banyaknya jumlah data citra disetiap proses pemodelan (pelatihan dan pengujian). Tahapan ini dilakukan dengan cara membagi data (*data splitting*) kedalam dua bagian, yaitu : data latih dan data validasi dengan perbandingan tertentu. Selain kedua jenis data yang digunakan untuk pembangunan model tersebut, terdapat juga satu bagian data yang digunakan untuk pengujian model, yaitu data uji.

Adapun penjelasan terkait pembagian tiga bagian *dataset* citra, dapat dilihat melalui Tabel 8 di bawah ini :

Tabel 8 Pembagian Dataset Citra

No.	Jenis Data	Jumlah
1.	Data Latih	80% Dari Keseluruhan Dataset Citra (1434 Citra)
2.	Data Validasi	20% Dari Keseluruhan Dataset Citra (358 Citra)
3.	Data Uji	40 Data citra baru (masing-masing kelas 10 Citra)

Adapun tampilan data citra uji tiap kelas diperlihatkan oleh Gambar 16, Gambar 19, Gambar 20, Gambar 21 dan Gambar 22 di bawah ini.



Gambar 19 Data citra uji kelas 01 'Abadi'



Gambar 20 Data citra uji kelas 02 'Abu'



Gambar 21 Data citra uji kelas 03 'Acara'



Gambar 22 Data citra uji kelas 04 'Ada'

Citra Latih merupakan citra yang akan digunakan dalam melatih program, yaitu pada proses *training* program. Citra validasi merupakan citra yang akan digunakan untuk memvalidasi keberhasilan sistem program bersama dengan citra latih pada proses *training program*. Citra Uji merupakan citra yang akan digunakan dalam pengujian program, berupa *testing* program.

Normalisasi resolusi (ukuran) merupakan tahapan menyetarakan setiap resolusi citra kedalam resolusi yang sama, dalam penelitian ini seluruh resolusi citra yang digunakan diubah menjadi 128x128 piksel.

### 3.3.2.2. *Augmentasi Data Citra*

*Augmentasi data* citra merupakan tahapan modifikasi *dataset* citra yang berfungsi menambah variasi citra dan meningkatkan akurasi citra. Dikarenakan data citra berupa tulisan tangan, maka proses yang akan dilakukan hanyalah dua, yaitu menggeser citra sebesar 0.2 dan memutar citra sebesar  $20^0$

### 3.3.3 Pembuatan Model Arsitektur CNN

Pembuatan model arsitektur CNN merupakan tahapan inti pembuatan sistem ini. Tahapan ini merupakan tahapan perancangan instruksi-instruksi program pengenalan dan transliterasi aksara Lampung baik pada proses latih maupun proses uji pada program menggunakan *software visual studio code* dengan bahasa pemrograman *python 3.9.4 64 bit*. Terdapat dua tahapan dalam pembuatan model arsitektur CNN, yaitu tahapan pengenalan (*feature extraction*) dan tahapan transliterasi (*classification*).

#### 3.3.3.1 Tahapan pengenalan (*feature extraction*)

Pada tahapan pertama, pengenalan (*feature extraction*), proses yang akan dilakukan adalah menentukan banyaknya proses konvolusi, jenis pooling pada setiap konvolusi, banyaknya filter pada setiap konvolusi dan *pooling*, ukuran *kernel/filter* pada setiap konvolusi, besarnya *stride* (pergeseran) pada setiap konvolusi dan *pooling*, dan fungsi aktivasi pada setiap konvolusi.

Adapun penjelasan proses pengenalan (*feature extraction*) lebih lanjut diperlihatkan oleh Tabel 9 di bawah ini :

Tabel 9 *Hyperparameter* tahapan pengenalan (*feature extraction*)

No.	Hyperparameter	Keterangan proses
1	Konvolusi	Dilakukan sebanyak 4 kali
2	<i>Pooling</i>	Dilakukan dengan jenis <i>Max Pooling</i>
3	<i>kernel_size</i>	Dilakukan dengan ukuran 3x3
4	<i>strides</i>	Bergeser sebanyak 1 piksel (konvolusi) dan sebanyak 2 piksel ( <i>pooling</i> )
5	Fungsi aktivasi	Dilakukan dengan <i>ReLU</i>

Berdasarkan keterangan *hyperparameter* proses yang diperlihatkan oleh Tabel 9 di atas. Merujuk pada Gambar 9, Gambar 10 dan Gambar 11. Nilai intensitas piksel Citra 8x8 pada masing-masing *channel* sebelumnya. Maka, proses pengenalan aksara lampung dapat direpresentasikan sebagai berikut

#### a. Convolution Layer

Konvolusi dilakukan pada masing-masing *channel* menggunakan *kernel\_size* 3x3, sama seperti yang telah diterapkan pada penelitian terdahulu [2][3] dan *strides* 1 piksel dengan representasi nilai seperti diperlihatkan pada Gambar 23 di bawah ini. Perhitungan di bawah ini merupakan contoh cara menghitung konvolusi pada *kernel* 8x8, dengan ukuran *kernel* 3x3.

-1	2	0
0	-1	-1
2	0	-1

Gambar 23 Representasi kernel ukuran 3x3

Perhitungan konvolusi *channel Red* (CR) :

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_1 = (167*-1) + (167*2) + (163*0) + (159*0) + (153*-1) + (145*-1) + (139*2) + (125*0) + (112*-1) = 35$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_2 = (167*-1) + (163*2) + (159*0) + (153*0) + (145*-1) + (139*-1) + (125*2) + (112*0) + (108*-1) = 17$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_3 = (163*-1) + (159*2) + (157*0) + (145*0) + (139*-1) + (136*-1) + (112*2) + (108*0) + (106*-1) = 98$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_4 = (159*-1) + (157*2) + (161*0) + (139*0) + (136*-1) + (140*-1) + (108*2) + (106*0) + (109*-1) = -16$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_5 &= (157 \cdot -1) + (161 \cdot 2) + (166 \cdot 0) + \\
 &\quad (136 \cdot 0) + (140 \cdot -1) + (150 \cdot -1) + \\
 &\quad (106 \cdot 2) + (109 \cdot 0) + (129 \cdot -1) \\
 &= -40
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_6 &= (161 \cdot -1) + (166 \cdot 2) + (170 \cdot 0) + \\
 &\quad (140 \cdot 0) + (150 \cdot -1) + (160 \cdot -1) + \\
 &\quad (109 \cdot 2) + (129 \cdot 0) + (148 \cdot -1) \\
 &= -69
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_7 &= (159 \cdot -1) + (153 \cdot 2) + (145 \cdot 0) + \\
 &\quad (139 \cdot 0) + (125 \cdot -1) + (112 \cdot -1) + \\
 &\quad (115 \cdot 2) + (98 \cdot 0) + (85 \cdot -1) \\
 &= 65
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_8 &= (153 \cdot -1) + (145 \cdot 2) + (139 \cdot 0) + \\
 &\quad (125 \cdot 0) + (112 \cdot -1) + (108 \cdot -1) + \\
 &\quad (98 \cdot 2) + (85 \cdot 0) + (82 \cdot -1) \\
 &= 31
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_9 &= (145*-1) + (139*2) + (136*0) + \\
 &\quad (112*0) + (108*-1) + (106*-1) + \\
 &\quad (85*2) + (81*0) + (78*-1) \\
 &= 11
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{10} &= (139*-1) + (136*2) + (140*0) + \\
 &\quad (108*0) + (106*-1) + (109*-1) + \\
 &\quad (81*2) + (78*0) + (82*-1) \\
 &= 2
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{11} &= (136*-1) + (140*2) + (150*0) + \\
 &\quad (106*0) + (109*-1) + (129*-1) + \\
 &\quad (78*2) + (82*0) + (106*-1) \\
 &= -34
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{12} &= (140*-1) + (150*2) + (160*0) + \\
 &\quad (109*0) + (129*-1) + (148*-1) + \\
 &\quad (82*2) + (106*0) + (130*-1) \\
 &= -28
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{13} &= (139*-1) + (125*2) + (112*0) + \\
 &\quad (115*0) + (98*-1) + (85*-1) + \\
 &\quad (98*2) + (80*0) + (67*-1) \\
 &= 57
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{14} &= (125*-1) + (112*2) + (108*0) + \\
 &\quad (98*0) + (85*-1) + (81*-1) + \\
 &\quad (80*2) + (67*0) + (65*-1) \\
 &= 28
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{15} &= (112*-1) + (108*2) + (106*0) + \\
 &\quad (85*0) + (81*-1) + (78*-1) + \\
 &\quad (67*2) + (65*0) + (62*-1) \\
 &= 80
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{16} &= (108*-1) + (106*2) + (109*0) + \\
 &\quad (81*0) + (78*-1) + (82*-1) + \\
 &\quad (65*2) + (62*0) + (64*-1) \\
 &= 10
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{17} &= (106 * -1) + (109 * 2) + (129 * 0) + \\
 &\quad (78 * 0) + (82 * -1) + (106 * -1) + \\
 &\quad (62 * 2) + (64 * 0) + (85 * -1) \\
 &= -39
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{18} &= (109 * -1) + (129 * 2) + (148 * 0) + \\
 &\quad (82 * 0) + (106 * -1) + (130 * -1) + \\
 &\quad (64 * 2) + (85 * 0) + (106 * -1) \\
 &= -35
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{19} &= (115 * -1) + (98 * 2) + (85 * 0) + \\
 &\quad (98 * 0) + (80 * -1) + (67 * -1) + \\
 &\quad (83 * 2) + (69 * 0) + (60 * -1) \\
 &= 40
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{20} &= (98 * -1) + (85 * 2) + (81 * 0) + \\
 &\quad (80 * 0) + (67 * -1) + (65 * -1) + \\
 &\quad (69 * 2) + (60 * 0) + (58 * -1) \\
 &= 20
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{21} &= (85 * -1) + (81 * 2) + (78 * 0) + \\
 &\quad (67 * 0) + (65 * -1) + (62 * -1) + \\
 &\quad (60 * 2) + (58 * 0) + (57 * -1) \\
 &= 13
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{22} &= (81 * -1) + (78 * 2) + (82 * 0) + \\
 &\quad (65 * 0) + (62 * -1) + (64 * -1) + \\
 &\quad (58 * 2) + (57 * 0) + (58 * -1) \\
 &= 7
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{23} &= (78 * -1) + (82 * 2) + (106 * 0) + \\
 &\quad (62 * 0) + (64 * -1) + (85 * -1) + \\
 &\quad (57 * 2) + (58 * 0) + (74 * -1) \\
 &= -13
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{24} &= (82 * -1) + (106 * 2) + (130 * 0) + \\
 &\quad (64 * 0) + (85 * -1) + (106 * -1) + \\
 &\quad (58 * 2) + (74 * 0) + (89 * -1) \\
 &= 66
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{25} = (98 * -1) + (80 * 2) + (67 * 0) + (83 * 0) + (69 * -1) + (60 * -1) + (68 * 2) + (61 * 0) + (53 * -1) = 16$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{26} = (80 * -1) + (67 * 2) + (65 * 0) + (69 * 0) + (60 * -1) + (58 * -1) + (61 * 2) + (53 * 0) + (50 * -1) = 8$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{27} = (67 * -1) + (65 * 2) + (62 * 0) + (60 * 0) + (58 * -1) + (57 * -1) + (53 * 2) + (50 * 0) + (51 * -1) = 3$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{28} = (65 * -1) + (62 * 2) + (64 * 0) + (58 * 0) + (57 * -1) + (58 * -1) + (50 * 2) + (51 * 0) + (55 * -1) = -11$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{29} &= (62 * -1) + (64 * 2) + (85 * 0) + \\
 &\quad (57 * 0) + (58 * -1) + (74 * -1) + \\
 &\quad (51 * 2) + (55 * 0) + (66 * -1) \\
 &= -30
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{30} &= (64 * -1) + (85 * 2) + (106 * 0) + \\
 &\quad (58 * 0) + (74 * -1) + (89 * -1) + \\
 &\quad (55 * 2) + (66 * 0) + (73 * -1) \\
 &= -20
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{31} &= (83 * -1) + (69 * 2) + (60 * 0) + \\
 &\quad (68 * 0) + (61 * -1) + (53 * -1) + \\
 &\quad (62 * 2) + (59 * 0) + (52 * -1) \\
 &= 13
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$\begin{aligned}
 CR_{32} &= (69 * -1) + (60 * 2) + (58 * 0) + \\
 &\quad (61 * 0) + (53 * -1) + (50 * -1) + \\
 &\quad (59 * 2) + (52 * 0) + (45 * -1) \\
 &= 21
 \end{aligned}$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{33} = (60 * -1) + (58 * 2) + (57 * 0) + (53 * 0) + (50 * -1) + (51 * -1) + (52 * 2) + (45 * 0) + (47 * -1) = 12$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{34} = (58 * -1) + (57 * 2) + (58 * 0) + (50 * 0) + (51 * -1) + (55 * -1) + (45 * 2) + (47 * 0) + (52 * -1) = -12$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{35} = (57 * -1) + (58 * 2) + (74 * 0) + (51 * 0) + (55 * -1) + (66 * -1) + (47 * 2) + (52 * 0) + (58 * -1) = -26$$

167	167	163	159	157	161	166	170
159	153	145	139	136	140	150	160
139	125	112	108	106	109	129	148
115	98	85	81	78	82	106	130
98	80	67	65	62	64	85	106
83	69	60	58	57	58	74	89
68	61	53	50	51	55	66	73
62	59	52	45	47	52	58	60

$$CR_{36} = (58 * -1) + (74 * 2) + (89 * 0) + (55 * 0) + (66 * -1) + (73 * -1) + (52 * 2) + (58 * 0) + (60 * -1) = -5$$

Dimana :

\* = Perkalian titik

Hasil perhitungan konvolusi *channel Red* (CR) :

35	17	98	-16	-40	-69
65	31	11	2	-34	-28
57	28	80	10	-39	-35
40	20	13	7	-13	66
15	8	3	-11	-30	-20
13	21	12	-12	-26	-5

Perhitungan konvolusi *channel Green* (CG) :

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_1 = (182 * -1) + (185 * 2) + (187 * 0) + (180 * 0) + (177 * -1) + (173 * -1) + (170 * 2) + (161 * 0) + (152 * -1) = 136$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_2 = (185 * -1) + (187 * 2) + (185 * 0) + (177 * 0) + (173 * -1) + (171 * -1) + (161 * 2) + (152 * 0) + (150 * -1) = 17$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_3 = (187 * -1) + (185 * 2) + (181 * 0) + (173 * 0) + (171 * -1) + (166 * -1) + (152 * 2) + (150 * 0) + (148 * -1) = 2$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_4 = (185 * -1) + (181 * 2) + (181 * 0) + (171 * 0) + (166 * -1) + (166 * -1) + (150 * 2) + (148 * 0) + (148 * -1) = -3$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_5 = (181 * -1) + (181 * 2) + (180 * 0) + (166 * 0) + (166 * -1) + (170 * -1) + (148 * 2) + (148 * 0) + (159 * -1) = -18$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_6 = (181 * -1) + (180 * 2) + (180 * 0) + (166 * 0) + (170 * -1) + (176 * -1) + (148 * 2) + (159 * 0) + (174 * -1) = -44$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_7 = (180 * -1) + (177 * 2) + (173 * 0) + (170 * 0) + (161 * -1) + (152 * -1) + (161 * 2) + (147 * 0) + (138 * -1) = 45$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_8 = (177*-1) + (173*2) + (171*0) + (161*0) + (152*-1) + (150*-1) + (147*2) + (138*0) + (138*-1) = 23$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_9 = (173*-1) + (171*2) + (166*0) + (152*0) + (150*-1) + (148*-1) + (138*2) + (138*0) + (135*-1) = 12$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{10} = (171*-1) + (166*2) + (166*0) + (150*0) + (148*-1) + (148*-1) + (138*2) + (135*0) + (135*-1) = 6$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{11} = (166*-1) + (166*2) + (170*0) + (148*0) + (148*-1) + (159*-1) + (135*2) + (135*0) + (148*-1) = -19$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{12} = (166 \cdot -1) + (170 \cdot 2) + (176 \cdot 0) + (148 \cdot 0) + (159 \cdot -1) + (174 \cdot -1) + (135 \cdot 2) + (148 \cdot 0) + (171 \cdot -1) = -60$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{13} = (170 \cdot -1) + (161 \cdot 2) + (152 \cdot 0) + (161 \cdot 0) + (147 \cdot -1) + (138 \cdot -1) + (159 \cdot 2) + (145 \cdot 0) + (137 \cdot -1) = 48$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{14} = (161 \cdot -1) + (152 \cdot 2) + (150 \cdot 0) + (147 \cdot 0) + (138 \cdot -1) + (138 \cdot -1) + (145 \cdot 2) + (137 \cdot 0) + (137 \cdot -1) = 20$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{15} = (152 \cdot -1) + (150 \cdot 2) + (148 \cdot 0) + (138 \cdot 0) + (138 \cdot -1) + (135 \cdot -1) + (137 \cdot 2) + (137 \cdot 0) + (134 \cdot -1) = 15$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{16} = (150 * -1) + (148 * 2) + (148 * 0) + (138 * 0) + (135 * -1) + (135 * -1) + (137 * 2) + (134 * 0) + (132 * -1) = 18$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{17} = (148 * -1) + (148 * 2) + (159 * 0) + (135 * 0) + (135 * -1) + (148 * -1) + (134 * 2) + (132 * 0) + (142 * -1) = -9$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{18} = (148 * -1) + (159 * 2) + (174 * 0) + (135 * 0) + (148 * -1) + (171 * -1) + (132 * 2) + (142 * 0) + (162 * -1) = -47$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{19} = (161 * -1) + (147 * 2) + (138 * 0) + (159 * 0) + (145 * -1) + (137 * -1) + (156 * 2) + (146 * 0) + (142 * -1) = 21$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{20} = (147*-1) + (138*2) + (138*0) + (145*0) + (137*-1) + (137*-1) + (146*2) + (142*0) + (142*-1) = 5$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{21} = (138*-1) + (138*2) + (135*0) + (137*0) + (137*-1) + (134*-1) + (142*2) + (142*0) + (141*-1) = 8$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{22} = (138*-1) + (135*2) + (135*0) + (137*0) + (134*-1) + (132*-1) + (142*2) + (141*0) + (139*-1) = 11$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{23} = (135*-1) + (135*2) + (148*0) + (134*0) + (132*-1) + (142*-1) + (141*2) + (139*0) + (144*-1) = -1$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{24} = (135 \cdot -1) + (148 \cdot 2) + (171 \cdot 0) + (132 \cdot 0) + (142 \cdot -1) + (162 \cdot -1) + (139 \cdot 2) + (144 \cdot 0) + (157 \cdot -1) = -22$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{25} = (159 \cdot -1) + (145 \cdot 2) + (137 \cdot 0) + (156 \cdot 0) + (146 \cdot -1) + (142 \cdot -1) + (151 \cdot 2) + (149 \cdot 0) + (147 \cdot -1) = -2$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{26} = (145 \cdot -1) + (137 \cdot 2) + (137 \cdot 0) + (146 \cdot 0) + (142 \cdot -1) + (142 \cdot -1) + (149 \cdot 2) + (147 \cdot 0) + (147 \cdot -1) = -4$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{27} = (137 \cdot -1) + (137 \cdot 2) + (134 \cdot 0) + (142 \cdot 0) + (142 \cdot -1) + (141 \cdot -1) + (147 \cdot 2) + (147 \cdot 0) + (145 \cdot -1) = 3$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{28} = (137*-1) + (134*2) + (132*0) + (142*0) + (141*-1) + (139*-1) + (147*2) + (145*0) + (146*-1) = -1$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{29} = (134*-1) + (132*2) + (142*0) + (141*0) + (139*-1) + (144*-1) + (145*2) + (146*0) + (146*-1) = -1$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{30} = (132*-1) + (142*2) + (162*0) + (139*0) + (144*-1) + (157*-1) + (146*2) + (146*0) + (151*-1) = -11$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{31} = (156*-1) + (146*2) + (142*0) + (151*0) + (149*-1) + (147*-1) + (152*2) + (152*0) + (150*-1) = -6$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{32} = (146*-1) + (142*2) + (142*0) + (149*0) + (147*-1) + (147*-1) + (152*2) + (150*0) + (148*-1) = 0$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{33} = (142*-1) + (142*2) + (141*0) + (147*0) + (147*-1) + (145*-1) + (150*2) + (148*0) + (146*-1) = -2$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{34} = (142*-1) + (141*2) + (139*0) + (147*0) + (145*-1) + (146*-1) + (148*2) + (146*0) + (146*-1) = -1$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{35} = (141*-1) + (139*2) + (144*0) + (145*0) + (146*-1) + (146*-1) + (146*2) + (146*0) + (145*-1) = -8$$

182	185	187	185	181	181	180	180
180	177	173	171	166	166	170	176
170	161	152	150	148	148	159	174
161	147	138	138	135	135	148	171
159	145	137	137	134	132	142	162
156	146	142	142	141	139	144	157
151	149	147	147	145	146	146	151
152	152	150	148	146	146	145	142

$$CG_{36} = (139 * -1) + (144 * 2) + (157 * 0) + (146 * 0) + (146 * -1) + (151 * -1) + (146 * 2) + (145 * 0) + (142 * -1) = -2$$

Dimana :

\* = Perkalian titik

Hasil perhitungan konvolusi *channel Green* (CG) :

137	17	2	-3	-18	-44
45	23	12	6	-19	-60
48	20	15	18	-9	-47
21	5	8	11	-1	-22
-2	-4	3	-1	-1	11
-6	0	-2	-1	-8	-2

Perhitungan konvolusi *channel Blue* (CB) :

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_1 = (187 * -1) + (189 * 2) + (189 * 0) + (183 * 0) + (179 * -1) + (174 * -1) + (172 * 2) + (161 * 0) + (151 * -1) = 131$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_2 = (189 \cdot -1) + (189 \cdot 2) + (186 \cdot 0) + (179 \cdot 0) + (174 \cdot -1) + (170 \cdot -1) + (161 \cdot 2) + (151 \cdot 0) + (146 \cdot -1) = 21$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_3 = (189 \cdot -1) + (186 \cdot 2) + (181 \cdot 0) + (174 \cdot 0) + (170 \cdot -1) + (166 \cdot -1) + (151 \cdot 2) + (146 \cdot 0) + (144 \cdot -1) = 5$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_4 = (186 \cdot -1) + (181 \cdot 2) + (182 \cdot 0) + (170 \cdot 0) + (166 \cdot -1) + (165 \cdot -1) + (146 \cdot 2) + (144 \cdot 0) + (143 \cdot -1) = -6$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_5 = (181 \cdot -1) + (182 \cdot 2) + (181 \cdot 0) + (166 \cdot 0) + (165 \cdot -1) + (171 \cdot -1) + (144 \cdot 2) + (143 \cdot 0) + (157 \cdot -1) = -22$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_6 &= (182 \cdot -1) + (181 \cdot 2) + (182 \cdot 0) + \\
 &\quad (165 \cdot 0) + (171 \cdot -1) + (176 \cdot -1) + \\
 &\quad (143 \cdot 2) + (157 \cdot 0) + (173 \cdot -1) \\
 &= -54
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_7 &= (183 \cdot -1) + (179 \cdot 2) + (174 \cdot 0) + \\
 &\quad (172 \cdot 0) + (161 \cdot -1) + (151 \cdot -1) + \\
 &\quad (159 \cdot 2) + (143 \cdot 0) + (132 \cdot -1) \\
 &= 49
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_8 &= (179 \cdot -1) + (174 \cdot 2) + (170 \cdot 0) + \\
 &\quad (161 \cdot 0) + (151 \cdot -1) + (146 \cdot -1) + \\
 &\quad (143 \cdot 2) + (132 \cdot 0) + (131 \cdot -1) \\
 &= 67
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_9 &= (174 \cdot -1) + (170 \cdot 2) + (166 \cdot 0) + \\
 &\quad (151 \cdot 0) + (146 \cdot -1) + (144 \cdot -1) + \\
 &\quad (132 \cdot 2) + (131 \cdot 0) + (128 \cdot -1) \\
 &= 12
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{10} = (170 * -1) + (166 * 2) + (165 * 0) + (146 * 0) + (144 * -1) + (143 * -1) + (131 * 2) + (128 * 0) + (127 * -1) = 10$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{11} = (166 * -1) + (165 * 2) + (171 * 0) + (144 * 0) + (143 * -1) + (157 * -1) + (128 * 2) + (127 * 0) + (144 * -1) = -24$$

⊞ ⊞ ⊞ ⊞

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{12} = (165 * -1) + (171 * 2) + (176 * 0) + (143 * 0) + (157 * -1) + (173 * -1) + (127 * 2) + (144 * 0) + (165 * -1) = -61$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{13} = (172 * -1) + (161 * 2) + (151 * 0) + (159 * 0) + (143 * -1) + (132 * -1) + (154 * 2) + (137 * 0) + (127 * -1) = 56$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{14} = (161 \cdot -1) + (151 \cdot 2) + (146 \cdot 0) + (143 \cdot 0) + (132 \cdot -1) + (131 \cdot -1) + (137 \cdot 2) + (127 \cdot 0) + (125 \cdot -1) = 27$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{15} = (151 \cdot -1) + (146 \cdot 2) + (144 \cdot 0) + (132 \cdot 0) + (131 \cdot -1) + (128 \cdot -1) + (127 \cdot 2) + (125 \cdot 0) + (122 \cdot -1) = 14$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{16} = (146 \cdot -1) + (144 \cdot 2) + (143 \cdot 0) + (131 \cdot 0) + (128 \cdot -1) + (127 \cdot -1) + (125 \cdot 2) + (122 \cdot 0) + (121 \cdot -1) = 16$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{17} = (144 \cdot -1) + (143 \cdot 2) + (157 \cdot 0) + (128 \cdot 0) + (127 \cdot -1) + (144 \cdot -1) + (122 \cdot 2) + (121 \cdot 0) + (135 \cdot -1) = -20$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{18} &= (143 \cdot -1) + (157 \cdot 2) + (173 \cdot 0) + \\
 &\quad (127 \cdot 0) + (144 \cdot -1) + (165 \cdot -1) + \\
 &\quad (121 \cdot 2) + (135 \cdot 0) + (153 \cdot -1) \\
 &= -49
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{19} &= (159 \cdot -1) + (143 \cdot 2) + (132 \cdot 0) + \\
 &\quad (154 \cdot 0) + (137 \cdot -1) + (127 \cdot -1) + \\
 &\quad (149 \cdot 2) + (136 \cdot 0) + (130 \cdot -1) \\
 &= 31
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{20} &= (143 \cdot -1) + (132 \cdot 2) + (131 \cdot 0) + \\
 &\quad (137 \cdot 0) + (127 \cdot -1) + (125 \cdot -1) + \\
 &\quad (136 \cdot 2) + (130 \cdot 0) + (127 \cdot -1) \\
 &= 14
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{21} &= (132 \cdot -1) + (131 \cdot 2) + (128 \cdot 0) + \\
 &\quad (127 \cdot 0) + (125 \cdot -1) + (122 \cdot -1) + \\
 &\quad (130 \cdot 2) + (127 \cdot 0) + (126 \cdot -1) \\
 &= 17
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{22} &= (131 \cdot -1) + (128 \cdot 2) + (127 \cdot 0) + \\
 &\quad (125 \cdot 0) + (122 \cdot -1) + (121 \cdot -1) + \\
 &\quad (127 \cdot 2) + (126 \cdot 0) + (124 \cdot -1) \\
 &= 9
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{23} &= (128 \cdot -1) + (127 \cdot 2) + (144 \cdot 0) + \\
 &\quad (122 \cdot 0) + (121 \cdot -1) + (135 \cdot -1) + \\
 &\quad (126 \cdot 2) + (124 \cdot 0) + (133 \cdot -1) \\
 &= -11
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{24} &= (127 \cdot -1) + (144 \cdot 2) + (165 \cdot 0) + \\
 &\quad (121 \cdot 0) + (135 \cdot -1) + (153 \cdot -1) + \\
 &\quad (124 \cdot 2) + (133 \cdot 0) + (146 \cdot -1) \\
 &= -15
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$\begin{aligned}
 CB_{25} &= (154 \cdot -1) + (137 \cdot 2) + (127 \cdot 0) + \\
 &\quad (149 \cdot 0) + (136 \cdot -1) + (130 \cdot -1) + \\
 &\quad (143 \cdot 2) + (137 \cdot 0) + (133 \cdot -1) \\
 &= 7
 \end{aligned}$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{26} = (137*-1) + (127*2) + (125*0) + (136*0) + (130*-1) + (127*-1) + (137*2) + (133*0) + (130*-1) = 4$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{27} = (127*-1) + (125*2) + (122*0) + (130*0) + (127*-1) + (126*-1) + (133*2) + (130*0) + (129*-1) = 17$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{28} = (125*-1) + (122*2) + (121*0) + (127*0) + (126*-1) + (124*-1) + (130*2) + (129*0) + (129*-1) = 0$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{29} = (122*-1) + (121*2) + (135*0) + (126*0) + (124*-1) + (133*-1) + (129*2) + (129*0) + (133*-1) = -12$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{30} = (121 * -1) + (135 * 2) + (153 * 0) + (124 * 0) + (133 * -1) + (146 * -1) + (129 * 2) + (133 * 0) + (138 * -1) = -10$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{31} = (149 * -1) + (136 * 2) + (130 * 0) + (143 * 0) + (137 * -1) + (133 * -1) + (142 * 2) + (141 * 0) + (135 * -1) = 2$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{32} = (136 * -1) + (130 * 2) + (127 * 0) + (137 * 0) + (133 * -1) + (130 * -1) + (141 * 2) + (135 * 0) + (131 * -1) = 12$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{33} = (130 * -1) + (127 * 2) + (126 * 0) + (133 * 0) + (130 * -1) + (129 * -1) + (135 * 2) + (131 * 0) + (128 * -1) = 7$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{34} = (127 * -1) + (126 * 2) + (124 * 0) + (130 * 0) + (129 * -1) + (129 * -1) + (131 * 2) + (128 * 0) + (130 * -1) = -31$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{35} = (126 * -1) + (124 * 2) + (133 * 0) + (129 * 0) + (129 * -1) + (133 * -1) + (128 * 2) + (130 * 0) + (129 * -1) = -13$$

187	189	189	186	181	182	181	182
183	179	174	170	166	165	171	176
172	161	151	146	144	143	157	173
159	143	132	131	128	127	144	165
154	137	127	125	122	121	135	153
149	136	130	127	126	124	133	146
143	137	133	130	129	129	133	138
142	141	135	131	128	130	129	128

$$CB_{36} = (124 * -1) + (133 * 2) + (146 * 0) + (129 * 0) + (133 * -1) + (138 * -1) + (130 * 2) + (129 * 0) + (128 * -1) = 3$$

Dimana :

\* = Perkalian titik

Hasil perhitungan konvolusi *channel Blue* (CB) :

131	21	5	-6	-22	-54
49	67	12	10	-24	-61
56	27	14	16	-20	-49
31	14	17	9	-11	-15
7	4	17	0	-12	-10
2	12	7	-31	-13	3

b. *ReLU Layer*

*ReLU* dilakukan pada masing-masing *channel* dengan representasi perhitungan sebagai berikut :

Hasil perhitungan *ReLU channel Red (R)* :

35	17	98	-16	-40	-69
65	31	11	2	-34	-28
57	28	80	10	-39	-35
40	20	13	7	-13	66
15	8	3	-11	-30	-20
13	21	12	-12	-26	-5

→ ReLU →

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

Hasil perhitungan *ReLU channel Green (G)* :

137	17	2	-3	-18	-44
45	23	12	6	-19	-60
48	20	15	18	-9	-47
21	5	8	11	-1	-22
-2	-4	3	-1	-1	11
-6	0	-2	-1	-8	-2

→ ReLU →

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

Hasil perhitungan *ReLU channel Blue (B)* :

131	21	5	-6	-22	-54
49	67	12	10	-24	-61
56	27	14	16	-20	-49
31	14	17	9	-11	-15
7	4	17	0	-12	-10
2	12	7	-31	-13	3

→ ReLU →

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

### b. Pooling Layer

Pooling dilakukan pada masing-masing *channel* menggunakan *Max\_Pooling*, *kernel\_size* 3x3 dan *strides* 2 piksel dengan representasi perhitungan sebagai berikut :

Perhitungan *Max\_Pooling channel Red* (PR) :

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_1 = \text{Max} \begin{bmatrix} 35 & 17 & 98 \\ 65 & 31 & 11 \\ 57 & 28 & 80 \end{bmatrix} = 98$$

98		

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_2 = \text{Max} \begin{bmatrix} 98 & 0 & 0 \\ 11 & 2 & 0 \\ 80 & 10 & 0 \end{bmatrix} = 98$$

98	98	

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_3 = \text{Max} \begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \\ 10 & 0 & 0 \end{bmatrix} = 10$$

98	98	10

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_4 = \text{Max} \begin{bmatrix} 57 & 28 & 80 \\ 40 & 20 & 13 \\ 15 & 8 & 3 \end{bmatrix} = 80$$

98	98	10
80		

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_5 = \text{Max} \begin{bmatrix} 80 & 10 & 0 \\ 13 & 7 & 0 \\ 3 & 0 & 0 \end{bmatrix} = 80$$

98	98	10
80	80	

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_6 = \text{Max} \begin{bmatrix} 10 & 0 & 0 \\ 7 & 0 & 66 \\ 0 & 0 & 0 \end{bmatrix} = 66$$

98	98	10
80	80	66

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_7 = \text{Max} \begin{bmatrix} 40 & 20 & 13 \\ 15 & 8 & 3 \\ 13 & 21 & 12 \end{bmatrix} = 40$$

98	98	10
80	80	66
40		

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_8 = \text{Max} \begin{bmatrix} 13 & 7 & 0 \\ 3 & 0 & 0 \\ 12 & 0 & 0 \end{bmatrix} = 13$$

98	98	10
80	80	66
40	13	

35	17	98	0	0	0
65	31	11	2	0	0
57	28	80	10	0	0
40	20	13	7	0	66
15	8	3	0	0	0
13	21	12	0	0	0

$$PR_9 = \text{Max} \begin{bmatrix} 7 & 0 & 66 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = 66$$

98	98	10
80	80	66
40	13	66

Perhitungan *Max\_Pooling channel Green (PG)* :

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_1 = \text{Max} \begin{bmatrix} 137 & 17 & 2 \\ 45 & 23 & 12 \\ 48 & 20 & 15 \end{bmatrix} = 137$$

137		

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_2 = \text{Max} \begin{bmatrix} 2 & 0 & 0 \\ 12 & 6 & 0 \\ 15 & 18 & 0 \end{bmatrix} = 18$$

137	18	

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_3 = \text{Max} \begin{bmatrix} 0 & 0 & 0 \\ 6 & 0 & 0 \\ 18 & 0 & 0 \end{bmatrix} = 18$$

137	18	18

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_4 = \text{Max} \begin{bmatrix} 48 & 20 & 15 \\ 21 & 5 & 8 \\ 0 & 0 & 3 \end{bmatrix} = 48$$

137	18	18
48		

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_5 = \text{Max} \begin{bmatrix} 15 & 18 & 0 \\ 8 & 11 & 0 \\ 3 & 0 & 0 \end{bmatrix} = 18$$

137	18	18
48	18	

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_6 = \text{Max} \begin{bmatrix} 18 & 0 & 0 \\ 11 & 0 & 0 \\ 0 & 0 & 11 \end{bmatrix} = 18$$

137	18	18
48	18	18

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_7 = \text{Max} \begin{bmatrix} 21 & 5 & 8 \\ 0 & 0 & 3 \\ 0 & 0 & 0 \end{bmatrix} = 21$$

137	18	18
48	18	18
21		

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_8 = \text{Max} \begin{bmatrix} 8 & 11 & 0 \\ 3 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = 11$$

137	18	18
48	18	18
21	11	

137	17	2	0	0	0
45	23	12	6	0	0
48	20	15	18	0	0
21	5	8	11	0	0
0	0	3	0	0	11
0	0	0	0	0	0

$$PG_9 = \text{Max} \begin{bmatrix} 11 & 0 & 0 \\ 0 & 0 & 11 \\ 0 & 0 & 0 \end{bmatrix} = 11$$

137	18	18
48	18	18
21	11	11

Perhitungan *Max\_Pooling channel Blue* (PB) :

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_1 = \text{Max} \begin{bmatrix} 131 & 21 & 5 \\ 49 & 67 & 12 \\ 56 & 27 & 14 \end{bmatrix} = 131$$

131		

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_2 = \text{Max} \begin{bmatrix} 5 & 0 & 0 \\ 12 & 10 & 0 \\ 14 & 16 & 0 \end{bmatrix} = 16$$

131	16	

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_3 = \text{Max} \begin{bmatrix} 0 & 0 & 0 \\ 10 & 0 & 0 \\ 16 & 0 & 0 \end{bmatrix} = 16$$

131	16	16

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_4 = \text{Max} \begin{bmatrix} 56 & 27 & 14 \\ 31 & 14 & 17 \\ 7 & 4 & 17 \end{bmatrix} = 56$$

131	16	16
56		

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_5 = \text{Max} \begin{bmatrix} 14 & 16 & 0 \\ 17 & 9 & 0 \\ 17 & 0 & 0 \end{bmatrix} = 17$$

131	16	16
56	17	

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_6 = \text{Max} \begin{bmatrix} 16 & 0 & 0 \\ 9 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = 16$$

131	16	16
56	17	16

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_7 = \text{Max} \begin{bmatrix} 31 & 14 & 17 \\ 7 & 4 & 17 \\ 2 & 12 & 7 \end{bmatrix} = 31$$

131	16	16
56	17	16
31		

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_8 = \text{Max} \begin{bmatrix} 17 & 9 & 0 \\ 17 & 0 & 0 \\ 7 & 0 & 0 \end{bmatrix} = 17$$

131	16	16
56	17	16
31	17	

131	21	5	0	0	0
49	67	12	10	0	0
56	27	14	16	0	0
31	14	17	9	0	0
7	4	17	0	0	0
2	12	7	0	0	3

$$PB_9 = \text{Max} \begin{bmatrix} 9 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 3 \end{bmatrix} = 9$$

131	16	16
56	17	16
31	17	9

Untuk melanjutkan ketahap berikutnya, yaitu transliterasi (*classification*). Hasil *pooling* disetiap *channel* dijumlahkan dan diterapkan proses *flatten* terlebih dahulu, sebagai berikut :

<i>Channel Red</i>				<i>Channel Green</i>				<i>Channel Blue</i>		
98	98	10		137	18	18		131	16	16
80	80	60	+	48	18	18		56	17	16
40	13	66		21	11	11		31	17	9

Proses dan hasil penjumlahan, sebagai berikut :

$$M_1 = 98 + 137 + 131 = 1248$$

$$M_2 = 98 + 18 + 18 = 122$$

$$M_3 = 10 + 18 + 16 = 44$$

$$M_4 = 80 + 48 + 56 = 184$$

$$M_5 = 80 + 68 + 17 = 115$$

$$M_6 = 60 + 18 + 16 = 94$$

$$M_7 = 40 + 21 + 31 = 92$$

$$M_8 = 13 + 11 + 17 = 41$$

$$M_9 = 66 + 11 + 9 = 86$$

1248	122	44
184	115	94
92	41	86

Kemudian hasil penjumlahan diterapkan proses *flatten*, sebagai berikut :

1248	122	44	→	1248
184	115	94		122
92	41	86		44
				184
				115
				94
				92
				41
				86

### 3.3.3.2. Tahapan transliterasi (*Classification*)

Pada tahapan kedua, transliterasi (*Classification*), proses diawali dengan proses *flattening*, kemudian menentukan banyaknya proses klasifikasi, banyaknya *neural (node)*, fungsi aktivasi dan nilai *dropout*. Arsitektur *neural network* diperlihatkan oleh Gambar 6 pada Bab 2 sebelumnya. Adapun penjelasan proses transliterasi (*classification*) lebih lanjut diperlihatkan oleh Tabel 10 di bawah ini :

Tabel 10 *Hyperparameter* tahapan transliterasi (*classification*)

No.	Hyperparameter	Keterangan proses
1	Klasifikasi	Dilakukan sebanyak 3 kali
2	Banyaknya <i>neural (node)</i>	Neural pertama = Jumlah total <i>node flatten</i> Neural kedua = $\frac{\text{Jumlah total node flatten}}{\text{banyaknya kelas}}$ Neural Ketiga = Banyaknya kelas
3	Fungsi <i>Dropout</i>	Dilakukan sebesar 0.01
4	Fungsi Aktivasi	Dilakukan dengan <i>softmax</i>

Berdasarkan keterangan *hyperparameter* proses yang diperlihatkan oleh Tabel 10 di atas. Merujuk kepada hasil *flattening* dari proses sebelumnya. Tahapan transliterasi (*classification*) aksara lampung dapat dihitung menggunakan persamaan (2.8) untuk *neural (node)* X, (2.9) untuk *neural (node)* Y, dan (2.10) untuk *neural (node)* Z, serta nilai bobot yang direpresentasikan sebagai berikut :

$$\begin{array}{lll}
 \text{bobot node x:} & \text{bobot node y :} & \text{bobot node z:} \\
 w_{x_1} = 0.2 & w_{y_1} = 0.3 & w_{z_1} = 0.04 \\
 w_{x_2} = 0.4 & w_{y_2} = 0.4 & w_{z_2} = 0.01 \\
 w_{x_3} = 0.1 & w_{y_3} = 0.1 & w_{z_3} = 0.05 \\
 w_{x_4} = 0.5 & w_{y_4} = 0.2 & w_{z_4} = 0.02 \\
 w_{x_5} = 0.3 & w_{y_5} = 0.6 & \\
 w_{x_6} = 0.6 & w_{y_6} = 0.5 & 
 \end{array}$$

Perhitungan *neural (node) X* :

$$x_1 = (1248 \times 0.2) + (122 \times 0.2) + (44 \times 0.2) + (184 \times 0.2) + (115 \times 0.2) + (94 \times 0.2) + (92 \times 0.2) + (41 \times 0.2) + (86 \times 0.2) \\ = 402$$

$$x_2 = (1248 \times 0.4) + (122 \times 0.4) + (44 \times 0.4) + (184 \times 0.4) + (115 \times 0.4) + (94 \times 0.4) + (92 \times 0.4) + (41 \times 0.4) + (86 \times 0.4) \\ = 804$$

$$x_3 = (1248 \times 0.1) + (122 \times 0.1) + (44 \times 0.1) + (184 \times 0.1) + (115 \times 0.1) + (94 \times 0.1) + (92 \times 0.1) + (41 \times 0.1) + (86 \times 0.1) \\ = 201$$

$$x_4 = (1248 \times 0.5) + (122 \times 0.5) + (44 \times 0.5) + (184 \times 0.5) + (115 \times 0.5) + (94 \times 0.5) + (92 \times 0.5) + (41 \times 0.5) + (86 \times 0.5) \\ = 1013$$

$$x_5 = (1248 \times 0.3) + (122 \times 0.3) + (44 \times 0.3) + (184 \times 0.3) + (115 \times 0.3) + (94 \times 0.3) + (92 \times 0.3) + (41 \times 0.3) + (86 \times 0.3) \\ = 607.8$$

$$x_6 = (1248 \times 0.6) + (122 \times 0.6) + (44 \times 0.6) + (184 \times 0.6) + (115 \times 0.6) + (94 \times 0.6) + (92 \times 0.6) + (41 \times 0.6) + (86 \times 0.6) \\ = 1215.6$$

Perhitungan *neural (node) Y* :

$$y_1 = (402 \times 0.3) + (804 \times 0.3) + (201 \times 0.3) + (1013 \times 0.3) + (607.8 \times 0.3) + (1215.6 \times 0.3) \\ = 1273.02$$

$$y_2 = (402 \times 0.4) + (804 \times 0.4) + (201 \times 0.4) + (1013 \times 0.4) + (607.8 \times 0.4) + (1215.6 \times 0.4) \\ = 1697.36$$

$$y_3 = (402 \times 0.1) + (804 \times 0.1) + (201 \times 0.1) + (1013 \times 0.1) + (607.8 \times 0.1) + (1215.6 \times 0.1) \\ = 424.34$$

$$y_4 = (402 \times 0.2) + (804 \times 0.2) + (201 \times 0.2) + (1013 \times 0.2) + (607.8 \times 0.2) + (1215.6 \times 0.2) \\ = 848.68$$

$$y_5 = (402 \times 0.6) + (804 \times 0.6) + (201 \times 0.6) + (1013 \times 0.6) + (607.8 \times 0.6) + (1215.6 \times 0.6) \\ = 2546.04$$

$$y_6 = (402 \times 0.5) + (804 \times 0.5) + (201 \times 0.5) + (1013 \times 0.5) + (607.8 \times 0.5) + (1215.6 \times 0.5) \\ = 2121.7$$

Perhitungan *neural(node) Z* :

$$z_1 = (1273.02 \times 0.04) + (1697.36 \times 0.04) + (424.34 \times 0.04) + (848.68 \times 0.04) + (2546.04 \times 0.04) + (2121.7 \times 0.04) \\ = 356.456$$

$$z_2 = (1273.02 \times 0.01) + (1697.36 \times 0.01) + (424.34 \times 0.01) + (848.68 \times 0.01) + (2546.04 \times 0.01) + (2121.7 \times 0.01) \\ = 89.114$$

$$z_3 = (1273.02 \times 0.05) + (1697.36 \times 0.05) + (424.34 \times 0.05) + (848.68 \times 0.05) + (2546.04 \times 0.05) + (2121.7 \times 0.05) \\ = 389.157$$

$$z_4 = (1273.02 \times 0.02) + (1697.36 \times 0.02) + (424.34 \times 0.02) + (848.68 \times 0.02) + (2546.04 \times 0.02) + (2121.7 \times 0.02) \\ = 178.228$$

Kemudian, setiap hasil perhitungan *neural Z* di atas diberikan fungsi aktivasi *softmax*, diperlihatkan oleh persamaan (2.11) . Adapun perhitungan adalah sebagai berikut :

$$Z_1 = \frac{e^{356.456}}{e^{356.456} + e^{89.114} + e^{389.57} + e^{178.228}} = 0.42$$

$$Z_2 = \frac{e^{89.114}}{e^{356.456} + e^{89.114} + e^{389.57} + e^{178.228}} = 0.33$$

$$Z_3 = \frac{e^{389.57}}{e^{356.456} + e^{89.114} + e^{389.57} + e^{178.228}} = 0.09$$

$$Z_4 = \frac{e^{178.228}}{e^{356.456} + e^{89.114} + e^{389.57} + e^{178.228}} = 0.16$$

Terlihat pada perhitungan di atas bahwa *neural Z<sub>1</sub>* memiliki hasil tertinggi, yaitu 0.43. Maka hasil yang didapat adalah kelas 1, yaitu ‘Abadi’.

#### 3.3.4. Melatih Sistem

Melatih sistem merupakan tahapan dalam mempersiapkan dan menentukan keberjalanan program. Tahapan ini dilakukan dengan cara menentukan jenis fungsi *loss* dan fungsi *optimizer* yang akan digunakan. Adapun fungsi *loss* dan fungsi *optimizer* yang digunakan dalam tugas akhir ini adalah *crossentropy* dan *RMSProp (Root Mean Squared Propagation)*.

### 3.3.5. Pengujian Sistem

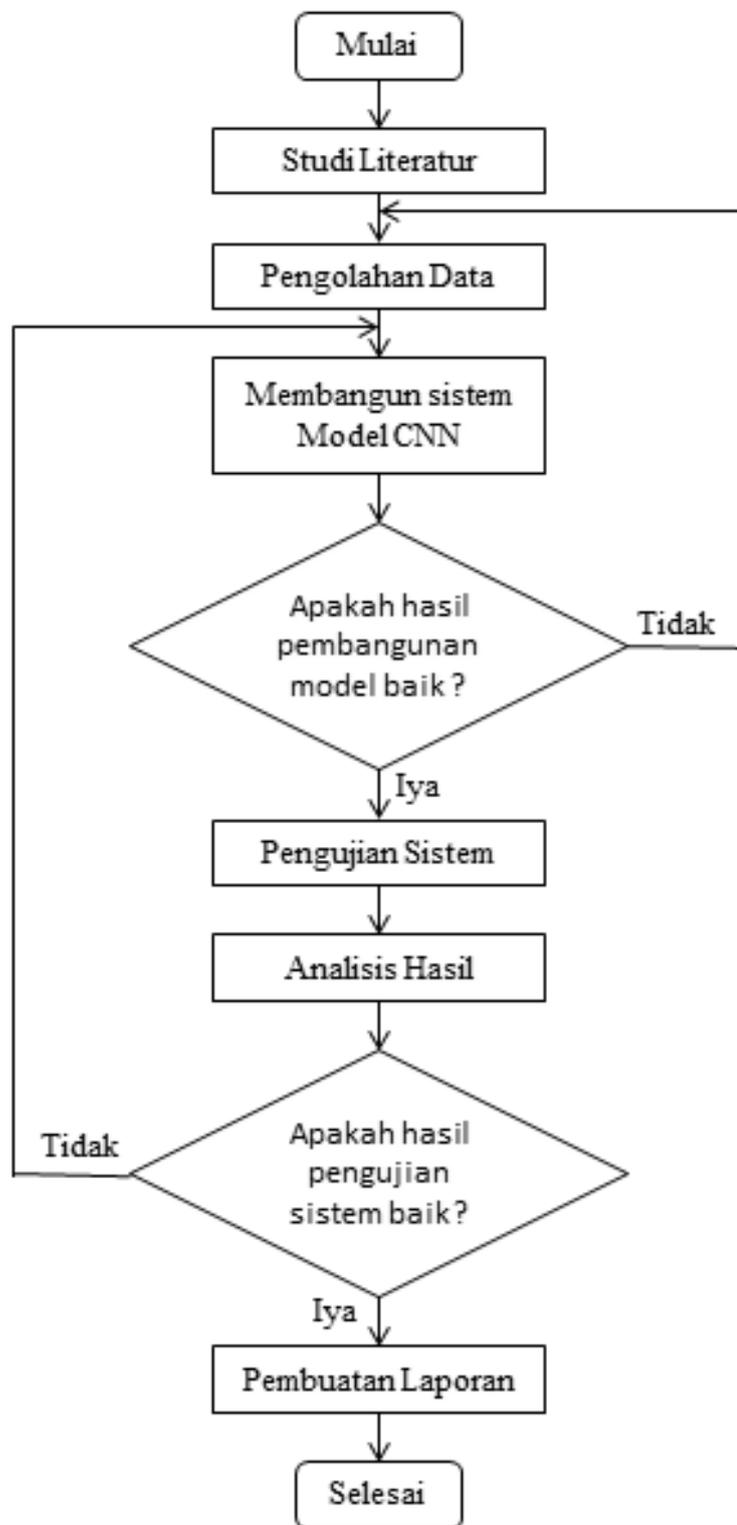
Pengujian aplikasi ini menggunakan *software visual studio code*, bahasa pemrograman *python 3.9.4. 64 bit*. Proses yang menjadi fokus dalam pengujian adalah seberapa akurat dan tepat sistem dalam melakukan pengenalan dan pengujian. Pada tahapan ini proses yang akan dilakukan adalah mengambil data citra uji melalui direktori dari laptop. Setelah citra diambil, Sistem akan mendeteksi dan menampilkan hasil transliterasi dari citra tersebut.

### 3.3.6 Analisis dan Pembahasan

Analisis dan pembahasan merupakan tahapan evaluasi terhadap keberhasilan pembuatan sistem model. Terdapat dua hal yang menjadi fokus dalam Analisis dan pembahasan, yaitu pertama hasil pengenalan dan transliterasi aksara lampung dan kedua keberjalanan program (arsitektur model CNN dan akurasi program).

## 3.4 Diagram Alir Penelitian

Diagram alir penelitian merupakan gambaran terhadap tahapan-tahapan yang akan dilakukan dalam menjalankan penelitian. Tahapan Penelitian ditunjukkan pada Gambar 24 di bawah ini.



Gambar 24 Diagram alir penelitian

Terlihat pada Gambar 24 di atas bahwa tahapan pada penelitian ini dimulai dengan melakukan studi literatur yang kemudian dilanjutkan dengan pengolahan data citra mentah aksara Lampung (RGB) melalui referensi yang telah diperoleh. Setelah berhasil proses selanjutnya adalah data tersebut digunakan dalam proses pembuatan sistem menggunakan *software visual studio code*, bahasa pemrograman *python 3.9.4. 64 bit*.

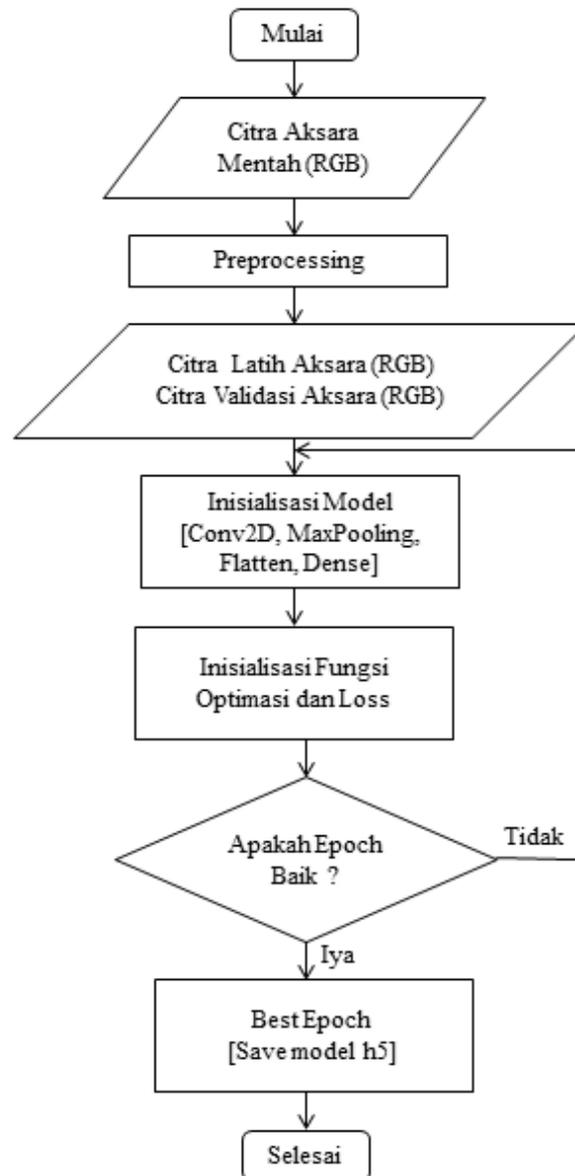
Tahap selanjutnya adalah proses membangun sistem (pelatihan dan validasi) model CNN, dengan maksud memastikan keberhasilan sistem. Apabila proses dapat berjalan sesuai dengan tujuan yang diinginkan, yaitu memperoleh *epoch* terbaik maka tahap selanjutnya adalah melakukan proses pengujian model. Namun, apabila sistem belum berhasil berjalan dengan baik, maka proses diulang kembali ke tahapan pengolahan data dan seterusnya sampai sistem berhasil mendapatkan nilai *epoch* terbaik.

Setelah sistem berhasil mendapatkan nilai *epoch* terbaik, tahapan selanjutnya adalah melakukan pengujian sistem. Kemudian, melakukan analisis terhadap hasil yang diperoleh. Apabila setelah proses pengujian ternyata program mengalami *error* atau hasil yang salah/kurang baik, maka proses akan diulang kembali ke tahapan pembangunan model sampai sistem menghasilkan hasil transliterasi yang baik dan benar. Tahapan terakhir setelah pengujian selesai adalah data terbaik akan digunakan dalam pembuatan laporan akhir dan penelitian selesai.

### **3.5 Diagram Alir Sistem Perancangan**

Diagram Alir Sistem Perancangan merupakan gambaran tahapan-tahapan yang akan dijalankan sistem dalam menjalankan fungsi untuk mencapai tujuan berupa

pengenalan dan transliterasi aksara lampung. Adapun diagram alir sistem perancangan terbagi menjadi dua proses, yaitu proses pembangunan (pelatihan dan validasi) sistem dan proses pengujian sistem yang diperlihatkan oleh Gambar 25 dan Gambar 26 di bawah ini.

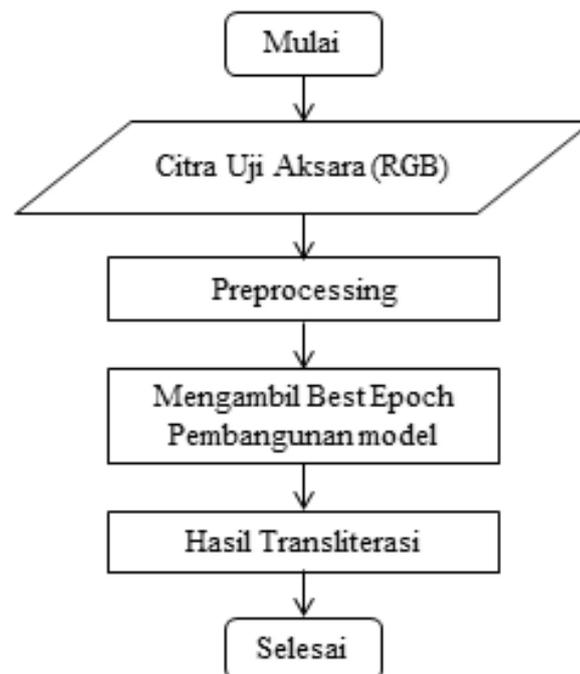


Gambar 25 Diagram Alir Pembangunan (pelatihan dan validasi) Sistem

Terlihat pada Gambar 25 Diagram Alir Pembangunan Sistem di atas, bahwa proses pembangunan sistem dimulai dengan membuat *dataset input* citra aksara mentah (RGB), kemudian data tersebut dimasukkan ke dalam sistem untuk dimodifikasi dan dipisah menjadi dua kategori, yaitu data latih dan data validasi didalam proses *pre-processing data*. Setelah data *input* citra aksara dimodifikasi dan dipisah, proses akan mengambil data-data tersebut untuk digunakan ke dalam model pembelajaran CNN.

Sebelum diproses, terlebih dahulu model pembelajaran CNN harus dibangun dengan cara menentukan bagian-bagian dalam *conv2D*, *Pooling2D*, *Flatten* dan *Dense*. Setelah itu, *input* data citra latih dan validasi dimasukkan kedalam proses yang telah ditentukan fungsi *optimizer* dan *loss* sebelumnya.

Setelah proses pembangunan sistem dilakukan, proses selanjutnya adalah evaluasi. Apakah *epoch* sudah baik ?, jika belum tahapan akan diulang ke proses membangun model pembelajaran CNN sampai *epoch* yang dihasilkan sudah baik dalam melakukan pelatihan. Jika, *epoch* yang dihasilkan sudah baik, nilai *epoch* disimpan menjadi '*best epoch*' dengan format '*model.h5*' dan proses pembangunan sistem selesai.



Gambar 26 Diagram Alir Pengujian Sistem

Terlihat pada Gambar 26 Diagram Alir Pengujian Sistem di atas bahwa pada proses pengujian secara umum sama seperti pada proses pembangunan sistem sebelumnya. Hanya saja, perbedaan terletak pada jenis data input yang digunakan, yaitu data uji. Dan *epoch* yang digunakan. *Epoch* yang digunakan berupa nilai *epoch* terbaik dari proses yang telah disimpan dari proses pembangunan sistem sebelumnya.

## **BAB V SIMPULAN DAN SARAN**

### **5.1. Simpulan**

Dari hasil penelitian yang telah dilakukan diperoleh simpulan bahwa

1. Telah terealisasi sistem pengenalan dan transliterasi aksara Lampung berbasis penerapan CNN dengan tingkat akurasi pengujian sebesar 80 %
2. Sistem pengenalan dan transliterasi aksara Lampung dengan model pembelajaran CNN menggunakan 1732 data citra yang terbagi kedalam 4 kelas (*output*) dengan masing-masing 423 citra untuk membangun sistem dan masing-masing 10 citra untuk menguji sistem pada masing-masing katanya, dengan fungsi optimasi *Root Mean Square Propagation (RMSProp)*, jenis *pooling* berupa *max pooling*, *dropout* 0.01, *features extraction* (pengenalan) sebanyak 4 layer dan *classification* (transliterasi) sebanyak 3 layer.

### **5.2. Saran**

Berikut ini beberapa hal yang disarankan terkait penelitian ini :

1. Melakukan Akuisisi data dengan variasi bentuk tulisan tangan yang berbeda dapat membuat sistem yang akan dibangun selanjutnya belajar

lebih banyak jenis data, sehingga lebih baik dalam memprediksi data (aksara Lampung) terutama dalam segi bentuk tulisan.

2. Melakukan pembagian data mentah menjadi data latih dan data validasi dengan persentase yang tidak terlalu jauh dapat dijadikan pertimbangan untuk memperbaiki dan merancang kembali model pembelajaran.
3. Mengaplikasikan metode validasi silang dalam proses pembangunan dapat membantu model memvalidasi dengan lebih terarah, sehingga dapat mengatasi hasil distribusi keberhasilan pengujian data yang tidak merata.
4. Merancang model dengan kategori kata dengan jumlah huruf yang sama dapat meminimalisir kesalahan dalam memprediksi atau mentransliterasi data citra aksara Lampung

## DAFTAR PUSTAKA

- [1] Setiawan, W. 2017. *Era Digital dan Tantangannya*. Universitas Pendidikan Indonesia, Bandung, Indonesia.
- [2] Sazqiah, N. dkk, 2022. *Pengenalan Aksara Lampung Menggunakan Metode CNN (Convolution Neural Network)*. Prosiding Seminar Nasional Insinyur Profesional (SNIP) Vol 2, No.1. ISSN 2828-5794. Universitas Lampung, Lampung, Indonesia.
- [3] Silalahi, D. dkk, 2020. *Implementasi Convolution Neural Network Untuk Klasifikasi Kata Pada Citra Teks*. Seminar Nasional Informatika, Sistem Informasi dan Keamanan Siber (SEINASI-KESI), Jakarta, Indonesia.
- [4] Mulyanto, A. dkk, 2021. *Penerapan Convolutional Neural Network (CNN) pada Pengenalan Aksara Lampung berbasis Optical Character Recognition (OCR)*. Vol. 7, No.1. jepin. Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia.
- [5] Pudjiastuti, T. 1996. *Aksara dan Naskah Kuno Lampung dalam Pandangan Masyarakat Lampung Kini*. Edisi 1996. Proyek Pengkajian dan Pembinaan Nilai-nilai Budaya Direktorat Sejarah dan Nilai Tradisional Direktorat Jenderal Kebudayaan Departemen Pendidikan dan Kebudayaan, Jakarta, Indonesia.
- [6] Indrayati, R.I. dan Migotuwio, N. 2020. *Identifikasi Anatomi Aksara Lampung*. AKSA : Jurnal Desain Komunikasi Visual Vol. 4, No 1. ISSN : 2615 – 1111. Institut Teknologi Sumatera, Lampung, Indonesia
- [7] Fadjeri, A. dkk. 2020. *Pengolahan Citra Digital Untuk Menghitung Ekstraksi Ciri Greenbeen Kopi Robusta dan Arabika [Studi kasus ; Kopi Temanggung]*. Vol 4 No 2. Indonesia Journal of Applied Informatics (IJAI). Universitas Amikom Yogyakarta, Indonesia.

- [8] Kadir, A. dan Susanto, A. 2012. *Teori dan Aplikasi Pengolahan Citra*. Penerbit Andi, Yogyakarta, Indonesia.
- [9] Paulus, E. dkk. 2019. *Analisa dan Pengolahan Citra Digital, Study Kasus Tulisan Tangan Sunda Kuno*. Betread Publishing, Indonesia.
- [10] Munantri, N.Z. dkk. 2019. *Aplikasi Pengolahan Citra Digital Untuk Identifikasi Umur Pohon*. Telematika. Vol. 16 No.2, Pp 97-104. Yogyakarta, Indonesia.
- [11] Hidayat, J. dkk. 2019. *Perbandingan Metode Perbaikan Kualitas Citra Histogram Equalization pada Citra Satelit*. Journal of Electical Technology. Vol 4, No 3, ISSN 2598 – 1099. Politeknik Penerbangan Medan, Medan, Indonesia.
- [12] Sharma, N. dkk.2018. *An Analysis of Convolutional Neural Networks For Image Classification*, Procedia Comput. Sci., vol. 132, pp. 377–384.
- [13] Shanmugamani, R. 2018. *Deep Learning for Computer Vision*. Birmingham: Packt Publishing.
- [14] Gunawan, I. 2020. *Optimasi Model Artificial Neural Network untuk Klasifikasi Paket Jaringan*. SIMETRIS Vol 14, No 2, Desember 2020. E-ISSN 2686-312x. Sekolah Tinggi Teknologi Ronggolawe Cepu. Indonesia.
- [15] Iqbal, MI. 2021. *Deteksi Kerusakan Ban Menggunakan Algoritma Convolution Neural Network*. Skripsi. Fakultas Komputer, Universitas Pembangunan Nasional “Veteran” Jawa Timur, Indonesia.