

**PERBANDINGAN KINERJA ARSITEKTUR *LENET* DAN *ALEXNET*  
UNTUK KLASIFIKASI CITRA AKSARA LAMPUNG**

**(Skripsi)**

Oleh  
**OKTA TOYIBAH**  
1917051023



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2023**

## ABSTRAK

### PERBANDINGAN KINERJA ARSITEKTUR *LENET* DAN *ALEXNET* UNTUK KLASIFIKASI CITRA AKSARA LAMPUNG

Oleh

**OKTA TOYIBAH**

Aksara atau Had Lampung merupakan salah satu unsur kekayaan Indonesia yang berkembang di selatan Pulau Sumatera, dimana penggunaannya semakin lama kian berkurang karena berbagai faktor seperti globalisasi dan arus transmigrasi. Salah satu upaya peneliti dalam pelestarian Aksara Lampung yaitu, menjadikan Aksara Lampung sebagai objek penelitian menggunakan pendekatan *Deep Learning* yaitu CNN (*Convolutional Neural Network*) dengan membandingkan arsitektur *LeNet* dan *AlexNet* untuk klasifikasi citra Aksara Lampung. Sehingga, hasil dari penelitian ini dapat menjadi dasar dalam pengembangan suatu aplikasi yang terintegrasi dengan klasifikasi Aksara Lampung seperti *machine translation system*, aplikasi kamus bahasa lampung, *game* edukasi dan lainnya, yang memiliki potensi untuk mempertahankan dan memperkuat warisan budaya Lampung. Penelitian ini terbagi menjadi 4 skema menggunakan teknik augmentasi dimana, skema 1 data tidak dilakukan perlakuan, skema 2 dilakukan augmentasi dengan operasi *random rotation*, skema 3 dilakukan augmentasi dengan operasi *random shear* dan skema 4 dilakukan penggabungan antar kedua operasi *random rotation* dan *random shear*. Dalam penelitian ini menggunakan data sebanyak 32.140 karakter citra dengan 18 kelas. Hasil terbaik yang diperoleh pada penelitian ini, yaitu nilai akurasi sebesar 98,02% untuk arsitektur *LeNet* pada skema 4, sedangkan dengan arsitektur *AlexNet* memperoleh tingkat akurasi tertinggi sebesar 91,04% pada skema 4.

Kata kunci: Aksara Lampung, *LeNet*, *AlexNet*.

## **ABSTRACT**

### **PERFORMANCE COMPARISON OF LENET AND ALEXNET ARCHITECTURES FOR CLASSIFICATION OF LAMPUNG SCRIPT IMAGES**

**By**

**OKTA TOYIBAH**

Lampung script or Had Lampung is one of the elements of Indonesian wealth that developed in the south of Sumatra Island, where its use is decreasing due to various factors such as globalization and transmigration. One of the researcher's efforts in preserving Lampung Script is to make Lampung Script an object of research using a Deep Learning approach, namely CNN (Convolutional Neural Network) by comparing the LeNet and AlexNet architectures for the classification of Lampung Script images. Thus, the results of this research can be the basis for developing an application that is integrated with the classification of Lampung Script such as machine translation systems, Lampung language dictionary applications, educational games and others, which have the potential to maintain and strengthen Lampung's cultural heritage. This research is divided into 4 schemes using augmentation techniques where, scheme 1 data is not treated, scheme 2 is augmented with random rotation operations, scheme 3 is augmented with random shear operations and scheme 4 is combined between both random rotation and random shear operations. In this study using data as many as 32,140 image characters with 18 classes. The best results obtained in this study, namely the accuracy value of 98.02% for the LeNet architecture in scheme 4, while the AlexNet architecture obtained the highest accuracy rate of 91.04% in scheme 4.

**Keywords:** Lampung script, LeNet, AlexNet.

**PERBANDINGAN KINERJA ARSITEKTUR *LENET* DAN *ALEXNET*  
UNTUK KLASIFIKASI CITRA AKSARA LAMPUNG**

**Oleh**

**OKTA TOYIBAH**

**Skripsi**

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

**Pada**

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG**

**2023**

Judul Skripsi : **PERBANDINGAN KINERJA  
ARSITEKTUR *LENET* DAN *ALEXNET*  
UNTUK KLASIFIKASI CITRA AKSARA**

Nama Mahasiswa : **Okta Toyibah**

Nomor Pokok Mahasiswa : 1917051023

Program Studi : *S1 Ilmu Komputer*

Jurusan : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam



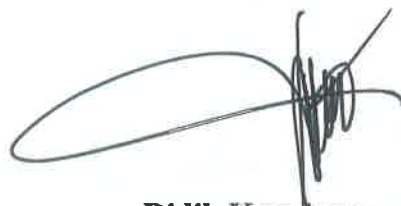
**MENYETUJUI**

1. Komisi Pembimbing



**Dr. rer. nat. Akmal Junaidi, M.Sc.**  
NIP. 19710129 199702 1 001

2. Ketua Jurusan Ilmu Komputer



**Didik Kurniawan, S.Si., M.T.**  
NIP. 19800419 200501 1 004

**MENGESAHKAN**

1. Tim Penguji

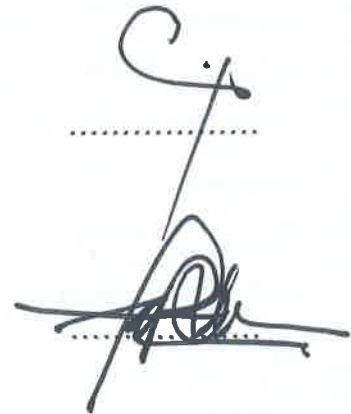
Ketua : **Dr. rer. nat. Akmal Junaidi, M.Sc.**



Sekretaris : **Rico Andrian, S.Si., M.Kom.**



Penguji Bukan Pembimbing : **Favorisen R. Lumbanraja, Ph.D.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



**Dr. Eng. Heri Satria, S.Si., M.Si.**  
NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: **28 Juli 2023**

## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Okta Toyibah

NPM : 1917051023

Dengan ini menyatakan bahwa skripsi saya yang berjudul “**Perbandingan Kinerja Arsitektur LeNet Dan AlexNet Untuk Klasifikasi Citra Aksara Lampung**” merupakan karya saya sendiri, bukan karya orang lain. Semua tulisan yang tertulis dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Jika di kemudian hari terbukti bahwa karya tulis ilmiah saya terbukti hasil menjiplak karya orang lain, maka saya siap menerima sanksi berupa pencabutan gelar yang saya peroleh.

Bandar Lampung, Juli 2023



Okta Toyibah

NPM. 1917051023

## RIWAYAT HIDUP



Penulis lahir di Prabumulih, pada 23 Oktober 2001, sebagai anak ketiga dari empat bersaudara. Penulis menyelesaikan pendidikan formal di SD Palm Kids Prabumulih dan selesai pada tahun 2013. Kemudian pendidikan menengah pertama di SMPN 1 Prabumulih yang diselesaikan pada tahun 2016, lalu melanjutkan ke pendidikan menengah atas di SMAN 3 Prabumulih yang diselesaikan pada tahun 2019.

Pada tahun 2019 penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur SNMPTN. Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan antara lain.

1. Menjadi anggota bidang Kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2019/2020.
2. Menjadi Staff Ahli bidang Komunikasi dan Informasi Badan Eksekutif Mahasiswa (BEM) Universitas Lampung pada periode 2020/2021.
3. Menjadi Sekretaris Koordinator Divisi HPDD Program Orientasi Mahasiswa Jurusan Ilmu Komputer 2020.
4. Menjadi Panitia Bidang dalam kegiatan Gebyar Adapter 2019 dan Pesta KBM 2020 bidang Acara.
5. Menjadi Sekretaris Koordinator PDD Papermob Universitas Lampung 2020.
6. Menjadi Bendahara Pelaksana LKMMIKTD 2020.
7. Mengikuti Program Kampus Merdeka KMMI *Android Apps with Flutter* 2021.
8. Mengikuti Program MBKM Jurusan di Nusantara Regas Jakarta bidang Divisi Keuangan dan Sistem Informasi dari Juli – Desember 2022.



9. Mengikuti KKN Periode 1 tahun 2022 di Desa Durian, Peninjauan OKU dan KWI di Lampung Timur tahun 2019.
10. Asisten Dosen pada matakuliah Logika, Matematika Diskrit dan Basis Data 2020/2021.

## MOTTO

*“...And He found you lost and guided you...”*

*- Ad Dhuha: 7 -*

*“Anyone from anywhere can (be/do) anything”*

*- Author -*

*“Pembagian data , 80% data latih, 10% data validasi dan 10% data uji.  
Pembagian hidup, 80% uji-an hidup, 10% latih-an hidup dan 10% evaluasi”*

*- Author -*

## **PERSEMBAHAN**

### *Alhamdulillahirobbilamin*

Puji dan syukur tercurahkan kepada Allah Subhanahu Wa Ta'ala atas segala Rahmat dan Karunia-Nya sehingga saya dapat menyelesaikan skripsi ini. Shalawat serta salam selalu tercurahkan kepada Nabi Muhammad SAW.

Kupersembahkan karya ini kepada:

### **Kedua Orang Tuaku dan Keluarga Tercinta**

Yang senantiasa memberikan yang terbaik, dukungan dan doa yang selalu menyertaiku. Terima kasih banyak atas semangat, motivasi, dan apresiasi yang selalu mengiringi. Dan terima kasih banyak telah tanpa lelah mendidik dan membesarkanku dengan penuh cinta, kesabaran dan pengorbanan.

### **Seluruh Keluarga Besar Ilmu Komputer 2019**

Yang selalu memberikan semangat dan dukungan.

### **Almamater Tercinta, Universitas Lampung dan Jurusan Ilmu Komputer**

Tempat belajar mengemban seluruh ilmu untuk menjadi bekal hidup.

## SANWACANA

Puji syukur kehadiran Allah SWT atas berkah, rahmat dan hidayat-Nya, serta petunjuk dan pedoman dari Rasulullah Nabi Muhammad SAW Penulis dapat menyelesaikan Skripsi yang berjudul “ Perbandingan Kinerja Arsitektur *LeNet* Dan *AlexNet* Untuk Klasifikasi Citra Aksara Lampung” dengan baik.

Selama proses penulisan skripsi ini tidak terlepas dari dukungan banyak pihak yang telah membimbing, membantu, dan memberi semangat kepada saya, sehingga pada kesempatan ini saya ingin menyampaikan ungkapan terima kasih kepada:

1. Allah Subhanahu Wa Ta'ala yang telah memberikan rahmat dan karunia-Nya selama menjalani penelitian hingga laporan dapat diselesaikan dengan baik.
2. Kedua Orang Tua tercinta, Ayuk, Adik dan Kakak yang selalu memberi dukungan, doa, semangat, dan kasih sayang yang tak terhingga. Semoga Allah SWT selalu memberikan kebahagiaan dan keberkahan dalam kehidupan kalian di dunia dan akhirat.
3. Bapak Dr. Eng. Heri Satria, S.Si., M.Si., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam.
4. Bapak Didik Kurniawan, S.Si., M.T., selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
5. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc. selaku Pembimbing Utama yang telah memberikan bimbingan, arahan, ide, ilmu, serta saran kepada Penulis sehingga dapat menyelesaikan skripsi ini dengan baik.
6. Bapak Rico Andrian, S.Si., M.Kom. selaku Pembahas I yang telah memberikan ilmu, masukan dan saran yang bermanfaat dalam penelitian ini.
7. Bapak Favorisen R. Lumbanraja, Ph.D. selaku Pembahas II yang telah memberikan ilmu, masukan dan saran yang bermanfaat dalam penelitian ini.

8. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah membantu penulis selama proses perkuliahan dan memberikan ilmu yang bermanfaat kepada penulis.
9. Seluruh Staf dan karyawan Fakultas MIPA Universitas Lampung: Ibu Ade Nora Maela, Bang Zainuddin, Mas Syam, Mas Ardi Novalia, dan lainnya, yang telah membantu segala urusan administrasi penulis.
10. Nur Rizky Mawadha, Aghita Namira Yuliza, Fanirizki Sofyana, Nur Ayu Octarina, Mantika Dewi Lestari dan teman-teman 875, yang telah memberikan banyak dukungan moril, menyemangati, mendengar keluh kesah dan menemani penulis.
11. Rekan-rekan Ilmu Komputer 2019, kakak dan adik tingkat yang tidak dapat disebut satu persatu.
12. Diriku sendiri. Terima kasih karena telah bertahan dan tidak menyerah dalam banyak hal. Kamu hebat.

Penulis menyadari bahwa Skripsi ini masih jauh dari kata sempurna, semoga Skripsi ini membawa manfaat dan keberkahan bagi semua yang membacanya Aamiin Ya Rabbal Aalamiin.

Bandar Lampung, Juli 2023

Okta Toyibah

NPM. 1917051023

## DAFTAR ISI

	Halaman
<b>DAFTAR GAMBAR</b> .....	<b>vii</b>
<b>DAFTAR TABEL</b> .....	<b>ix</b>
<b>DAFTAR KODE</b> .....	<b>xi</b>
<b>I. PENDAHULUAN</b> .	
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	5
<b>II. TINJAUAN PUSTAKA</b>	
2.1 Penelitian Terdahulu.....	6
2.2 Aksara Lampung .....	8
2.3 <i>Deep Learning</i> .....	11
2.4 <i>Convolutional Neural Network (CNN)</i> .....	12
2.4.1 <i>Convolutional Layer</i> .....	14
2.4.2 Fungsi Aktivasi .....	15
2.4.3 <i>Pooling Layer</i> .....	15
2.4.4 <i>Fully-Connected Layer</i> .....	16
2.5 <i>LeNet</i> .....	17
2.6 <i>AlexNet</i> .....	17
2.7 <i>Hyperparameter</i> .....	18
2.7.1 <i>Epoch</i> .....	19

2.7.2	<i>Optimizer</i> .....	19
2.7.3	<i>Learning Rate</i> .....	19
2.7.4	<i>Batch Size</i> .....	20
2.8	<i>Confusion Matrix</i> .....	20
2.8.1	<i>Accuracy</i> .....	21
2.8.2	<i>Recall</i> .....	21
2.8.3	<i>Precision</i> .....	22
2.8.4	<i>F1-Score</i> .....	22
2.9	<i>Multiclass Classification</i> .....	23
2.10	Teknik Augmentasi.....	23

### III. METODOLOGI PENELITIAN

3.1	Tempat dan Waktu Penelitian.....	25
3.2	Alat dan Data.....	26
3.2.1	Alat.....	26
3.2.2	Data.....	28
3.3	Tahapan Penelitian.....	32
3.3.1	<i>Dataset</i> .....	33
3.3.2	Separasi Data.....	33
3.3.3	Pra Pemrosesan Data.....	34
3.3.4	Pelatihan Model.....	35
3.3.5	Pengujian Model <i>LeNet</i> dan <i>AlexNet</i> .....	36
3.3.6	Evaluasi Hasil ( <i>Classification Report</i> ).....	36
3.3.7	Penarikan Kesimpulan.....	36

### IV. HASIL DAN PEMBAHASAN

4.1	Pengumpulan Dataset.....	37
4.2	Separasi Data.....	37
4.3	Pra pemrosesan Data.....	41
4.4	Proses <i>Training</i> Arsitektur <i>LeNet</i> dan <i>AlexNet</i> .....	43
4.5	Hasil <i>Testing</i> Arsitektur <i>LeNet</i> dan <i>AlexNet</i> .....	55
4.6	Penelitian Terdahulu Terkait Sumber <i>Dataset</i> .....	55

**V. SIMPULAN DAN SARAN**

5.1 Simpulan..... 86

5.2 Saran..... 87

**DAFTAR PUSTAKA**



## DAFTAR GAMBAR

Gambar	Halaman
1. Aksara Dasar (kelabai surat) Lampung.....	9
2. Diakritik (benah surat) pada Aksara Dasar. ....	10
3. Contoh Penggunaan Diakritik pada Aksara Lampung.....	10
4. Tanda Baca pada Aksara Lampung.....	11
5. Ilustrasi Arsitektur CNN. ....	13
6. Operasi Konvolusi.....	14
7. Fungsi Aktivasi ( <i>Sigmoid, Tanh, ReLU</i> ).....	15
8. <i>Pooling Layer</i> .....	16
9. Arsitektur <i>LeNet</i> . ....	17
10. Arsitektur <i>AlexNet</i> . ....	18
11. <i>Confusion Matrix</i> . ....	20
12. <i>Pie Chart</i> Proporsi Distribusi Kemunculan Karakter Aksara Dalam Korpus. ....	30
13. <i>Bar Chart</i> Frekuensi Kemunculan Karakter Aksara Dalam Korpus. ....	30
14. Alur Tahapan Penelitian Klasifikasi Menggunakan Arsitektur <i>LeNet</i> dan <i>AlexNet</i> . .....	32
15. <i>Bar Chart Data Training</i> . ....	39
16. <i>Bar Chart Data Validation</i> . ....	40
17. <i>Bar Chart Data Testing</i> .....	40
18. Contoh Aksara Setelah Dilakukan Teknik Augmentasi.....	42
19. Bar Chart Perbandingan Data Training Setelah dan Sebelum Dilakukan Teknik Augmentasi. ....	42
20. <i>Visualkeras</i> Model Arsitektur <i>LeNet</i> .....	44

21. <i>Visualkeras</i> Model Arsitektur <i>AlexNet</i> . .....	46
22. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>LeNet</i> pada Skema 1. ....	51
23. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>LeNet</i> pada Skema 2. ....	51
24. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>LeNet</i> pada Skema 3. ....	51
25. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>LeNet</i> pada Skema 4. ....	52
26. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>AlexNet</i> pada Skema 1. ....	52
27. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>AlexNet</i> pada Skema 2. ....	52
28. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>AlexNet</i> pada Skema 3. ....	53
29. Grafik Nilai <i>Accuracy</i> dan <i>Loss</i> dengan Arsitektur <i>AlexNet</i> pada Skema 4. ....	53
30. Aksara "a", "la", "ga", "sa", "na" dan "ja". .....	67
31. Perbandingan Tingkat Akurasi Pada Penelitian Junaidi, 2011. ....	85

## DAFTAR TABEL

Tabel	Halaman
1. Penelitian Terdahulu Terkait Klasifikasi Tulisan Tangan. ....	6
2. <i>Gantt Chart</i> Penelitian .....	25
3. Jumlah Data Per Kelas. ....	29
4. <i>Hyperparameter</i> . ....	35
5. Jumlah Data Per Kelas Setiap Subset Data. ....	38
6. <i>Summary</i> Struktur Hasil <i>Output</i> Arsitektur <i>LeNet</i> . ....	44
7. <i>Summary</i> Struktur Hasil <i>Output</i> Arsitektur <i>AlexNet</i> . ....	46
8. <i>Arguments ReduceLROnPlateau</i> . ....	48
9. Nilai <i>Accuracy</i> dan <i>Loss</i> Pada <i>Training</i> dan <i>Validation</i> Arsitektur <i>LeNet</i> . ....	49
10. Nilai <i>Accuracy</i> dan <i>Loss</i> Pada <i>Training</i> dan <i>Validation</i> Arsitektur <i>AlexNet</i> . ....	49
11. Nilai <i>Accuracy</i> dan <i>Loss</i> Pada <i>Data Testing</i> untuk Arsitektur <i>LeNet</i> . ....	55
12. Nilai <i>Accuracy</i> dan <i>Loss</i> pada <i>Data Testing</i> untuk Arsitektur <i>AlexNet</i> .....	56
13. Selisih Tingkat <i>Accuracy</i> pada <i>Data Training</i> dan <i>Testing</i> . ....	56
14. <i>Confusion Matrix</i> Arsitektur <i>AlexNet</i> pada Skema 1. ....	58
15. <i>Confusion Matrix</i> Arsitektur <i>AlexNet</i> pada Skema 2. ....	59
16. <i>Confusion Matrix</i> Arsitektur <i>AlexNet</i> pada Skema 3. ....	60
17. <i>Confusion Matrix</i> Arsitektur <i>AlexNet</i> pada Skema 4. ....	61
18. <i>Confusion Matrix</i> Arsitektur <i>LeNet</i> pada Skema 1. ....	62
19. <i>Confusion Matrix</i> Arsitektur <i>LeNet</i> pada Skema 2. ....	63
20. <i>Confusion Matrix</i> Arsitektur <i>LeNet</i> pada Skema 3. ....	64
21. <i>Confusion Matrix</i> Arsitektur <i>LeNet</i> pada Skema 4. ....	65
22. <i>False Prediction</i> Aksara "a" pada "la", Skema 1 dengan Arsitektur <i>AlexNet</i> . ....	67

23. <i>False Prediction</i> Aksara "a" pada "la", Skema 2 dengan Arsitektur <i>AlexNet</i> .....	69
24. <i>False Prediction</i> Aksara "a" pada "la", Skema 3 dengan Arsitektur <i>AlexNet</i> .....	71
25. <i>False Prediction</i> Aksara "a" pada "la", Skema 4 dengan Arsitektur <i>AlexNet</i> .....	72
26. <i>False Prediction</i> Aksara "a" pada "la", Skema 1 dengan Arsitektur <i>LeNet</i> .....	73
27. <i>False Prediction</i> Aksara "ka" pada "a", Skema 2 dengan Arsitektur <i>LeNet</i> .....	74
28. <i>False Prediction</i> Aksara "ka" pada "a", Skema 3 dengan Arsitektur <i>LeNet</i> .....	74
29. <i>False Prediction</i> Aksara "ga" pada "sa", Skema 4 dengan Arsitektur <i>LeNet</i> . ....	75
30. <i>Classification Report</i> Arsitektur <i>LeNet</i> pada Skema 1. ....	76
31. <i>Classification Report</i> Arsitektur <i>LeNet</i> pada Skema 2. ....	77
32. <i>Classification Report</i> Arsitektur <i>LeNet</i> pada Skema 3. ....	78
33. <i>Classification Report</i> Arsitektur <i>LeNet</i> pada Skema 4. ....	79
34. <i>Classification Report</i> Arsitektur <i>AlexNet</i> pada Skema 1. ....	80
35. <i>Classification Report</i> Arsitektur <i>AlexNet</i> pada Skema 2. ....	81
36. <i>Classification Report</i> Arsitektur <i>AlexNet</i> pada Skema 3. ....	82
37. <i>Classification Report</i> Arsitektur <i>AlexNet</i> pada Skema 4. ....	83

## DAFTAR KODE

Kode	Halaman
1. Potongan Kode untuk Separasi Data dengan Modul <i>Splitfolders</i> . .....	38

## I. PENDAHULUAN

### 1.1 Latar Belakang

Aksara merupakan alat komunikasi, baik lisan maupun tertulis. Sebagai peninggalan masyarakat zaman lampau sejak ribuan tahun lalu yang diwariskan secara turun-temurun dari generasi ke generasi, aksara tidak bisa diabaikan keberadaannya (Roza, 2017). Salah satu jenis aksara, yaitu aksara Nusantara yang digunakan di wilayah Nusantara. Sebagian besar aksara Nusantara masih diajarkan sebagai bagian dari muatan lokal di daerah masing-masing, tetapi dengan penerapan yang terbatas dalam kehidupan sehari-hari. Selain sebagai alat komunikasi dan pencatatan sejarah, aksara Nusantara juga berperan penting dalam kehidupan spiritual dan kepercayaan masyarakat Nusantara. Beberapa aksara tulisan daerah dinamai berdasarkan urutan penyusunan huruf-hurufnya atau menurut nama *abecedarium* aksara tersebut. Aksara Jawa modern dan Aksara Bali disebut Aksara *Hanacaraka*; sedangkan Aksara Rejang, Aksara Kerinci, Aksara Lampung, dan Aksara Sunda Baku disebut juga Aksara Kaganga mengikuti *abecedarium* Aksara Pallawa: *ka kha ga gha nga* (Dewi dan Muslihah, 2022).

Aksara Lampung atau Had Lampung adalah salah satu aksara tradisional Indonesia yang berkembang di selatan Pulau Sumatera. Aksara ini digunakan untuk menulis rumpun Bahasa Lampung dan Bahasa Melayu. Aksara Lampung merupakan turunan dari Aksara Brahmi melalui perantara Aksara Kawi. Terdapat dua atau tiga ragam Bahasa Lampung, yaitu: Lampung Api (juga disebut Pesisir atau dialek A), Lampung Nyo (juga disebut Abung atau dialek O), dan Komering. Ragam terakhir terkadang

dianggap sebagai bagian dari Lampung Api, tetapi terkadang juga dianggap sebagai bahasa yang berdiri sendiri terpisah dari Bahasa Lampung. Aksara Lampung tidak lagi digunakan untuk baca tulis secara fungsional. Fungsi Aksara Lampung secara *de facto* tergantikan oleh Aksara Latin. Pemerintah provinsi mencoba menghidupkan kembali penggunaan Aksara Lampung dengan menggelar musyawarah pembakuan Aksara Lampung pada tanggal 23 Februari 1985. Hasil musyawarah tersebut hingga hari ini masih menuai perdebatan dan ketidaksetujuan dari beberapa pihak. Berdasarkan peraturan pemerintah daerah Lampung nomor 2 tahun 2008 tentang pemeliharaan kebudayaan Lampung, Aksara Lampung menjadi unsur kekayaan yang wajib dilestarikan dan dikembangkan (Indrayati dan Migotuwio, 2020). Salah satu usaha dalam pelestariannya, Aksara Lampung dapat dijumpai pada gapura perbatasan, berbagai dekorasi, nama jalan bahkan terdapat Kongres Bahasa Lampung yang dibuka oleh Gubernur Provinsi Lampung. Salah satu pendekatan ilmiah sebagai bentuk pelestarian Aksara Lampung yang dilakukan oleh peneliti, yaitu menjadikan Aksara Lampung sebagai objek penelitian dengan memanfaatkan kemajuan teknologi, yaitu *deep learning*.

Penelitian ini dilakukan sebagai salah satu upaya dalam melestarikan budaya Lampung. Bahasa Lampung merupakan bahasa minoritas di Provinsi Lampung sendiri. Kekhawatiran akan keberlangsungan Bahasa Lampung telah membuat pemerintah daerah setempat mengimplementasikan kebijakan pengajaran bahasa dan Aksara Lampung bagi sekolah-sekolah pada tingkat dasar dan menengah di Provinsi tersebut. Dengan mengembangkan metode klasifikasi Aksara Lampung yang akurat dan efektif, penelitian ini dapat membantu mempertahankan dan melestarikan keberadaan Aksara Lampung sebagai warisan budaya Lampung yang penggunaannya semakin lama kian berkurang karena berbagai faktor seperti globalisasi dan arus transmigrasi.

Penelitian ini dapat menjadi dasar dalam pengembangan suatu aplikasi seperti *machine translation system*, aplikasi kamus bahasa Lampung, *game* edukasi dan lainnya. Dimana sistem tersebut terintegrasi dengan klasifikasi Aksara Lampung yang akurat. Sehingga, menjadi media yang bisa memberikan akses lebih mudah, menarik dan efektif yang diharapkan dapat memperluas penggunaan Aksara Lampung. Secara keseluruhan, pengembangan metode klasifikasi Aksara Lampung ini memiliki potensi besar untuk mempertahankan dan memperkuat warisan budaya Lampung serta mengintegrasikannya ke dalam era modern.

*Deep Learning* merupakan salah satu cabang dari *machine learning*. Model *Deep Learning* dapat mempelajari komputasinya sendiri dengan menggunakan otaknya sendiri yang dirancang untuk terus menganalisis data seperti pada otak manusia dalam pengambilan keputusan (Paliwang *et al.*, 2020). *Convolutional Neural Network* (CNN) adalah pendekatan *Deep Learning* yang populer digunakan untuk memecahkan masalah yang kompleks (Indolia *et al.*, 2018). *Character recognition* merupakan salah satu aspek penting dalam pengembangan teknologi terkait bahasa dan budaya. Dengan melakukan penelitian ini dapat meningkatkan kemampuan pengenalan karakter pada sistem *handwritten recognition* yang digunakan oleh masyarakat Lampung. Pendekatan tersebut terdiri dari *output layer*, serta beberapa *hidden layers*. *Hidden layers* pada CNN biasanya terdiri dari *convolutional layer*, *pooling layer*, *fully connected layer* dan *normalization layer* (Ashqar dan Abu-Naser, 2018).

Penelitian terdahulu terkait klasifikasi Aksara Lampung pernah dilakukan oleh Rikendry dan Maharil pada tahun 2022 dengan menggunakan arsitektur *VGG-16* dan *ResNet-50*. Dari hasil penelitian tersebut arsitektur *VGG-16* memiliki tingkat akurasi yang lebih baik dibandingkan *ResNet-50*. Adapun penelitian terdahulu yang membandingkan arsitektur *LeNet* dan



*AlexNet* dengan menggunakan *dataset Malayalam Handwritten Character*. Dari hasil perbandingan tersebut arsitektur *AlexNet* adalah arsitektur terbaik dalam penelitian tersebut.

Penelitian yang akan dilakukan oleh peneliti, yaitu untuk mengetahui perbandingan kinerja dua arsitektur pada pendekatan CNN, yaitu menggunakan *LeNet* dan *AlexNet* untuk klasifikasi citra Aksara Lampung. Sehingga, dari penelitian tersebut akan diperoleh tingkat akurasi masing-masing arsitektur.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan, rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Mengklasifikasikan Aksara Lampung dengan menggunakan metode arsitektur *LeNet* dan *AlexNet*.
2. Membandingkan hasil tingkat akurasi dalam klasifikasi citra Aksara Lampung menggunakan arsitektur *LeNet* dan *AlexNet*.
3. Melihat pengaruh penggunaan *callbacks API ReduceLRonPlateau* dalam mengoptimalkan laju pembelajaran (*learning rate*) selama pelatihan model arsitektur *LeNet* dan *AlexNet*.

## 1.3 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Penelitian ini menggunakan *dataset* Aksara Lampung sebagai objek penelitian yang diperoleh dari Technische Universitat Dortmund pada bagian *resources*, Lampung *dataset (handwritten characters)* (Junaidi , 2011).
2. Penelitian ini berfokus pada perbandingan kinerja yang dilakukan dengan menggunakan arsitektur *LeNet* dan *AlexNet* untuk klasifikasi

citra Aksara Lampung dengan mencari tingkat akurasi pada kedua arsitektur.

3. Menggunakan *hyperparameter* yang sama untuk arsitektur *LeNet* dan *AlexNet*.

#### **1.4 Tujuan Penelitian**

Tujuan dari penelitian adalah melihat perbandingan kinerja arsitektur *LeNet* dan *AlexNet* untuk klasifikasi citra Aksara Lampung berdasarkan tingkat akurasi uji kedua arsitektur.

#### **1.5 Manfaat Penelitian**

Manfaat dari penelitian ini adalah sebagai berikut:

1. Dapat dapat menjadi dasar dalam pengembangan suatu aplikasi seperti *machine translation system* dalam Bahasa Lampung, aplikasi kamus Bahasa Lampung, *game* edukasi dan lainnya.
2. Dapat dijadikan sebagai bahan acuan dalam mengembangkan penelitian selanjutnya dengan menggunakan *deep learning* dan parameter lainnya mengenai klasifikasi citra Aksara Lampung.
3. Untuk mengembangkan model arsitektur pada penelitian selanjutnya dengan menggunakan aksara dari berbagai wilayah daerah lainnya

## II. TINJAUAN PUSTAKA

### 2.1 Penelitian Terdahulu

Penelitian terdahulu merupakan penelitian yang berfungsi sebagai bentuk perbandingan suatu penelitian yang akan dilakukan dengan penelitian yang sebelumnya sudah pernah dilakukan. Pada bagian ini peneliti mencantumkan berbagai hasil penelitian terdahulu terkait dengan penelitian yang hendak dilakukan. Berikut merupakan penelitian terdahulu yang memiliki keterkaitan dengan tema peneliti pada Tabel 1.

Tabel 1. Penelitian Terdahulu Terkait Klasifikasi Tulisan Tangan.

No	Nama Peneliti (Tahun Penelitian)	Judul Penelitian	Data	Hasil Penelitian
1	Ajay James, Manjusha J, dan Chandran Saravanan (2018)	<i>Malayalam Handwritten Character Recognition Using AlexNet Based Architecture</i>	<i>Dataset P-ARTS KAYYEZHUTU</i> terdiri dari 90.000 data dan <i>Dataset Compoun Characters</i> terdiri dari 100.000 data dengan 36 kelas karakter.	Pada penelitian ini <i>learning rate</i> yang ditetapkan sebesar 0,001. <i>Epoch</i> yang digunakan berbeda seperti 5, 10, 20 dan 30 dicoba dan ditetapkan menjadi 20. Nilai <i>accuracy</i> yang diperoleh menggunakan Arsitektur <i>AlexNet</i> sebesar 98,74% dan Arsitektur <i>LeNet</i> sebesar 85,25% dengan menggunakan <i>Dataset P-ARTS Kayyvezhuthu</i> . Sedangkan nilai <i>accuracy</i> yang diperoleh menggunakan

				Arsitektur <i>AlexNet</i> sebesar 98,42% dan Arsitektur <i>LeNet</i> sebesar 87,54% dengan menggunakan <i>Dataset Compoun Characters</i> . Dalam penelitian ini Arsitektur <i>AlexNet</i> memiliki tingkat akurasi yang lebih baik dibandingkan Arsitektur <i>LeNet</i> .
2	Rikendry dan Azzin Maharil (2022)	Perbandingan Arsitektur <i>VGG-16</i> Dan <i>ResNet-50</i> Untuk Rekognisi Tulisan Tangan Aksara Lampung	<i>Dataset</i> Gambar Tulisan Tangan Aksara Lampung yang terdiri dari 4.780 data dengan 20 kelas karakter.	Hasil dari penelitian ini menunjukkan bahwa model Arsitektur <i>VGG-16</i> memberikan hasil <i>accuracy</i> sebesar 81,28% untuk hasil pengujian serta waktu pelatihan yang lebih baik, sedangkan untuk <i>ResNet-50</i> memberikan hasil <i>accuracy</i> sebesar 66,67% untuk hasil pengujian dan memerlukan waktu komputasi yang lebih lama dan hasil akurasi yang kecil. disimpulkan bahwa arsitektur <i>VGG-16</i> adalah arsitektur yang cukup baik dalam hal akurasi maupun jumlah parameter dibandingkan arsitektur <i>ResNet-50</i>
3	Muhammad Ezar Al Rivin dan Alwyn Giovri Riyadi	Perbandingan Arsitektur <i>LeNet</i> dan <i>AlexNet</i> Pada Metode <i>Convolutional Neural Network</i> Untuk	<i>Dataset American Sign Language</i> terdiri dari 24.000 data dengan 24 kelas alfabet.	Hasil dari penelitian ini, Arsitektur <i>LeNet</i> merupakan arsitektur terbaik dibandingkan <i>AlexNet</i> . Arsitektur <i>LeNet</i> memiliki struktur yang lebih ringkas dibandingkan dengan Arsitektur <i>AlexNet</i> . <i>LeNet</i>

		Pengenalan <i>American Sign Language.</i>		menghasilkan nilai <i>accuracy</i> 92,47%, <i>precision</i> 92,88%, dan <i>recall</i> 92,47%. <i>AlexNet</i> menghasilkan nilai <i>accuracy</i> 91,62%, <i>precision</i> 92,07%, dan <i>recall</i> 91,60% .
--	--	------------------------------------------------------	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------




















## 2.2 Aksara Lampung

Aksara Lampung atau yang disebut dengan Had Lampung merupakan salah satu kekayaan Indonesia. Bahasa Lampung utamanya digunakan sebagai sarana komunikasi sesama penutur rumpun bahasa Lampung. Bahkan saat masa pra kemerdekaan Indonesia, orang Lampung akan merasa malu bila tidak fasih membaca dan menulis Aksara Lampung.

Arah penulisan Aksara Lampung adalah kiri ke kanan. Gabungan antara Aksara Lampung, Rejang, Lembak, Serawai, Pasemah, dan Incung membentuk rumpun Aksara Hulu (Surat Ulu) (Sarwono dan Rahayu, 2014). Rumpun aksara ini memiliki ciri khas, yaitu bentuknya berupa goretan patah-patah/lengkung, tidak memiliki pasangan, dan bentuk lebih sederhana ketimbang keturunan Aksara Kawi lainnya (seperti Jawa dan Bali).













Had Lampung atau Surat Lampung, disebut juga dengan Kaganga dan memiliki gaya visual aksara sendiri, karena tiga huruf awal aksara berbunyi demikian (*ka-ga-nga*) (Budiman *et al.*, 2019). Huruf induk dalam Aksara Lampung lama berjumlah 19 buah yang ditulis dan dibaca dari kiri ke kanan. Anak huruf dalam Aksara Lampung berfungsi sebagai penambah vokal dalam huruf induk (Indrayati dan Migotuwio, 2020). Aksara Lampung mempunyai tiga unsur yaitu induk huruf (*kelabai surat*), anak huruf atau tanda bunyi (*benah surat*), dan tanda-tanda baca. Berikut 19 karakter aksara dasar (*kelabai surat*) Lampung beserta huruf latin yang

merepresentasikan satu suku kata dengan vokal inheren diakhiri dengan huruf "a" pada Gambar 1.

Aksara Dasar Lampung (kelabai surat)				
 a	 ba	 ca	 da	 ga
 ha	 ja	 ka	 la	 ma
 na	 nga	 nya	 pa	 ra
 sa	 ta	 wa	 ya	

Gambar 1. Aksara Dasar (*kelabai surat*) Lampung.

Diakritik (*benah surat*) adalah tanda yang ditambahkan pada aksara dasar untuk mengubah vokal inheren aksara dasar tersebut dan/atau menutup suatu suku kata dengan konsonan. Dalam Aksara Lampung terdapat tanda baca seperti aksara Arab, tanda yang letaknya di atas huruf, yaitu *fathah* sedangkan tanda baca yang di bawah, yaitu *kasrah*. Dalam Aksara Lampung terdapat 12 jenis diakritik yang digunakan, dapat dilihat pada Gambar 2.

Di Atas Aksara					
-i	-è	-e	-n	-ng	-r
					
ulan	ulan	bicek	datas	teklubang	rejengjung
Di Bawah Aksara			Sejajar Aksara		
-u	-o	-w	-y	-h	virama
					
bitan	bitan	teklengu	teklingai	klengiyah	nengen


Gambar 2. Diakritik (*benah surat*) pada Aksara Dasar.

Dengan catatan pada Gambar 2, [1] /e/ sebagaimana 'e' dalam kata "enak", [2] /ə/ sebagaimana 'e' dalam kata "empat" dan [3] Diakritik baru (tidak ditemukan di naskah bersejarah). Berikut contoh penggunaan 12 jenis diakritik pada Aksara Lampung "pa" pada Gambar 3.

Contoh Penggunaan Diakritik pada Aksara Lampung "Pa"					
ulan	ulan	bicek	datas	teklubang	rejengjung
					
pi	pè	pe	pan	pang	par
bitan	bitan	teklengu	teklingai	klengiyah	nengen
					
pu	po	pau	pai	pah	p

Gambar 3. Contoh Penggunaan Diakritik pada Aksara Lampung.

Aksara Lampung memiliki lima tanda baca, awalnya hanya memiliki dua tanda baca diantaranya, yaitu bulatan matahari dan bulan yang merupakan tanda baca, yang keduanya masing-masing berperan sebagai tanda awal dan akhir dari suatu paragraf atau teks. Tanda baca lainnya disahkan dalam Musyawarah Pembakuan Aksara Lampung yang dilaksanakan 38 tahun lalu tepat pada tanggal 23 Februari 1985 (Indrayanti dan Migotuwio, 2020). Tanda baca dalam Aksara Lampung dapat dilihat pada Gambar 4 dibawah ini.

kuma	beradu	seru	ngulih	ngemula
				
Tanda koma	Tanda titik	Tanda seru	Tanda tanya	Tanda pembuka paragraf

Gambar 4. Tanda Baca pada Aksara Lampung.

### 2.3 *Deep Learning*

*Deep Learning* merupakan sub bidang *machine learning* yang algoritmanya terinspirasi dari struktur otak manusia. Model *Deep Learning* dapat mengenali pola kompleks dalam gambar, teks, suara, dan data lain untuk menghasilkan wawasan dan prediksi yang akurat. *Deep Learning* memiliki kemampuan yang baik dalam *computer vision*, yaitu *image classification* atau klasifikasi objek pada citra dalam bentuk dua dimensi misalnya gambar dan suara (Pratiwi *et al.*, 2021). Metode *Deep Learning* menggunakan CPU dan RAM pada saat melakukan komputasi, dan dapat memanfaatkan GPU untuk mempercepat pengolahan data dalam jumlah banyak (Ilahiyah dan Nilogiri, 2018).

*Deep Learning* memiliki keunggulan, yaitu kemampuannya untuk memperoleh fitur-fitur penting secara otomatis sehingga lebih efisien dan akurat. Semakin banyak data yang digunakan juga mempengaruhi tingkat

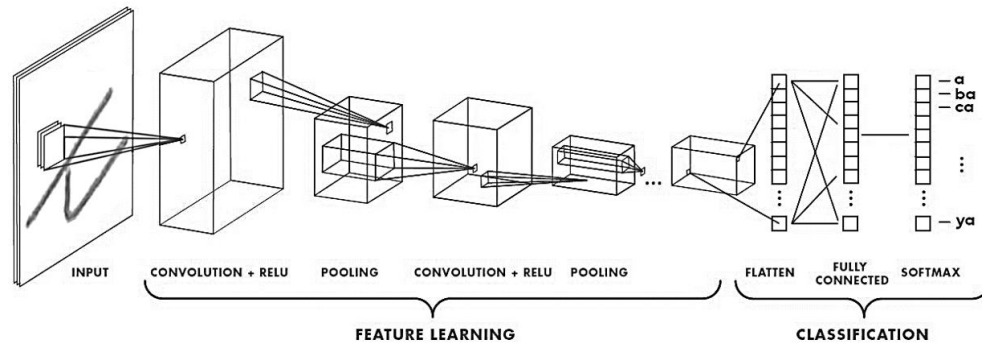


akurasi dan ketepatan mengidentifikasi suatu objek. *Deep Learning* juga memiliki kemampuan untuk melakukan *transfer learning*, yaitu memanfaatkan model yang sudah ada untuk memecahkan masalah yang berbeda. Hal ini sangat berguna dalam situasi ketika data yang ada terbatas atau saat ingin menghemat waktu dan biaya dalam membangun model baru. *Deep Learning* terdiri dari beberapa jaringan saraf tiruan yang saling berhubungan seperti *Convolutional Neural Network* (CNN), *Recurrent Neural Network* (RNN), *Long Short Term Memory Network* (LSTM), *Self Organizing Maps* (SOM) dan lainnya.

#### **2.4 Convolutional Neural Network (CNN)**

*Convolutional Neural Network* (CNN/*ConvNet*) adalah salah satu algoritma dari *Deep Learning* yang merupakan pengembangan dari *Multilayer Perceptron* (MLP) yang dirancang untuk mengolah data dalam bentuk dua dimensi, misalnya gambar atau suara (Emanuella *et al.*, 2021). CNN termasuk dalam jenis *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. Nama CNN menunjukkan bahwa jaringan tersebut menggunakan operasi matematika yang disebut konvolusi. CNN merupakan jaringan saraf yang menggunakan konvolusi sebagai ganti dari penerapan matriks umum dimana paling tidak terdapat satu konvolusi di setiap *layer*. Algoritma ini adalah jenis jaringan saraf khusus untuk memproses data yang memiliki kemiripan *grid topology*.

CNN digunakan untuk mengklasifikasi data yang terlabel dengan menggunakan metode *supervised learning*. CNN terdiri dari *neuron* yang memiliki *weight*, bias dan *activation function*. Proses yang dilakukan oleh *Convolutional Neural Network*, yaitu: *Convolutional Layer*, *Non-Linearity Layer* (ReLU *Layer*), *Pooling Layer*, dan *Fully Connected Layer* (Lorentius *et al*, 2019).



Gambar 5. Ilustrasi Arsitektur CNN.

Dari Gambar 5 di atas dapat dilihat bahwa terdapat dua lapisan arsitektur, yaitu:

**a. *Feature Learning***

*Feature Learning* merupakan lapisan pertama dari CNN, proses yang terjadi pada bagian ini adalah melakukan “*encoding*” dari sebuah gambar menjadi *features* yang berupa angka-angka yang merepresentasikan gambar tersebut (*feature extraction*). *Feature extraction layer* terdiri dari dua bagian, yaitu *convolutional layer* dan *pooling layer*. Namun terkadang ada beberapa riset yang tidak menggunakan *pooling layer* (Suhardin *et al.*, 2021). Lapisan yang terdapat dalam proses ini terdiri atas lapisan *pooling* dan lapisan konvolusi, yang mana pada masing-masing proses lapisan akan menghasilkan *feature maps* dalam bentuk angka-angka yang merepresentasikan gambar yang kemudian akan diteruskan ke bagian lapisan klasifikasi.

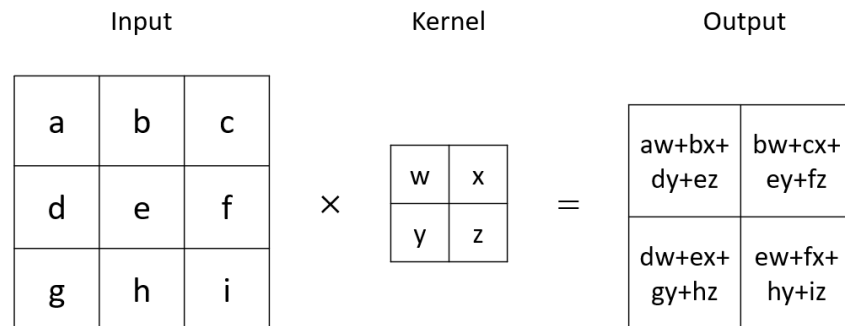
**b. *Classification Layer***

Pada lapisan *Classification Layer* terdiri dari lapisan *neuron* yang tersambung seluruhnya dengan lapisan yang lainnya (*fully connected*). Lapisan ini menerima *input* dari hasil keluaran *layer*

ekstraksi fitur gambar berupa vektor, kemudian ditransformasikan seperti *multi neural networks* dengan tambahan beberapa *hidden layer* (Ilahiyah dan Nilogiri, 2018). Lapisan ini menerima *input* yang berasal dari *output layer* dari bagian *feature learning* yang kemudian akan di proses di *flatten* dengan menambahkan beberapa *hidden layer* pada *fully connected* sehingga menghasilkan keluaran berupa nilai *accuracy* dari klasifikasi setiap kelas.

#### 2.4.1 Convolutional Layer

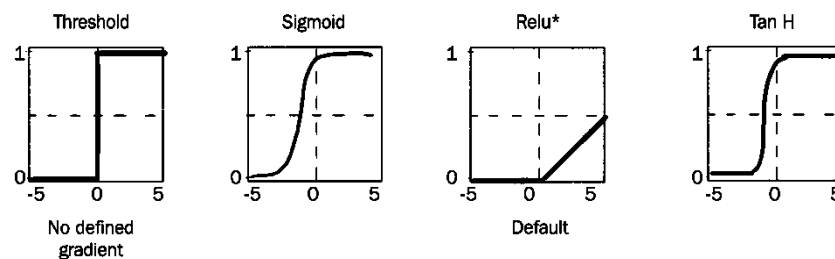
*Convolutional Layer* adalah lapisan utama yang sangat penting untuk digunakan. Lapisan ini digunakan untuk melakukan operasi konvolusi pada *output layer* sebelumnya. *Layer* ini termasuk blok utama pada *Convolutional Neural Network (CNN)* yang di dalamnya terdiri dari filter – filter yang dipelajari secara acak untuk melakukan operasi konvolusi yang bertujuan sebagai ekstraksi fitur untuk mempelajari representasi fitur dari *input layer* (Alwanda *et al.*, 2020).



Gambar 6. Operasi Konvolusi.

### 2.4.2 Fungsi Aktivasi

Fungsi aktivasi atau fungsi transfer adalah fungsi *non-linear* yang memungkinkan jaringan untuk memecahkan masalah *non-trivial*. Setiap fungsi aktivasi mengambil sebuah nilai dan melakukan operasi matematika. Dalam arsitektur CNN, fungsi aktivasi terletak pada perhitungan akhir keluaran *feature map* atau sesudah proses perhitungan konvolusi atau *pooling* untuk menghasilkan suatu pola fitur (Zufar dan Setiyono, 2016). Beberapa fungsi aktivasi yang sering digunakan seperti fungsi *sigmoid*, *tanh*, *ReLU*, dan lainnya.

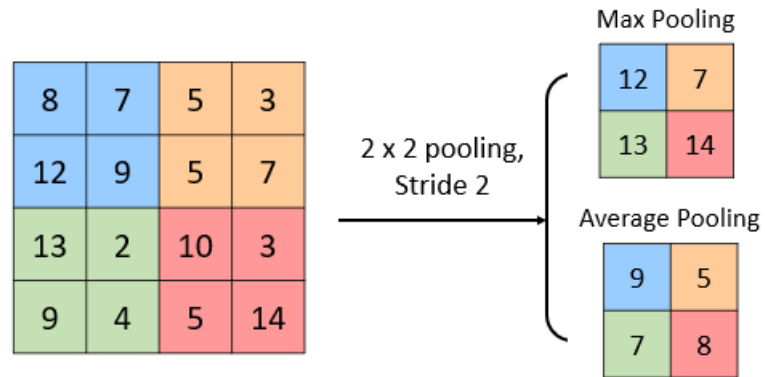


Gambar 7. Fungsi Aktivasi (*Sigmoid*, *Tanh*, *ReLU*).  
(Tashmit, 2023)

### 2.4.3 Pooling Layer

*Pooling layer* adalah lapisan yang berfungsi untuk mereduksi input secara spasial (mengurangi jumlah parameter) dengan operasi *down-sampling*. Dikarenakan tahapan ini merupakan *downsampling* sehingga memperkecil parameter, maka kecepatan komputasi dapat lebih meningkat. (Alwanda *et al.*, 2020). Umumnya, metode *pooling* yang sering digunakan adalah *max pooling* dan *average pooling*. *Max pooling* lebih banyak digunakan karena dengan *max pooling filter* dapat mengeksplorasi nilai tertinggi pada citra. Sedangkan

*average pooling* mencari nilai rata-rata pada tiap dimensi (Alamgunawan dan Kristian, 2020).



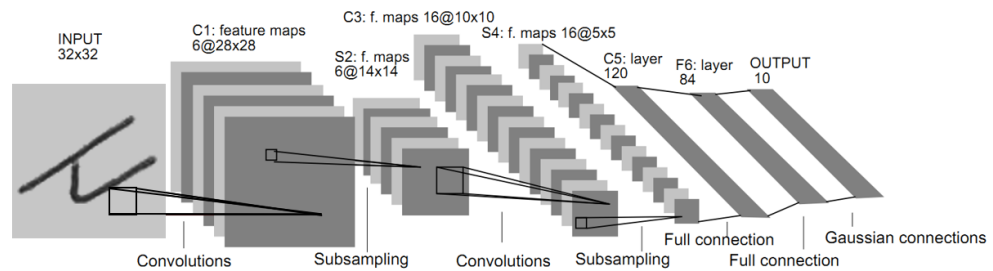
Gambar 8. *Pooling Layer*.

#### 2.4.4 *Fully-Connected Layer*

*Fully Connected Layer* digunakan pada MLP (*Multi Layer Perceptron*) yang mempunyai tujuan untuk melakukan transformasi pada dimensi data agar data dapat diklasifikasikan secara linear (Hendriyana dan Maulana, 2019). Untuk mendapatkan hasil keluaran dari *layer* ini tidak dibutuhkan operasi konvolusi, tetapi menggunakan komputasi perkalian matriks yang diikuti dengan bias *offset*. Proses dari *fully connected layer* terletak di akhir arsitektur yang mana prosesnya direpresentasikan sebagai perkalian dari matriks sederhana yang diikuti dengan penambahan vektor bias dan menerapkan fungsi *non-linear*.

## 2.5 *LeNet*

*LeNet* (LeCun, 1998) adalah suatu jaringan yang memiliki lapisan banyak berbasis *Convolutional Neural Network* (CNN) pertama kali yang dikenalkan oleh Yann LeCun pada tahun 1998, dalam *paper* penelitian yang berjudul *Gradient-Based Learning Applied to Document Recognition*. Alasan utama dibalik popularitas model ini adalah arsitekturnya yang sederhana dan *straightforward*. *LeNet* merupakan arsitektur CNN pertama yang digunakan untuk *handwritten recognition tasks* (Pius dan Johny, 2020). Arsitektur *LeNet* merupakan arsitektur yang cocok dan biasa digunakan untuk *handwritten recognition*. *LeNet* memiliki dua bagian *layer*, yaitu *feature extraction* dan klasifikasi yang digunakan untuk mengklasifikasikan objek (Zhao dan Liu, 2020). *LeNet* terdiri dari *input layer*, dua *convolutional layers*, dua *pooling layers* dan tiga *fully connected layers*.

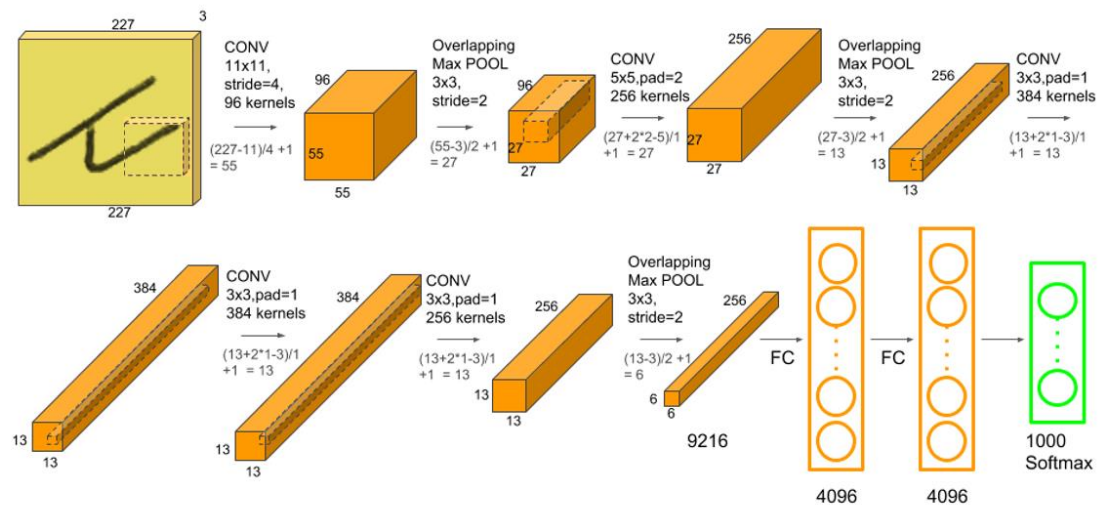


Gambar 9. Arsitektur *LeNet*.

## 2.6 *AlexNet*

*AlexNet* memenangkan kompetisi *Imagenet LSVRC (Large Scale Visual Recognition Challenge)* pada tahun 2012. LSVRC adalah kompetisi dimana tim peneliti mengevaluasi algoritma mereka pada kumpulan data besar gambar berlabel (*imageNet*) dan bersaing untuk mencapai akurasi yang

lebih tinggi pada beberapa tugas pengenalan visual. Model tersebut diusulkan pada tahun 2012 dalam *paper* penelitian berjudul *Imagenet Classification with Deep Convolutional Neural Network* oleh Alex Krizhevsky bekerja sama dengan Ilya Sutskever dan Geoffrey Hinton. Arsitektur *AlexNet* terdiri atas 5 *convolution layer*, 3 *pooling layer*, 2 *dropout layer*, dan 3 *fully connected layer* (Irfansyah *et al.*, 2021). Dalam model ini, kedalaman jaringan ditingkatkan dibandingkan dengan Arsitektur *LeNet*.



Gambar 10. Arsitektur *AlexNet*.

## 2.7 *Hyperparameter*

*Hyperparameter* memiliki peran yang sangat penting dalam mengoptimalkan kinerja suatu algoritma. *Hyperparameter* harus ditetapkan sebelum sebuah model menjalani proses pembelajarannya. Beberapa peneliti mengubah satu *hyperparameter* pada satu waktu dan mengukur pengaruhnya terhadap kinerja model (Nugraha dan Sasongko, 2022). *Hyperparameter* adalah variabel yang mempengaruhi *output* model dengan nilai tidak diubah selama model dioptimisasi. Model dengan jenis yang

sama namun *hyperparameter* berbeda bisa memiliki bentuk *output* yang berbeda pula. *Hyperparameter* yang biasa digunakan dalam suatu model yaitu *epoch*, *optimizer*, *learning rate*, dan *batch size*.

### 2.7.1 *Epoch*

*Epoch* adalah *hyperparameter* yang menentukan berapa kali algoritma pembelajaran akan bekerja mengolah seluruh *dataset training*. Satu *epoch* berarti bahwa setiap sampel dalam set data pelatihan memiliki kesempatan untuk memperbarui parameter model internal (Brownlee, 2022).

### 2.7.2 *Optimizer*

*Optimizer* adalah algoritma yang digunakan untuk mengganti parameter seperti *weight* dan bias pada *neural network* untuk mengurangi *error/loss*. Ada banyak sekali algoritma *optimizer* seperti *Gradient Descent*, *Stochastic Gradient Descent*, *Adaptive Moment Estimation*, *RMSprop*, dan lain sebagainya (Fawwaz et al., 2020).

### 2.7.3 *Learning Rate*

*Learning rate* merupakan salah satu parameter *training* untuk menghitung nilai koreksi bobot pada waktu proses *training*. Nilai *learning rate* ini berada pada *range* nol (0) sampai (1). Semakin besar nilai *learning rate*, maka proses *training* akan berjalan semakin cepat. Semakin besar *learning rate*, maka ketelitian jaringan akan semakin berkurang, tetapi berlaku sebaliknya, apabila *learning rate*-nya semakin kecil, maka ketelitian jaringan akan semakin besar atau bertambah dengan konsekuensi proses *training* akan memakan waktu yang semakin lama. *Hyperparameter* ini



tentunya akan mempengaruhi keakuratan jaringan suatu sistem (Ridwan *et al.*, 2020).

#### 2.7.4 *Batch Size*

*Batch size* juga perlu ditentukan dalam *training*. *Batch size* adalah istilah yang digunakan dalam pembelajaran mesin dan mengacu pada jumlah contoh pelatihan yang digunakan dalam satu iterasi dan merupakan salah satu *hyperparameter* terpenting untuk disesuaikan dengan sistem *deep learning* (Rochmawati *et al.*, 2021).

### 2.8 *Confusion Matrix*

*Confusion Matrix* dikenal juga dengan sebutan *error matrix* yang memvisualisasikan kinerja suatu algoritme berupa tabel khusus. Pada dasarnya *confusion matrix* memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh sistem (model) dengan hasil klasifikasi sebenarnya (Hakim *et al.*, 2023). Representasi hasil pada *confusion matrix* dapat ditunjukkan dengan Gambar 11 dimana berdasarkan tabel *confusion matrix* tersebut terdapat kondisi *actual* dan kondisi kelas prediksi/deteksi.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 11. *Confusion Matrix*.

Keterangan:

- a. *True Positive* (TP), merupakan jumlah data positif yang diprediksi benar.
- b. *True Negative* (TN), merupakan jumlah data negatif yang diprediksi benar.
- c. *False Positive* (FP) – *Type I Error*, merupakan jumlah data negatif namun diprediksi sebagai data positif.
- d. *False Negative* (FN) – *Type II Error*, merupakan jumlah data positif namun diprediksi sebagai data negatif.

Berdasarkan *confusion matrix* diatas terdapat beberapa metrik evaluasi untuk mengukur kinerja model adalah *accuracy*, *recall*, *precision* dan *f1-score* (Assegaf dan Wibowo, 2021):

### 2.8.1 Accuracy

*Accuracy* menggambarkan seberapa akurat model dapat mengklasifikasikan dengan benar. Maka, *accuracy* merupakan rasio prediksi benar (positif dan negatif) dengan keseluruhan data. Dengan kata lain, *accuracy* merupakan tingkat kedekatan nilai prediksi dengan nilai aktual (sebenarnya) (Rustam *et al.*, 2021). Nilai *accuracy* dapat diperoleh dengan Persamaan (1).

$$Accuracy = \frac{\text{correct predictions}}{\text{total predictions}} \dots\dots\dots(1)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots(1)$$

### 2.8.2 Recall

*Recall* atau *sensitivitas* (*True Positive Rate* (TPR)) menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi. Maka, *recall* merupakan rasio prediksi benar positif

dibandingkan dengan keseluruhan data yang benar positif (Rustam *et al.*, 2021). Nilai *recall* dapat diperoleh dengan Persamaan (2).

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots(2)$$

### 2.8.3 Precision

*Precision* menggambarkan tingkat keakuratan antara data yang diminta dengan hasil prediksi yang diberikan oleh model. Maka, *precision* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Dari semua kelas positif yang telah diprediksi dengan benar, berapa banyak data yang benar-benar positif (Rustam *et al.*, 2021). Nilai *precision* dapat diperoleh dengan Persamaan (3).

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots(3)$$

### 2.8.4 F1-Score

*F1-Score* menggambarkan perbandingan rata-rata *precision* dan *recall* yang dibobotkan. *Accuracy* tepat kita gunakan sebagai acuan performansi algoritma jika dataset kita memiliki jumlah data *False Negatif* dan *False Positif* yang sangat mendekati (*symmetric*). Namun jika jumlahnya tidak mendekati, maka sebaiknya kita menggunakan *f1-score* sebagai acuan. Nilai ini menunjukkan keseimbangan antara *precision* dan *recall* (Rustam *et al.*, 2021). Nilai *F1-Score* dapat diperoleh dengan Persamaan (4).

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \dots\dots\dots(4)$$

## 2.9 *Multiclass Classification*

*Multiclass Classification* adalah merupakan kondisi di mana sebuah *dataset* memiliki lebih dari dua label atau kelas (*multi-class*) (Diantika *et al.*, 2023). Dalam klasifikasi ini, algoritma pembelajaran mesin atau model statistik dilatih pada set data yang telah diberi label dan diberikan tugas untuk mempelajari pola dalam data tersebut sehingga dapat memprediksi label kelas yang sesuai untuk data baru yang belum diberi label.

## 2.10 Teknik Augmentasi

Augmentasi data merupakan proses memodifikasi atau memanipulasi suatu citra sehingga menghasilkan data tambahan dan meningkatkan variasi data pelatihan agar model pembelajaran mesin dapat lebih umum dan dapat menggeneralisasi dengan baik pada data baru. Pada kebanyakan kasus, penggunaan augmentasi data berhasil menaikkan performa dari model. Peningkatan terjadi karena model berhasil mengenali lebih banyak objek dari bentuk serta pola yang beragam jenisnya (Zuhri *et al.*, 2022). Adapun teknik augmentasi yang umum digunakan yaitu *rotation*, *translation*, *cropping*, *flipping*, *shear*, *scaling* dan lainnya.

### 2.10.1 *Random Rotation*

Teknik augmentasi *random rotation* melibatkan menggabungkan rotasi dalam berbagai derajat acak pada setiap data gambar dalam dataset. Tujuannya adalah menciptakan variasi dalam orientasi objek dalam gambar. Hasilnya adalah setiap gambar dalam *dataset* akan diputar dengan sudut rotasi acak. Jika koordinat  $(x', y')$  akan menjadi  $x' = y$  dan  $y' = -x$  untuk rotasi  $90^\circ$ ,  $x' = -x$  dan  $y' = -y$  untuk rotasi untuk rotasi  $180^\circ$ , dan  $x' = -y$  dan  $y' = x$  untuk rotasi  $270^\circ$ .

### 2.10.2 *Random Shear*

Teknik augmentasi *random shear* menerapkan peregangan (*shear*) acak dalam sumbu x (horizontal) dan/atau sumbu y (vertikal) pada setiap data gambar dalam *dataset*. Peregangan acak ini menghasilkan variasi dalam bentuk objek dalam gambar. Jika koordinat x mengalami pergeseran, nilai shear positif maka x akan bergeser ke kanan. Jika nilai shear negatif, maka x akan bergeser ke kiri. Jika koordinat y mengalami pergeseran, nilai positif akan mendorong y ke bawah, sedangkan nilai negatif akan mendorong y ke atas. Jika koordinat (x', y') akan menjadi  $x' = x + shear\_factor * y$  untuk *shear* horizontal (x) dan  $y' = y + shear\_factor * x$  untuk *shear* vertikal (y).



## 3.2 Alat dan Data

Alat dan data yang digunakan dalam penelitian ini adalah sebagai berikut:

### 3.2.1 Alat

Alat yang digunakan dalam penelitian ini adalah sebagai berikut:

#### 3.2.1.1 Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam penelitian ini, yaitu Laptop ASUS Vivobook dengan spesifikasi RAM 8.00 Gb, 500 Gb SSD dan *processor* 11th Gen Intel® Core™ sebagai *hardware* utama dalam penelitian.

#### 3.2.1.2 Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam penelitian ini yaitu: Sistem Operasi Windows 11 64-bit, Google Colaboratory, Jupyter Notebook, Google Drive, Notepad dan beberapa jenis *Library Python* seperti:

a. *Library Numpy 1.22.4*

*Library Numpy* adalah *library python* yang fokus pada *scientific computing*. *Library* ini berfungsi untuk proses komputasi numerik. *NumPy* memiliki kemampuan untuk membuat objek N-dimensi *array*. *Array* merupakan sekumpulan variabel yang memiliki tipe data yang sama.

b. *Library Keras 2.12.0*

*Library Keras* adalah API pembelajaran mendalam yang ditulis dengan *python*, berjalan di atas platform pembelajaran mesin *TensorFlow*. *Library* ini berfungsi untuk mengurangi jumlah tindakan yang diperlukan dalam mengimplementasikan kode umum serta dapat menjelaskan kesalahan pengguna agar dapat ditindaklanjuti.

c. *Library TensorFlow 2.12.0*

*Library TensorFlow* adalah *platform end-to-end open-source* untuk membuat aplikasi *machine learning* atau komputasi numerik cepat yang dibuat dan dirilis oleh Google.

d. *Library Matplotlib 3.7.1*

*Library Matplotlib* adalah *library* yang berfungsi untuk merencanakan data numerik. *Library* ini bersifat *open source* ini dapat memplot angka-angka berdefinisi tinggi seperti diagram lingkaran, histogram, *scatter plot*, grafik, dan lain-lain.

e. *Library SciKit-Learn 1.2.2*

*Library Scikit-Learn* atau *Sklearn* adalah *library* berbasis *python* untuk membangun model pembelajaran mesin yang menyediakan banyak algoritma pembelajaran untuk regresi, pengelompokan, dan klasifikasi.

f. *Library Os*

*Library Os* adalah *library* yang menyediakan fungsi untuk berinteraksi dengan sistem operasi.

g. *Library Seaborn 0.12.2*

*Library Seaborn* adalah *library* untuk membuat grafik dan statistik dengan menggunakan *python*. *Library* ini dibangun berdasarkan *library matplotlib* yang sudah ada.



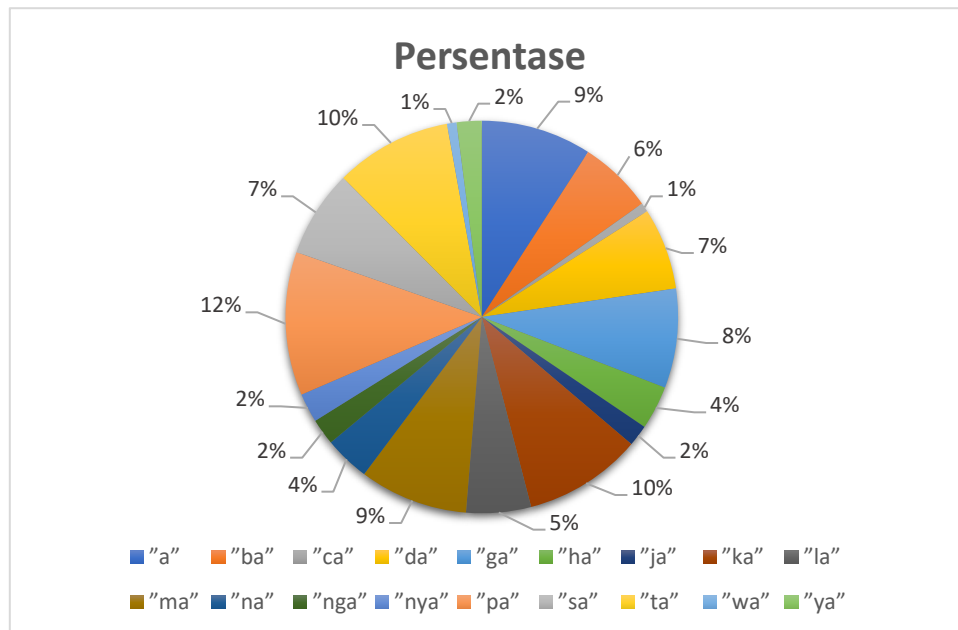
### 3.2.2 Data

Data yang digunakan dalam penelitian ini, yaitu *Lampung Handwritten Characters Dataset* (Junaidi *et al.*, 2011) sebagai objek penelitian yang diperoleh dari Technische Universitat Dortmund pada bagian *resources*, *Lampung dataset (handwritten characters)*. Dalam data tersebut terdapat jumlah kontributor yang berpartisipasi dalam pengumpulan data sebanyak 82 penulis. Kontributor tersebut adalah peserta didik SMKN 4 Bandar Lampung yang terdiri dari 20 laki-laki dan 62 perempuan berusia rata-rata 16 tahun. Para siswa tersebut diinstruksikan untuk mentranskripsikan sebagian teks yang dicetak dalam aksara latin dari dongeng Indonesia menjadi teks tulisan tangan dalam Aksara Lampung.

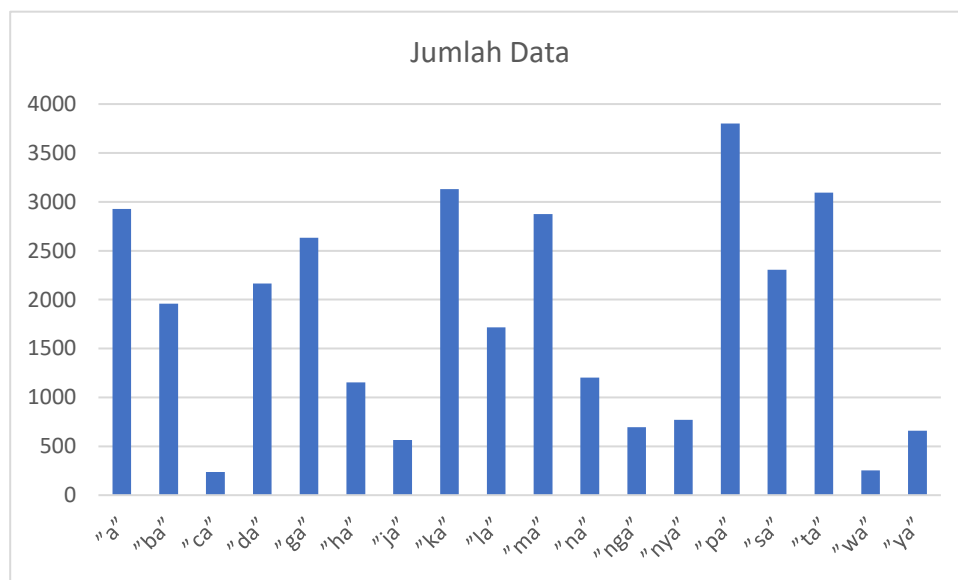
Dataset terdiri dari tiga komponen yaitu *raw images*, *extracted connected components* dan *annotation files*. Dalam penelitian ini peneliti menggunakan *dataset annotation files* dan *extracted connected components*. dalam folder *extracted connected components* terdapat *char\_css* yang berisikan 82 folder dengan total 32.140 *file* data citra Aksara Lampung. *Annotation files* berisi 82 *file* dengan ekstensi *anno*, dan jumlah *fields* sebanyak enam *columns*. Dimana kolom ke-1, yaitu nama *file extracted connected component* (gambar yang dapat dilihat pada folder *char\_css*), kolom ke-2 hingga ke-5, yaitu koordinat *bounding box* dari *connected component* di dalam dokumen *raw images* dan kolom ke-6 merupakan anotasi karakter, yaitu label karakter berdasarkan Aksara Lampung yang terdiri dari 18 kelas. Nama kelas dan jumlah data per kelas yang digunakan dapat dilihat pada Tabel 3.

Tabel 3. Jumlah Data Per Kelas.

<b>No.</b>	<b>Nama Kelas</b>	<b>Jumlah Data</b>	<b>Persentase</b>
<b>1.</b>	<b>"a"</b>	2928	9%
<b>2.</b>	<b>"ba"</b>	1957	6%
<b>3.</b>	<b>"ca"</b>	238	1%
<b>4.</b>	<b>"da"</b>	2164	7%
<b>5.</b>	<b>"ga"</b>	2633	8%
<b>6.</b>	<b>"ha"</b>	1155	4%
<b>7.</b>	<b>"ja"</b>	563	2%
<b>8.</b>	<b>"ka"</b>	3131	10%
<b>9.</b>	<b>"la"</b>	1715	5%
<b>10.</b>	<b>"ma"</b>	2874	9%
<b>11.</b>	<b>"na"</b>	1201	4%
<b>12.</b>	<b>"nga"</b>	695	2%
<b>13.</b>	<b>"nya"</b>	772	2%
<b>14.</b>	<b>"pa"</b>	3802	12%
<b>15.</b>	<b>"sa"</b>	2305	7%
<b>16.</b>	<b>"ta"</b>	3093	10%
<b>17.</b>	<b>"wa"</b>	254	1%
<b>18.</b>	<b>"ya"</b>	660	2%
<b>TOTAL</b>		32.140	100%



Gambar 12. *Pie Chart* Proporsi Distribusi Kemunculan Karakter Aksara Dalam Korpus.

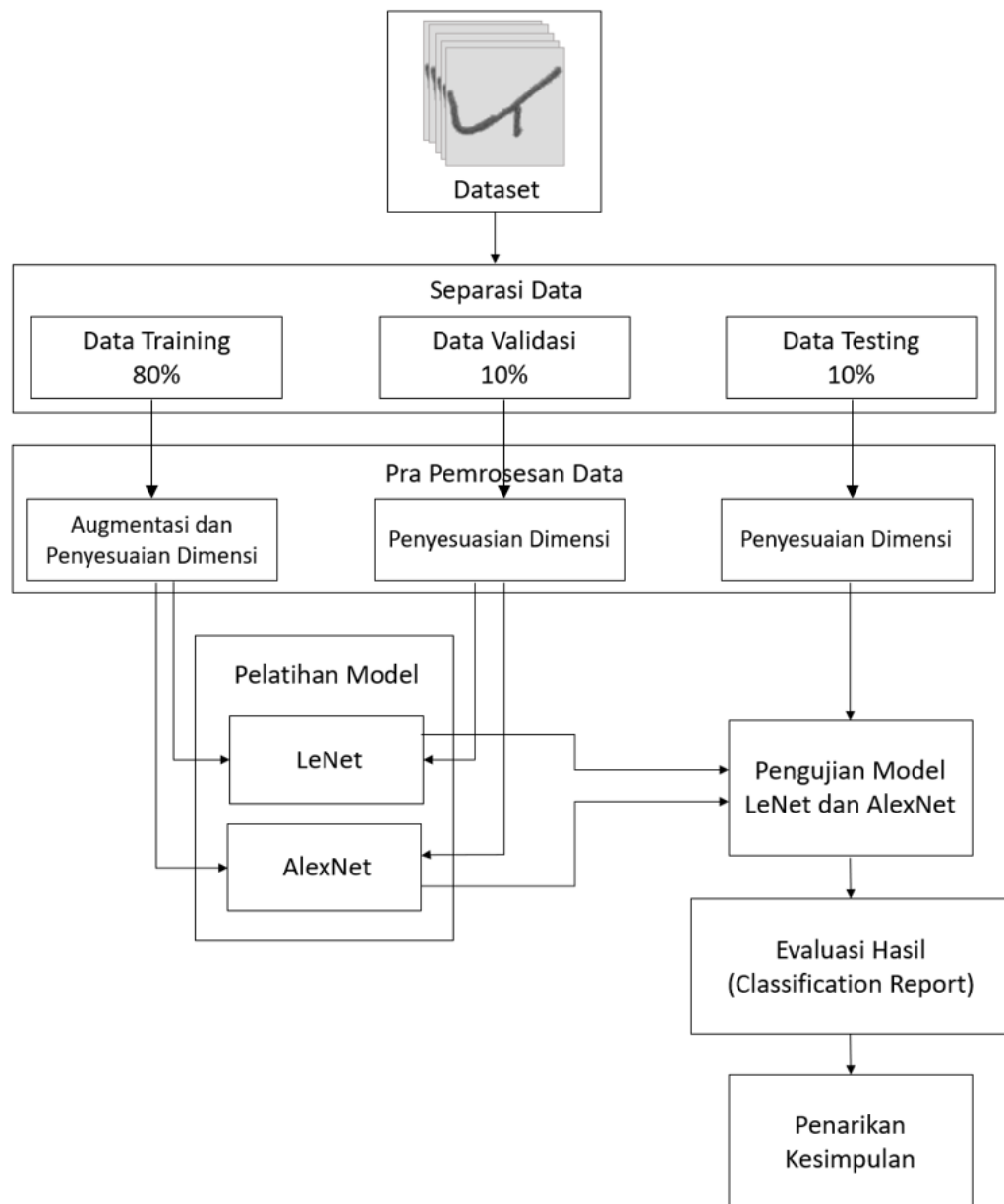


Gambar 13. *Bar Chart* Frekuensi Kemunculan Karakter Aksara Dalam Korpus.

Berdasarkan pada Gambar 12 dan 13 untuk nama kelas dan visualisasi data menggunakan *Pie Chart* dan *Bar Chart* memiliki rata-rata jumlah perkelasnya 1.786 data dari visualiasi tersebut dapat dilihat bahwa aksara "pa" memiliki jumlah data paling banyak, yaitu 3802 data. Hal ini karena aksara "pa" adalah aksara yang paling banyak digunakan dalam kata-kata Bahasa Lampung, aksara "pa" digunakan untuk menuliskan bunyi konsonan /p/ pada posisi awal, tengah, dan akhir suku kata. Oleh karena itu, aksara "pa" menjadi aksara yang lebih sering muncul dalam penulisan kata-kata dalam Bahasa Lampung. sehingga frekuensi penggunaannya lebih tinggi dan sering muncul dibandingkan dengan aksara lainnya dalam Bahasa Lampung.

### 3.3 Tahapan Penelitian

Adapun tahapan-tahapan penelitian yang digunakan dalam perbandingan arsitektur *LeNet* dan *AlexNet* untuk klasifikasi citra Aksara Lampung dapat dilihat pada Gambar 14.



Gambar 14. Alur Tahapan Penelitian Klasifikasi Menggunakan Arsitektur *LeNet* dan *AlexNet*.

### 3.3.1 *Dataset*

*Dataset* yang digunakan dalam penelitian ini sebagai objek diperoleh dari Technische Universitat Dortmund pada bagian *resources*, *Lampung dataset (handwritten characters)* yang dikelola oleh *Embedded Systems Group* pada bagian *datasets* oleh Junaidi 2011. Dengan jumlah data yang digunakan sebanyak 32.140 data citra Aksara Lampung yang terbagi dalam 18 kelas. *Dataset* berformat .PGM. *Dataset* tersebut disimpan dalam google drive untuk memudahkan peneliti dalam penggunaan data ke google colab.

### 3.3.2 *Separasi Data*

Separasi Data yang dilakukan dengan membagi *dataset* menjadi 3 bagian yaitu *data training*, *data validation* dan *data testing*.

#### 3.3.2.1 *Data Training*

*Data Training* adalah data utama yang digunakan untuk melatih atau membangun model. Dengan menggunakan *data training* yang cukup dan representatif, model akan memiliki kemampuan untuk menggeneralisasi dan menerapkan pemahaman yang diperoleh dari data tersebut untuk melakukan prediksi atau klasifikasi pada data yang belum pernah dilihat sebelumnya. *Data training* yang digunakan oleh peneliti pada model ini sebesar 80% dari keseluruhan data citra.

### 3.3.2.2 *Data Validation*

*Data Validation* adalah data yang digunakan untuk proses validasi model dan mencegah *overfitting*. *Overfitting* terjadi ketika model terlalu kompleks dan terlalu memperhatikan detail pada *data training*, sehingga model tidak mampu melakukan prediksi dengan akurasi yang baik pada data baru. *Data validation* yang digunakan oleh peneliti pada model ini sebesar 10% dari keseluruhan data citra.

### 3.3.2.3 *Data Testing*

*Data Testing* adalah kumpulan data yang digunakan untuk mengevaluasi kinerja model setelah proses pelatihan selesai dan model telah diuji dengan *data validation*. Tujuan utama dari *data testing* adalah untuk menilai kinerja akhir model dengan menggunakan data yang benar-benar baru dan tidak pernah digunakan sebelumnya. *Data testing* yang digunakan oleh peneliti pada model ini sebesar 10% dari keseluruhan data citra.

### 3.3.3 **Pra Pemrosesan Data**

Pra pemrosesan Data merupakan teknik untuk menyiapkan data agar lebih siap untuk digunakan lebih lanjut yang digunakan untuk mentransformasi gambar asli ke dalam format yang dapat dijadikan input pada tahap selanjutnya. Pada tahap ini digunakan untuk mengubah data mentah dalam format yang berguna dan efisien. Dalam penelitian ini menggunakan *tensorflow* sehingga format yang dapat dikenali, yaitu *.jpeg*, *.png*, *.gif*, dan *.bmp*. Data yang digunakan berformat *.pgm* akan dikonversi menjadi salah satu format yang dapat dikenali *tensorflow*, yaitu *.bmp*, format ini

menggunakan metode kompresi tak terkompresi (*uncompressed*) yang memungkinkan gambar memiliki kualitas yang baik dengan detail yang lebih tinggi dan pemakaian memori yang lebih kecil. Pada data ini memiliki nilai intensitas piksel, yaitu RGB. Kemudian dilakukan augmentasi pada *data training* dan penyesuaian reduksi atau *resize* pada setiap subset-subset data yang telah di *split* sebelumnya untuk menyamakan ukuran dan rasio tiap data yang selanjutnya dilakukan augmentasi data pada *data training*. Dan yang terakhir dilakukan *rescale* pada tiap data untuk mengubah skala citra untuk memudahkan *neural network* memproses data.

### 3.3.4 Pelatihan Model

Tahapan pelatihan model menggunakan dua arsitektur *LeNet* dan *AlexNet*. Kedua arsitektur tersebut digunakan untuk melakukan perbandingan akurasi yang dihasilkan keduanya. Dalam tahapan ini menggunakan beberapa *hyperparameter* seperti *optimizer*, *epoch*, *batch-size* dan *learning-rate* sebagai salah satu faktor keberhasilan model saat pelatihan maupun pengujian. Berikut nilai *hyperparameter* yang digunakan ditunjukkan pada Tabel 4.

Tabel 4. *Hyperparameter*.

<i>Hyperparameter</i>	Nilai
<i>Epoch</i>	20
<i>Batch-size</i>	32
<i>Optimizer</i>	<i>Adam</i>
<i>Learning-rate</i>	0,001



### 3.3.5 Pengujian Model *LeNet* dan *AlexNet*

Tahapan pengujian model dilakukan terhadap *data training* dan *data validation* setelah dilakukannya pelatihan pada model sebelumnya. Dimana pada tahap ini *data testing* akan digunakan untuk menunjukkan keberhasilan model yang telah dilatih sebelumnya. Hasil akurasi terhadap dua arsitektur dalam tahapan ini kemudian dilakukan perbandingan.

### 3.3.6 Evaluasi Hasil (*Classification Report*)

Tahapan evaluasi hasil pengujian ini dilakukan pencatatan hasil dari pengujian untuk menghitung tingkat keberhasilan kedua model pada *data testing* dengan *classification report* yang berisi metrik evaluasi dari *confusion matrix* yang menunjukkan jumlah prediksi benar dan salah untuk setiap kelas. Dalam tahapan ini juga dapat terlihat kecenderungan objek terhadap sebuah kelas. Tingkat keberhasilan dan performa model dalam klasifikasi *multiclass* sebuah objek dilihat dan dianalisis melalui nilai-nilai dalam *confusion matrix* yaitu *accuracy* (Persamaan (1)), *recall* (Persamaan (2)), *precision* (Persamaan (3)) dan *f1-score* (Persamaan (4)).

### 3.3.7 Penarikan Kesimpulan

Penarikan Kesimpulan merupakan tahap akhir dalam penelitian ini. Disimpulkan tingkat akurasi kedua arsitektur, yaitu arsitektur *LeNet* dan *AlexNet*, yang memiliki performa lebih baik dalam klasifikasi citra Aksara Lampung.

## V. SIMPULAN DAN SARAN

### 5.1 Simpulan

Berdasarkan hasil dan pembahasan mengenai klasifikasi karakter citra Aksara Lampung dengan membandingkan dua arsitektur, yaitu *LeNet* dan *AlexNet*, dapat disimpulkan hasil penelitian sebagai berikut:

1. *Dataset* citra Aksara Lampung yang digunakan memiliki distribusi per kelas yang tidak seimbang, sehingga perlu dilakukan teknik yang dapat membuat data menjadi seimbang agar hasil dari klasifikasi lebih optimal.
2. Hasil *accuracy* terbaik pada saat *training* diperoleh dengan menggunakan arsitektur *LeNet* dibandingkan dengan arsitektur *AlexNet* disetiap skema, dengan tingkat akurasi mendekati nilai sempurna, yaitu 100% untuk arsitektur *LeNet* disetiap skema, sedangkan *AlexNet* memperoleh tingkat akurasi terbesar hanya pada skema 4, yaitu sebesar 94,27%.
3. Jumlah *trainable parameter* arsitektur *AlexNet* lebih banyak dibandingkan arsitektur *LeNet*. Namun, tingkat akurasi yang dihasilkan arsitektur *LeNet* lebih besar yang membuktikan efisiensi kinerja arsitektur *LeNet* dengan jumlah *trainable* yang lebih sedikit.
4. Penggunaan *callback ReduceLROnPlateau* sangat efektif untuk memonitor *arguments* yang tidak mengalami perubahan sehingga model tidak terjebak pada suatu *plateau*.
5. Terdeteksi terdapat kesalahan penempatan label kelas pada *dataset* yang digunakan.

6. Hasil klasifikasi data pada *false negative* terbanyak pada tiap arsitektur terjadi pada aksara "a" dengan aksara "la" dengan arsitektur *AlexNet* setiap skema dan *false negative* terbanyak pada tiap arsitektur terjadi pada aksara "a" dengan aksara "la" pada skema 1, aksara "ka" dengan aksara "a" pada skema 2 dan 3, aksara "ga" dengan aksara "sa" pada skema 4 dengan arsitektur *LeNet*. Perbandingan antar aksara tersebut memiliki bentuk perbedaan yang signifikan.

## 5.2 Saran

Berikut saran yang diberikan untuk penelitian selanjutnya sebagai berikut.

1. Menggunakan berbagai arsitektur CNN lainnya seperti *ResNet-50*, *VGG-16*, *EfficientNetB4*, *Squeezenet*, *GoogleNet* dan arsitektur lain yang memiliki variasi sktruktur dan kedalaman *layer* yang beragam.
2. Melakukan variasi nilai *hyperparameter* yaitu *learning rate*, *batch size*, *optimizer*, dan *epoch* yang lebih beragam dari penelitian ini.
3. Mengembangkan penelitian ini kedalam sebuah aplikasi *mobile* ataupun *website*.
4. Dilakukan *screening* ulang secara menyeluruh pada *dataset* untuk memperbaiki penempatan data pada label yang tepat.
5. Mencari model yang lebih sederhana karena input ukuran datanya tidak besar.

## DAFTAR PUSTAKA

- Alamgunawan, S., & Kristian, Y. (2020). Klasifikasi Tekstur Serat Kayu pada Citra Mikroskopik Veneer Memanfaatkan Deep Convolutional Neural Network. *INSYST: Journal of Intelligent System and Computation*. 2(1): 06-11.
- Alwanda, M. R., Ramadhan, R. P. K., & Alamsyah, D. (2020). Implementasi metode convolutional neural network menggunakan arsitektur LeNet-5 untuk pengenalan doodle. *Jurnal Algoritme*. 1(1): 45-56.
- Al Rivan, M. E., & Riyadi, A. G. (2021). Perbandingan arsitektur lenet dan alexnet pada metode convolutional neural network untuk pengenalan american sign language. *Jurnal Komputer Terapan*, 7(1), 53-61.
- Ashqar, B. A., & Abu-Naser, S. S. (2018). Image-based tomato leaves diseases detection using deep learning. *International Journal of Academic Engineering Research (IJAER)*. 2(1): 10-16.
- Assegaf, M. R., & Wibowo, A. T. (2021). Klasifikasi Spesies Tanaman Monstera Berdasarkan Citra Daun Menggunakan Metode Convolutional Neural Network (Cnn). *eProceedings of Engineering*. 8(4).
- Brownlee, J. "What is the Difference Between a Batch and an Epoch in a Neural Network?," *Machine Learning Mastery*, 5 August 2022. [Online]. Available: <https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/>. [Accessed 10 August 2022].
- Budiman, A., Wijaya, P. W., Nur, M. H. A., Halawa, M. V., & Susyanti, S. (2019). Revisualisasi aksara Kaganga sebagai identitas Lampung. *Jurnal Bahasa Rupa*. 2(2): 118-125.
- Dewi, R., & Muslihah, N. N. (2022). Workshop Penulisan Aksara Ulu pada Guru dan Siswa Se-Kecamatan Sukakarya Kabupaten Musi Rawas. *PKM Linggau: Jurnal Pengabdian dan Pemberdayaan Masyarakat*. 2(1): 56-68.

- Diantika, S., Nalatissifa, H., Supriyadi, R., Maulidah, N., & Fauzi, A. (2023). Implementation Of Multi-Class Gradient Boosting To Classify Animal Species In Zoos. *Antivirus: Jurnal Ilmiah Teknik Informatika*, 17(1), 33-40.
- Emanuella, C. T., Musfita, M., & Lawi, A. (2021). Klasifikasi Suara Kucing dan Anjing Menggunakan Convolutional Neural Network. *Proceeding KONIK (Konferensi Nasional Ilmu Komputer)*. 5: 321-327.
- Fawwaz, M. A. A., Ramadhani, K. N., & Sthevanie, F. (2021). Klasifikasi Ras pada Kucing menggunakan Algoritma Convolutional Neural Network (CNN). *eProceedings of Engineering*. 8(1).
- Hakim, L., Rahmanto, H. R., Kristanto, S. P., & Yusuf, D. (2023). Klasifikasi Citra Motif Batik Banyuwangi Menggunakan Convolutional Neural Network. *Jurnal Teknoinfo*. 17(1): 203-211.
- Hendriyana, H., & Maulana, Y. H. (2020). Identification of Types of Wood using Convolutional Neural Network with Mobilenet Architecture. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*. 4(1): 70-76.
- Ilahiyah, S., & Nilogiri, A. (2018). Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network. *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*. 3(2): 49-56.
- Indolia, S., Goswami, A. K., Mishra, S. P., & Asopa, P. (2018). Conceptual understanding of convolutional neural network-a deep learning approach. *Procedia computer science*. 132: 679-688.
- Indrayati, R. I., & Migotuwio, N. (2020). Identifikasi Anatomi Aksara Lampung. *AKSA: Jurnal Desain Komunikasi Visual*, 4(1), 541-551.
- Irfansyah, D., Mustikasari, M., & Suroso, A. (2021). Arsitektur Convolutional Neural Network (CNN) Alexnet Untuk Klasifikasi Hama Pada Citra Daun Tanaman Kopi. *Jurnal Informatika*. 6(2).
- James, A., Manjusha, J., & Saravanan, C. (2018). Malayalam handwritten character recognition using AlexNet based architecture. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, 6(4), 393-400.
- Junaidi, A., Vajda, S., & Fink, G. A. (2011). Lampung-a new handwritten character benchmark: Database, labeling and recognition. In *Proceedings of the 2011 Joint Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data* (pp. 1-8).

- Lorentius, C. A., Gunadi, K., & Tjondrowiguno, A. N. (2019). Pengenalan aksara jawa dengan menggunakan metode convolutional neural network. *Jurnal Infra*. 7(1): 221-227.
- Mahmud, K. H., Adiwijaya, A., & Al Faraby, S. (2019). Klasifikasi Citra Multi-Kelas Menggunakan Convolutional Neural Network. *eProceedings of Engineering*. 6(1).
- Nugraha, W., & Sasongko, A. (2022). Hyperparameter Tuning on Classification Algorithm with Grid Search. *SISTEMASI*. 11(2): 391-401.
- Paliwang, A. A. A., Septian, M. R. D., Cahyanti, M., & Swedia, E. R. (2020). Klasifikasi Penyakit Tanaman Apel Dari Citra Daun Dengan Convolutional Neural Network. *Sebatik*. 24(2): 207-212.
- Pius, N. K., & Johny, A. (2020). Malayalam handwritten character recognition system using convolutional neural network. *International Journal of Applied Engineering Research*. 15(9): 918-920.
- Pratiwi, H. A., Cahyanti, M., & Lamsani, M. (2021). Implementasi Deep Learning Flower Scanner Menggunakan Metode Convolutional Neural Network. *Sebatik*. 25(1): 124-130.
- Ridwan, R., Lubis, H., & Kustanto, P. (2020). Implementasi algoritma neural network dalam memprediksi tingkat kelulusan mahasiswa. *Jurnal Media Informatika Budidarma*. 4(2): 286-293.
- Rikendry, R., & Maharil, A. (2022). Perbandingan Arsitektur VGG-16 Dan ResNet-50 Untuk Rekognisi Tulisan Tangan Aksara Lampung. *Jurnal Informatika dan Rekayasa Perangkat Lunak*, 3(2), 236-243.
- Rochmawati, N., Hidayati, H. B., Yamasari, Y., Tjahyaningtijas, H. P. A., Yustanti, W., & Prihanto, A. (2021). Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam. *Journal Information Engineering and Educational Technology*. 5(2): 44-48.
- Roza, E. (2017). Aksara Arab-Melayu di Nusantara dan Sumbangsihnya dalam Pengembangan Khazanah Intelektual. *Tsaqafah*. 13(1): 177-204.
- Rustam, F., Siddique, M. A., Siddiqui, H. U. R., Ullah, S., Mehmood, A., Ashraf, I., & Choi, G. S. (2021). Wireless capsule endoscopy bleeding images classification using CNN based model. *IEEE Access*. 9: 33675-33688.
- Sarwono, Sarwit., & Rahayu, Ngudining., (2014). Pusat Penulisan dan Para Penulis Manuskrip Ulu di Bengkulu. Universitas Bengkulu : UNIB Press.

- Suhardin, I., Patombongi, A., & Islah, A. M. (2021). Mengidentifikasi Jenis Tanaman Berdasarkan Citra Daun Menggunakan Algoritma Convolutional Neural Network. *Simtek: jurnal sistem informasi dan teknik komputer*, 6(2): 100-108.
- Zhao, H. H., & Liu, H. (2020). Multiple classifiers fusion and CNN feature extraction for handwritten digits recognition. *Granular Computing*, 5: 411-418.
- Zufar, M., & Setiyono, B. (2016). Convolutional neural networks untuk pengenalan wajah secara real-time. *Jurnal Sains dan Seni ITS*, 5(2), 128862.
- Zuhri, A. B., Maulana, D. I., & Maheswara, E. S. (2022). Optimization Image Classification Pada Ikan Hiu Dengan Metode Convolutional Neural Network Dan Data Augmentasi. *Jurnal Tika*, 7(1), 1-11.