

Perbandingan *Kernel Convolutional Neural Network* dalam Pengenalan dan Transliterasi Kata Aksara Lampung

<http://dx.doi.org/10.28932/jutisi.v11i2.10406>

Riwayat Artikel

Received: 28 November 2024 | Final Revision: 29 Juli 2025 | Accepted: 29 Juli 2025



Creative Commons License 4.0 (CC BY – NC)

Desi Rahma Utami^{✉ #1}, Umi Murdika^{#2}

[#] *Program Studi Teknik Elektro, Universitas Lampung*

Jalan Prof. Dr. Ir. Sumantri Brojonegoro No.1, Kota Bandar Lampung, Lampung 35141, Indonesia

¹deasyrahma676@gmail.com

²umi.murdika@eng.unila.ac.id

[✉]Corresponding author: deasyrahma676@gmail.com

Abstrak — Penelitian bertujuan untuk membuat suatu sistem yang dapat mengenali dan mentransliterasi data citra aksara Lampung dan melakukan perbandingan *kernel Convolutional Neural Network* (CNN) terhadap sistem pengenalan dan transliterasi kata aksara Lampung. Sistem pengenalan dan transliterasi aksara Lampung dengan model pembelajaran CNN diaplikasikan menggunakan bahasa pemrograman python 3.12.5 64 bit, dengan *stride* sebesar 1 untuk konvolusi dan 2 untuk pooling, variasi ukuran kernel yang digunakan adalah 2x2, 3x3 dan 5x5 yang diaplikasikan secara bersilang untuk ekstraksi ciri proses konvolusi dan pooling. Tipe *kernel* konvolusi 3x3 dan *kernel* pooling 3x3 menunjukkan kinerja terbaik dalam melakukan transliterasi dan pengenalan kata aksara Lampung dengan akurasi pengujian sebesar 0.9 dan kesetimpangan data hasil pengujian yang kecil, yaitu 2/10 atau 0.2 . Ukuran Kernel 3x3 menunjukkan kondisi yang ideal untuk digunakan terutama ketika ciri citra yang digunakan memiliki perbedaan ciri yang sangat sedikit.

Kata kunci— aksara Lampung; CNN; ekstraksi ciri; *kernel*; python.

Comparison of *Kernel Convolutional Neural Network* in Lampung Script Word Recognition and Transliteration

Abstract — The study aims to create a system that can recognize and transliterate Lampung script image data and compare the Convolutional Neural Network (CNN) kernel to the Lampung script word recognition and transliteration system. The Lampung script recognition and transliteration system with the CNN learning model is implemented using the python 3.9.4 64 bit programming language, with a stride of 1 for convolution and 2 for pooling, the kernel size variations used are 2x2, 3x3 and 5x5 which are applied crosswise for feature extraction of the convolution and pooling processes. The 3x3 convolution kernel type and 3x3 pooling kernel showed the best performance in transliterating and recognizing Lampung script words with a test accuracy of 0.9 and a small test result data inequality, which is 2/10 or 0.2. The 3x3 Kernel Size shows ideal conditions for use, especially when the image features used have very few differences in features.

Keywords—aksara Lampung; CNN; feature extraction; kernel; python.

I. PENDAHULUAN

Era digital memiliki potensi kebermanfaatan juga kerugian dalam setiap proses pembangunan. Pengaruh kemajuan dan kemunduran serta bermanfaat atau tidaknya era digital dalam kehidupan berbangsa dan bernegara tergantung pada bagaimana mengoptimalkan dan menjaganya, serta melestarikan dan mengembangkan teknologi, tanpa perlu mengurangi atau bahkan menghilangkan nilai-nilai luhur dan jati diri bangsa (Kebudayaan). Salah satu fokus keilmuan yang dinilai sangat efektif dalam mengembangkan kebudayaan di era digital saat ini adalah pengolahan dan pengoptimalan teknologi informasi. Dalam bidang penerapan teknologi dan informasi, tantangan nyata di era digital semakin kompleks, seperti memetakan dan mengatasi permasalahan yang ada, meningkatkan taraf efektivitas, merancang kreativitas, inovasi, dan membangun efisiensi dalam melaksanakan pekerjaan [1]. Dalam bidang sosial budaya, salah satu pengaruh teknologi di era digital terutama bidang teknologi informasi saat ini ialah menyebabkan terjadinya perubahan pola interaksi antar masyarakat terutama pada masyarakat ekonomi menengah ke atas terutama dalam hal penggunaan bahasa daerah dan aksara yang dianggap kuno dan tertinggal [1]. Salah satu wilayah di Indonesia yang memiliki aksara adalah provinsi Lampung. Aksara Lampung sering disebut oleh suku asli Lampung dengan sebutan 'Surat Lampung' (Had Lampung). Aksara Lampung terdiri atas tiga bagian yaitu; Huruf induk (rangkaian huruf), tanda bunyi/huruf anak (rangkaian huruf), dan tanda baca. Aksara Lampung memiliki 20 huruf induk dan 12 huruf anak [2],[3].

Menyikapi hal tersebut, perlu adanya perhatian khusus terhadap perkembangan teknologi informasi dan sosial budaya. Dinilai perlu untuk menyediakan suatu teknologi informasi yang mampu menghilangkan stigma kuno dan terbelakang tanpa menghilangkan nilai-nilai luhur suatu budaya, khususnya bahasa dan aksara yang menjadi alat komunikasi suatu bangsa. Salah satu fokus keilmuan yang dinilai mampu melakukan hal tersebut adalah penerapan teknologi pengolahan citra dan metode pembelajaran yang mampu melakukan proses pengenalan dan transliterasi data (citra aksara Lampung). Penelitian bertujuan untuk membuat suatu sistem yang dapat mengenali dan mentransliterasi data citra aksara Lampung dan melakukan perbandingan kernel *Convolutional Neural Network* (CNN) terhadap sistem pengenalan dan transliterasi kata aksara Lampung.

Perbedaan penelitian ini terhadap penelitian sebelumnya [4],[5],[6] dalam membangun sistem pengenalan dan transliterasi aksara Lampung, yaitu berupa data citra aksara Lampung dalam bentuk kata, sebagai berikut; Sazqiah,N. dkk [4], dalam upaya mengenali aksara Lampung menggunakan metode *Convolutional Neural Network* (CNN) yang menghasilkan akurasi sebesar 100%. Silalahi, dkk [5], dalam upaya mengimplementasikan *Convolutional 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 [6], dalam upaya mengenali aksara Lampung menggunakan penerapan Optical Character Recognition (OCR) dengan metode *Convolutional Neural Network* (CNN) yang menghasilkan tingkat klasifikasi dan pengenalan karakter (segmentasi) yang lebih akurat.

Selain itu, perbedaan penelitian ini terhadap penelitian sebelumnya [7],[8] dalam melakukan perbandingan kernel terletak pada fokus perbandingan, yaitu penelitian ini fokus dalam melakukan perbandingan hanya pada tiga jenis resolusi kernel (2x2, 3x3, dan 5x5) terhadap masing-masing proses konvolusi dan *pooling* dengan parameter atribut CNN dan metrik pembangunan sistem yang sama. Sementara penelitian sebelumnya [7],[8] berfokus pada perbandingan parameter atribut CNN dan metrik pembangunan sistem yang berbeda. Nanang Kasim dan Gibran Satya Nugraha [7], dalam upaya melakukan perbandingan atribut CNN dan metrik pembangunan sistem pada pengenalan pola tulisan aksara Arab menghasilkan hasil terbaik dengan *kernel_size* 5x5, ukuran citra 64x64, jumlah bobot dropout sebesar 30%, jumlah epoch sebanyak 80 dan pembagian dataset citra pembangunan sistem sebesar 70:30. Muhammad Alham Aqli Dilan [8], dalam upaya melakukan perbandingan atribut CNN dan metrik pembangunan sistem pada klasifikasi citra jenis sepatu lari berdasarkan lintasan menghasilkan hasil terbaik dengan *kernel_size* 2x2, ukuran citra 128x128, dan jumlah epoch sebanyak 80.

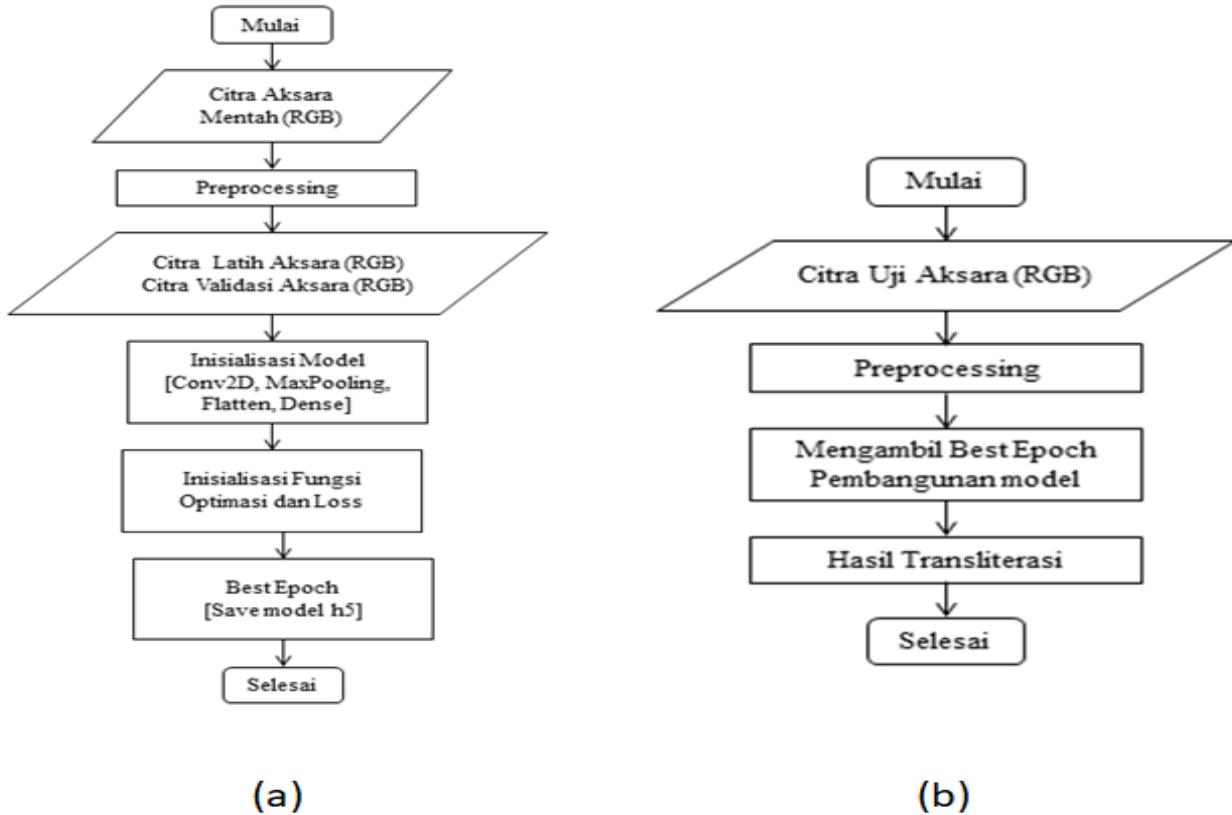
Penelitian ini menentukan kernel terbaik menggunakan metode perbandingan terhadap hasil keluaran citra, hasil pengujian sistem dan selisih hasil pengujian sistem. Penelitian diharapkan dapat berkembang dan menghasilkan aplikasi lanjutan yang terintegrasi kedalam perangkat digital ataupun perangkat elektronik, serta mampu mengenali dan mentransliterasi tulisan aksara lampung dalam klasifikasi yang lebih banyak lagi.

II. METODE

Penelitian ini menggunakan code editor *Visual Studio Code*, Bahasa pemrograman *python 3.12.5 64 bit*, operasi sistem *windows 11*, serta *Library external python* berupa *tensorflow 2.17.0*, *keras 3.5.0*, *numpy 1.26.4*, *matplotlib 3.9.2*, *open-cv 4.10.0.84*, dan *scikit-learn 1.5.1*.

A. Diagram Alir Perancangan Sistem

Diagram Alir Perancangan Sistem merupakan gambaran tahapan-tahapan yang akan dilakukan oleh sistem dalam menjalankan fungsinya untuk mencapai tujuan pengenalan dan transliterasi aksara Lampung. Diagram alir sistem perancangan dibagi menjadi dua proses, yaitu proses pengembangan sistem (pelatihan dan validasi) dan proses pengujian sistem yang ditunjukkan pada Gambar 1.



Gambar 1 Diagram Alir (a) Pembangunan (pelatihan dan validasi) Sistem (b) Pengujian Sistem

B. Akuisisi Data

Tabel 1 menunjukkan dua jenis *dataset* yang digunakan dalam perancangan sistem, yaitu *dataset* citra untuk pengembangan sistem dan *dataset* citra untuk pengujian sistem. *Dataset* citra pengembangan sistem [9][10][11][12][13] berfungsi untuk mengembangkan sistem melalui proses pengujian dan validasi, melakukan perancangan berkali-kali hingga menemukan sistem yang dapat belajar dan memvalidasi *dataset* yang telah dipersiapkan sebelumnya dengan hasil yang minim kesalahan. *Dataset* citra pengujian sistem merupakan kumpulan data baru dan tidak sama dengan *dataset* citra pembangunan, berfungsi untuk melihat seberapa akurat dan tepat sistem dalam melakukan pengenalan dan pengujian. Gambar 2 menunjukkan representasi masing-masing citra yang berupa tulisan tangan empat jenis kata, yaitu: Abadi, Abu, Acara, dan Ada.



Gambar 2 Citra Mentah Aksara Kelas (a) Abadi (b) Abu (c) Acara (d) Ada

TABEL 1
PROSES AKUISISI DATA

No	Jenis Citra	Fungsi Citra	Total Jumlah	Jumlah data /kelas	Proses Akuisisi
1	Pembangunan sistem	Dasar	1692	423	Ditulis dengan variasi warna serta media tulis yang berbeda. Diambil menggunakan kamera ponsel.
		Latih	1352	338	80% dari keseluruhan data dasar. Diolah menggunakan Bahasa pemrograman Python [dipilih secara acak menggunakan fungsi random, pengulangan, dan copyfile (seperti pada gambar 3)]
2	Pengujian sistem	Validasi	336	84	20% dari keseluruhan data dasar. Diolah menggunakan bahasa pemrograman Python [dipilih secara acak menggunakan fungsi random, pengulangan, dan copyfile (seperti pada gambar 4)]
		Uji	40	10	Ditulis dengan variasi tulisan tangan berbeda. Diambil menggunakan kamera ponsel.

```
# 01. Data Splitting
# Data Latih dan validasi kelas 01(Abadi)
latih_rasio = 0.8 #Rasio pembagian dataset citra latih

# Membaca jumlah dataset citra mentah kelas 01 'Abadi'
total_size01 = len(os.listdir(data01))
# Melakukan pembagian jumlah data citra mentah untuk citra latih [0.8 x 423] kelas 01 'Abadi'
latih_size01 = int(latih_rasio*total_size01)
# Melakukan pembagian jumlah data citra mentah untuk citra validasi [423 x (1-0.8)] kelas 01 'Abadi'
val_size01 = int(total_size01*(1-latih_rasio))

# Melakukan pengambilan secara acak data citra dari keseluruhan data citra mentah kelas 01 'Abadi'
randomized01 = random.sample (os.listdir(data01), total_size01 )

# Melakukan pengambilan acak data citra mentah untuk citra latih kelas 01 'Abadi'
latih01 = randomized01[0:latih_size01]
# Pengulangan perintah pengambilan dan pembagian data citra latih kelas 01 'Abadi'
for i in latih01 :
    # Menerapkan perintah pengulangan i, pengambilan data/eksplorasi (membaca + mengambil),
    # data citra mentah kelas 01 'Abadi'
    A1_X = data01 + i
    # Menerapkan perintah pengulangan i, pembagian data/eksplorasi (membaca + mengisi),
    # data citra mentah latih 01 'Abadi'
    A1_Y = data_latih01 + i
    # Melakukan penduplikasian data citra mentah menjadi data citra latih kelas 01 'Abadi',
    # sesuai dengan ketentuan pembagian jumlah data
    copyfile(A1_X, A1_Y)
```

Gambar 3 Perintah Proses Akuisisi Dataset Latih

```
# Melakukan pengambilan secara acak data citra dari keseluruhan data citra mentah kelas 01 'Abadi'
randomized01 = random.sample (os.listdir(data01), total_size01 )

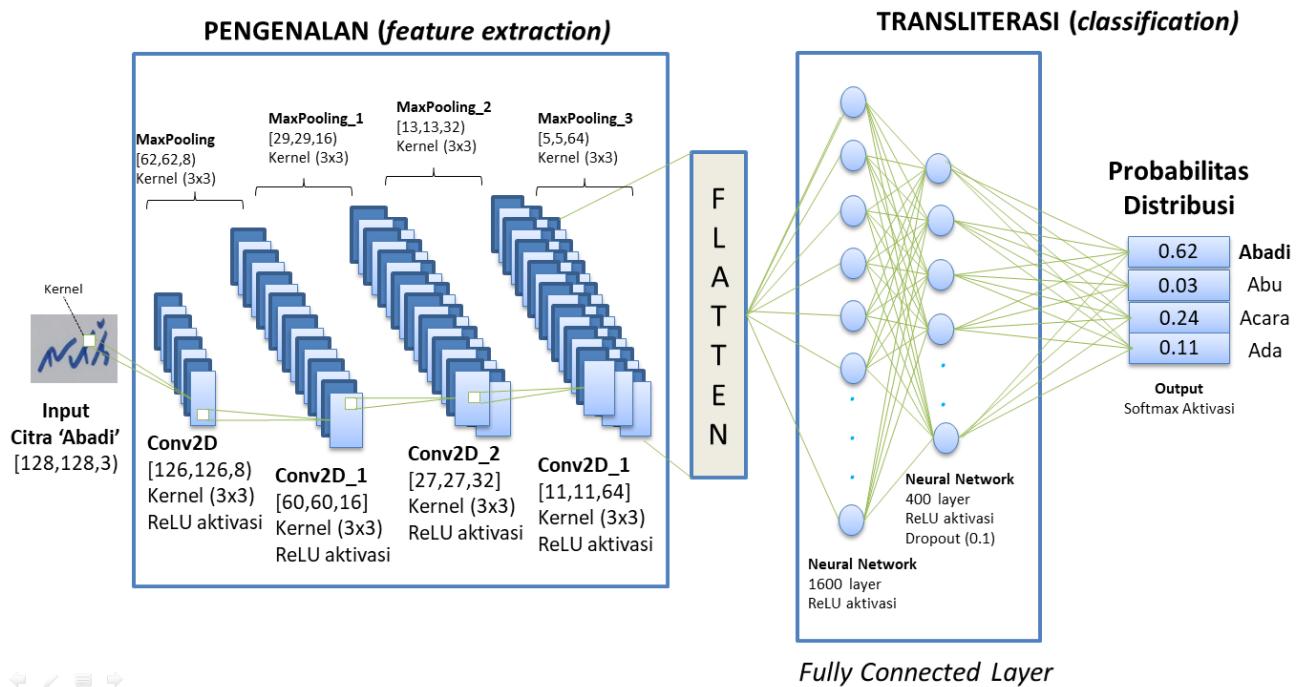
# Melakukan pembagian jumlah data citra mentah untuk citra validasi [423 x (1-0.8)] kelas 01 'Abadi'
val_size01 = int(total_size01*(1-latih_rasio))

# Melakukan pengambilan acak data citra mentah untuk citra validasi kelas 01 'Abadi'
val01 = randomized01[0:val_size01]
# Pengulangan perintah pengambilan dan pembagian data citra validasi kelas 01 'Abadi'
for j in val01 :
    # Menerapkan perintah pengulangan j, pengambilan data/eksplorasi (membaca + mengambil),
    # data citra mentah kelas 01 'Abadi'
    B1_X = data01 + j
    # Menerapkan perintah pengulangan j, pembagian data/eksplorasi (membaca + mengisi),
    # data citra mentah validasi 01 'Abadi'
    B1_Y = data_val01 + j
# Melakukan penduplikasian data citra mentah menjadi data citra validasi kelas 01 'Abadi'
copyfile(B1_X, B1_Y)
```

Gambar 4 Perintah Proses Akuisisi Dataset Validasi

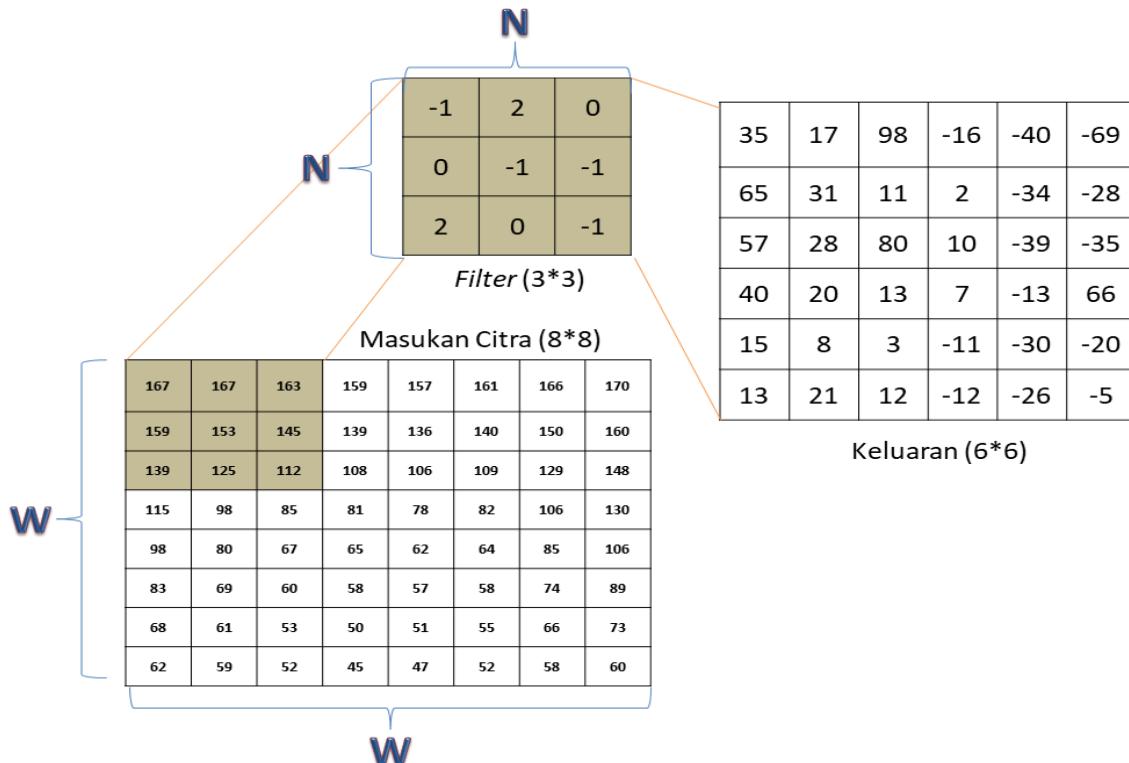
C. Convolutional Neural Network (CNN)

Tahap ini merupakan tahap perancangan perintah-perintah dalam sistem pengenalan dan transliterasi aksara Lampung baik pada proses pembangunan maupun pengujian. Terdapat dua tahap dalam pembuatan model arsitektur CNN yaitu tahap pengenalan (ekstraksi ciri) dan tahap transliterasi (klasifikasi) [14], seperti direpresentasikan pada Gambar 5.



Gambar 5 Representasi Model CNN [5]

Secara matematis perhitungan keluaran citra hasil pengenalan (ekstraksi ciri) pada sistem pembelajaran CNN yang ditunjukkan dalam persamaan (1) merupakan hasil komputasi pengaplikasian filter terhadap sebagian piksel gambar dengan pergeseran tertentu, seperti yang ditunjukkan oleh Gambar 6.



Gambar 6 Representasi Perhitungan proses pengenalan (ekstraksi ciri) citra.

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

Dimana :

O adalah *output* / keluaran

W adalah panjang/tinggi masukan

N adalah panjang/tinggi filter(*kernel_size*)

P adalah *padding*, biasanya difungsikan supaya ukuran piksel gambar tidak berubah

S adalah *stride*, seberapa banyak pergeseran

Berdasarkan Gambar 6, terlihat bahwa representasi citra masukan yang digunakan berupa citra berdimensi (WxW) 8 x 8, filter citra (NxN) 3x3, dengan asumsi stride dan padding yang digunakan adalah 1 dan 0 (zero padding). Maka, keluaran citra yang dihasilkan (O) dapat dihitung menggunakan persamaan (1), menjadi ;

$$O = \frac{8-3+2(0)}{1} + 1 = 6$$

Dikarenakan citra masukan berupa matriks persegi, maka perhitungan diatas berlaku untuk kedua sisi, panjang dan juga tinggi. Sehingga, citra keluaran juga berupa matriks persegi berdimensi 6x6.

Pada tahap pengenalan (ekstraksi ciri) [15][16][17], proses yang akan dilakukan adalah menentukan jumlah proses konvolusi, jenis pooling pada masing-masing konvolusi, jumlah filter pada masing-masing konvolusi dan pooling, ukuran kernel/filter pada masing-masing konvolusi, ukuran strides pada masing-masing konvolusi dan pooling, dan aktivasi fungsi pada masing-masing konvolusi.

1. *Layer Konvolusi* : lapisan yang memiliki k filter atau kernel. Filter tersebut memiliki dimensi n×n. n (baris/kolom) seringkali lebih kecil dari ukuran dimensi citra input.

2. *Layer Rectifier Linear Unit (ReLU)* : lapisan tambahan yang merupakan fungsi matematika, yang memungkinkan pelatihan lebih cepat dan efektif dengan memetakan nilai negatif menjadi nol dan mempertahankan nilai positif.
3. *Layer Pooling* : mengurangi ukuran spasial dan jumlah parameter dalam jaringan serta mempercepat komputasi dan kontrol *overfitting*. Ada dua jenis *pooling* yang sering digunakan, yaitu *Max Pooling* dan *Average Pooling*.

Pada tahap transliterasi (Klasifikasi) proses diawali dengan proses *flattening*, kemudian menentukan jumlah proses klasifikasi, jumlah neural (node), fungsi aktivasi dan nilai *dropout* [14],[15],[16],[17].

D. Pembangunan Sistem

TABEL 2
VARIASI KERNEL_SIZE

Kode Tipe	Kernel Size		Save model.h5
	Konvolusi	Pooling	
1A		2X2	Uji_Kernel1A.weights.h5
1B	2X2	3X3	Uji_Kernel1B.weights.h5
1C		5X5	Uji_Kernel1C.weights.h5
2A		2X2	Uji_Kernel2A.weights.h5
2B	3X3	3X3	Uji_Kernel2B.weights.h5
2C		5X5	Uji_Kernel2C.weights.h5
3A		2X2	Uji_Kernel3A.weights.h5
2B	5X5	3X3	Uji_Kernel3B.weights.h5
3C		5X5	Uji_Kernel3C.weights.h5

Tabel 2 menunjukkan bahwa pembangunan sistem dilakukan sebanyak 9 kali dengan variasi *kernel_size* proses konvolusi dan *pooling* yang berbeda. Masing-masing variasi *kernel_size* diterapkan ke dalam sistem pembelajaran dengan parameter atribut CNN dan metrik pembangunan sistem yang sama. Parameter atribut CNN [17] yang diterapkan;

1. *Masukan citra* : RGB 128x128.
2. *Jenis Pooling* : *Max_Pooling*
3. *Layer Convolutional*: 4 Layer dengan *channel* bertingkat bernilai 8, 16, 32 dan 64.
4. *Layer Fully Connected* : 3 Layer *Fully Connected* dengan *channel* menurun bernilai 1600, 400, 4
5. *Stride* : 1 untuk konvolusi dan 2 untuk *pooling*.
6. *Dropout* : sebesar 0.01.

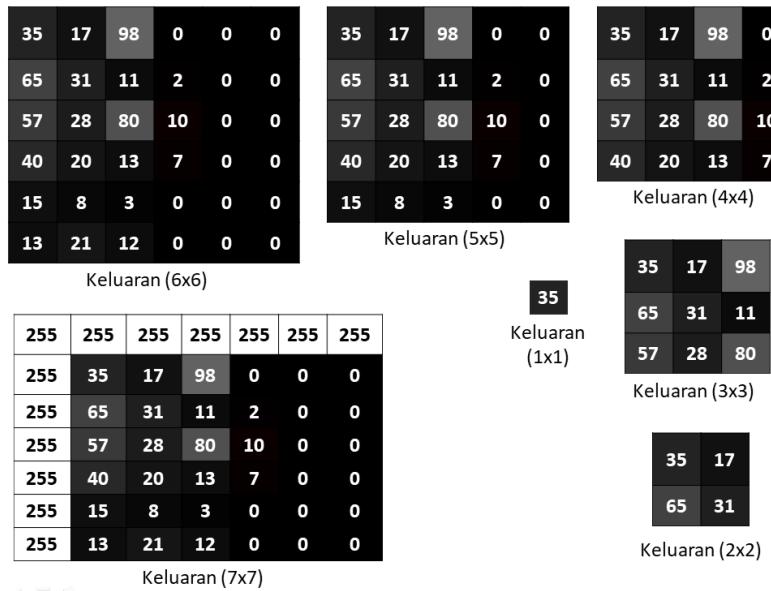
Parameter metrik pembangunan sistem yang diterapkan berupa fungsi *optimizer RMSProp* [16] dan *Epoch* sebanyak 20 kali. *Epoch* hasil pembangunan sistem kemudian disimpan dalam format ekstensi .h5, seperti diperlihatkan pada Tabel 2, untuk kemudian dapat digunakan kembali dalam tahap pengujian sistem.

E. Pengujian Sistem

Proses yang menjadi fokus dalam pengujian adalah seberapa akurat dan tepat sistem dalam melakukan pengenalan dan pengujian. Pada tahap ini proses yang akan dilakukan adalah mengambil data citra uji melalui direktori laptop. Setelah citra uji diambil, sistem akan mendekripsi dan menampilkan hasil transliterasi citra tersebut. Pengujian sistem dilakukan sebanyak 360 kali dengan masing-masing 40 kali pengujian untuk setiap *kernel_size*.

III. HASIL DAN PEMBAHASAN

Hasil dan pembahasan berfokus pada tiga poin perbandingan dalam menentukan kondisi terbaik penggunaan kernel, yaitu ; Hasil keluaran pengenalan (*features extraction*) CNN, Hasil pengujian Sistem, dan Selisih hasil pengujian data. Hasil keluaran pengenalan (*features extraction*) CNN, menentukan kondisi terbaik berdasarkan hasil keluaran ukuran piksel citra, yaitu tidak terlalu kecil dan tidak terlalu besar. Jika terlalu kecil, citra akan kehilangan banyak ciri/*features*, sehingga sistem mengalami kesulitan dalam menemukan ciri unik citra dan melakukan perbedaan terhadap citra lainnya. Jika terlalu besar, citra akan terlalu banyak ciri/*features*, sehingga sistem akan mengalami bias dan komputasi akan menjadi semakin berat dan lama, karena sistem harus mempelajari banyak ciri yang sebagian besar bukan ciri unik citra tersebut dan tidak seharusnya dikenali sistem. Gambar 7 menunjukkan perbandingan representasi piksel citra *grayscale* dari 1x1 hingga 7x7.



Gambar 7 Representasi piksel citra grayscale

Hasil pengujian Sistem, menentukan kondisi terbaik berdasarkan seberapa tepat dan akurat sistem dalam mengklasifikasi citra. Sementara selisih hasil pengujian data, menentukan kondisi terbaik berdasarkan ketimpangan pendistribusian data. Semakin kecil selisih hasil pengujian data, semakin baik sistem dalam mendistribusikan *dataset* citra pembangunan sistem, sehingga sistem melakukan pengujian dengan hasil keluaran yang akurat dan tepat di setiap masing-masing kategori/kelas *dataset* citra pengujian sistem. Semakin besar selisih hasil pengujian data, semakin buruk sistem dalam mendistribusikan *dataset* citra pembangunan sistem, sehingga sistem melakukan pengujian dengan hasil keluaran yang akurat dan tepat hanya di sebagian kategori/kelas *dataset* citra pengujian sistem. Umumnya, masalah pendistribusian *dataset* ini dapat diatasi menggunakan metode *cross validation*. Namun, dalam penelitian ini dikarenakan berfokus dalam melakukan perbandingan kernel, maka metode tersebut tidak digunakan dalam penelitian ini.

A. Features Extraction CNN

TABEL 3
NILAI PIKSEL KELUARAN CITRA PADA PROSES FEATURES EXTRACTION CNN

Tipe	Layer 1		Layer 2		Layer 3		Layer 4	
	Con2D	Pooling2D	Con2D	Pooling2D	Con2D	Pooling2D	Con2D	Pooling2D
1A	127x127	63x63	62x62	31x31	30x30	15x15	14x14	7x7
1B	127x127	63x63	62x62	30x30	29x29	14x14	13x13	6x6
1C	127x127	62x62	61x61	29x29	28x28	12x12	11x11	4x4
2A	126x126	63x63	61x61	30x30	28x28	14x14	12x12	6x6
2B	126x126	62x62	60x60	29x29	27x27	13x13	11x11	5x5
2C	126x126	61x61	59x59	28x28	26x26	11x11	9x9	3x3
3A	124x124	62x62	58x58	29x29	25x25	12x12	8x8	4x4
3B	124x124	61x61	57x57	28x28	24x24	11x11	7x7	3x3
3C	124x124	60x60	56x56	26x26	22x22	9x9	5x5	1x1

Tabel 3 merupakan hasil keluaran piksel citra proses *features extraction* sistem pembelajaran CNN dengan masukan awal piksel citra berukuran 128x128. Hasil keluaran dapat dihitung menggunakan persamaan (1). Berdasarkan Tabel 3 terlihat bahwa hasil akhir *features extraction*, diperlihatkan pada kolom paling kanan (Layer 4, *Pooling2D*), nilai piksel keluaran citra secara berurutan dari 1x1 hingga 7x7.

1. *Piksel 1x1*: hasil keluaran dari *kernel_size* tipe 3C.
2. *Piksel 3x3*: dihasilkan dari 2 tipe *kernel_size*, yaitu tipe 2C dan 3B.
3. *Piksel 4x4*: dihasilkan dari 2 tipe *kernel_size*, yaitu tipe 1C dan 3A.
4. *Piksel 5x5*: hasil keluaran dari *kernel_size* tipe 2B

5. *Piksel 6x6*: dihasilkan dari 2 tipe *kernel_size*, yaitu tipe 1B dan 2A
6. *Piksel 7x7*: hasil keluaran dari *kernel_size* tipe 1A

B. Kernel Konvolusi Tipe 1 [2x2]

TABEL 4
HASIL PENGUJIAN KERNEL KONVOLUSI 2X2

		Tipe 1A			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	10	0	0	0
	Abu	0	10	0	0
	Acara	6	1	3	0
	Ada	2	7	0	1
		Tipe 1B			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	8	2	0	0
	Abu	0	10	0	0
	Acara	0	2	8	0
	Ada	0	7	0	3
		Tipe 1C			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	5	4	1	0
	Abu	0	10	0	0
	Acara	3	4	3	0
	Ada	1	9	0	0

Tabel 4 merupakan hasil pengujian sistem pembelajaran terhadap ukuran *kernel_size* 2x2 untuk proses konvolusi dan *kernel_size* 2x2, 3x3 dan 5x5 untuk proses pooling. Hasil terbaik dan terburuk dalam pengujian;

1. *kernel_size type 1A*: Hasil terbaik dengan 10 benar dari keseluruhan 10 data uji, ditunjukan oleh dua kelompok data citra uji kata aksara lampung kelas ‘Abadi’ dan ‘Abu’. Sementara hasil terburuk dengan 1 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Ada’.
2. *kernel_size type 1B*: Hasil terbaik dengan 10 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Abu’. Sementara hasil terburuk dengan 3 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Ada’.
3. *kernel_size type 1C*: Hasil terbaik dengan 10 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Abu’. Sementara hasil terburuk dengan 3 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Ada’.

TABEL 5
SELISIH HASIL PENGUJIAN KERNEL KONVOLUSI 2X2

Tipe	Hasil Pengujian		Selisih
	Terbaik	Terburuk	
1A	10/10	1/10	9/10
1B	10/10	3/10	7/10
1C	10/10	0/10	10/10

Tabel 5 merupakan selisih hasil pengujian terhadap ukuran *kernel_size* 2x2 untuk proses konvolusi dan *kernel_size* 2x2, 3x3 dan 5x5 untuk proses pooling. Makin kecil nilai selisih pengujian antar kelas, maka makin kecil ketimpangan data pengujian dan makin baik sistem pembelajaran tersebut. Dikarenakan, dapat disimpulkan bahwa keakuratan (akurasi) dan ketepatan (presisi) sistem pengujian berlaku merata untuk setiap kategori/kelas. Tabel 5 menunjukan bahwa selisih pengujian terbaik ditunjukan oleh *kernel_size* tipe 1B dengan selisih sebesar 7/10, sementara selisih pengujian terburuk ditunjukan oleh *kernel_size* tipe 1C dengan selisih sebesar 10/10.

Hasil selisih pengujian terbaik (Tipe 1B) memiliki persebaran jumlah hasil pengujian benar yang cukup merata antar kelasnya, yaitu 8/10, 10/10, 8/10, dan 3/10. Dari data tersebut menunjukkan bahwa sistem dengan tipe ini sangat besar untuk menghasilkan pengujian dengan benar untuk setiap kategori/kelas sesuai dengan kategori/kelas-nya, kecuali kategori/kelas 4 (Ada)(3/10). Sementara hasil selisih pengujian terburuk (Tipe 1C) memiliki persebaran jumlah hasil pengujian benar yang sangat tidak merata antar kelasnya, yaitu 5/10, 10/10, 3/10, dan 0/10. Dari data tersebut menunjukkan bahwa sistem dengan tipe ini sangat besar (10/10) untuk menghasilkan pengujian dengan kategori/kelas 2 (Abu), meskipun data yang diujikan tersebut bukan merupakan kategori/kelas 2 (Abu) dan tidak akan (0/10) menghasilkan pengujian dengan kategori/kelas 4 (Ada), meskipun data yang diujikan tersebut merupakan kategori/kelas 4 (Ada).

C. Kernel Konvolusi Tipe 2 [3x3]

TABEL 6
HASIL PENGUJIAN KERNEL KONVOLUSI 3x3

		Tipe 2A			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	7	2	1	0
	Abu	0	10	0	0
	Acara	2	1	7	0
	Ada	3	5	0	2
		Tipe 2B			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	10	0	0	0
	Abu	0	10	0	0
	Acara	0	2	8	0
	Ada	0	2	0	8
		Tipe 2C			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	6	3	0	1
	Abu	1	8	1	0
	Acara	5	0	5	0
	Ada	0	3	0	7

Tabel 6 merupakan hasil pengujian sistem pembelajaran terhadap ukuran *kernel_size* 3x3 untuk proses konvolusi dan *kernel_size* 2x2, 3x3 dan 5x5 untuk proses pooling. Hasil terbaik dan terburuk dalam pengujian;

1. *kernel_size* tipe 2A: Hasil terbaik dengan 10 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Abu’. Sementara hasil terburuk dengan 2 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Ada’.
2. *kernel_size* tipe 2B: Hasil terbaik dengan 10 benar dari keseluruhan 10 data uji, ditunjukan oleh dua kelompok data citra uji kata aksara lampung kelas ‘Abadi’ dan ‘Abu’. Sementara hasil terburuk dengan 8 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Ada’.
3. *kernel_size* tipe 2C: Hasil terbaik dengan 8 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Abu’. Sementara hasil terburuk dengan 5 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Acara’.

TABEL 7
SELISIH HASIL PENGUJIAN KERNEL KONVOLUSI 2x2

Tipe	Hasil Pengujian		Selisih
	Terbaik	Terburuk	
2A	10/10	2/10	8/10
2B	10/10	8/10	2/10
2C	8/10	5/10	3/10

Tabel 7 merupakan selisih hasil pengujian terhadap ukuran *kernel_size* 3x3 untuk proses konvolusi dan *kernel_size* 2x2, 3x3 dan 5x5 untuk proses pooling. Makin kecil nilai selisih pengujian antar kelas, maka makin kecil ketimpangan data pengujian dan makin baik sistem pembelajaran tersebut. Dikarenakan, dapat disimpulkan bahwa keakuratan (akurasi) dan

ketepatan (presisi) sistem pengujian berlaku merata untuk setiap kategori/kelas. Tabel 7 menunjukkan bahwa selisih pengujian terbaik ditunjukan oleh *kernel_size* tipe 2B dengan selisih sebesar 2/10, sementara selisih pengujian terburuk ditunjukan oleh *kernel_size* tipe 2A dengan selisih sebesar 8/10.

Hasil selisih pengujian terbaik (Tipe 2B) memiliki persebaran jumlah hasil pengujian benar yang merata antar kelasnya, yaitu 10/10, 10/10, 8/10, dan 8/10. Dari data tersebut menunjukkan bahwa sistem dengan tipe ini sangat besar untuk menghasilkan pengujian dengan benar untuk setiap kategori/kelas sesuai dengan kategori/kelas-nya. Sementara hasil selisih pengujian terburuk (Tipe 2A) memiliki persebaran jumlah hasil pengujian benar yang tidak merata antar kelasnya, yaitu 7/10, 10/10, 7/10, dan 2/10. Dari data tersebut menunjukkan bahwa sistem dengan tipe ini sangat besar (10/10) untuk menghasilkan pengujian dengan kategori/kelas 2 (Abu) dengan benar dan sangat kecil (2/10) untuk menghasilkan pengujian dengan kategori/kelas 4 (Ada), meskipun data yang diujikan tersebut merupakan kategori/kelas 4 (Ada).

D. Kernel Konvolusi Tipe 3 [5x5]

TABEL 8
HASIL PENGUJIAN KERNEL KONVOLUSI 5X5

		Tipe 3A			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	10	0	0	0
	Abu	0	10	0	0
	Acara	6	0	4	0
	Ada	0	3	0	7
		Tipe 3B			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	10	0	0	0
	Abu	0	10	0	0
	Acara	5	0	5	0
	Ada	0	2	0	8
		Tipe 3C			
		Data Asli			
		Abadi	Abu	Acara	Ada
Data Prediksi	Abadi	9	1	0	0
	Abu	1	9	0	0
	Acara	3	3	4	0
	Ada	3	5	0	2

Tabel 8 merupakan hasil pengujian sistem pembelajaran terhadap ukuran *kernel_size* 5x5 untuk proses konvolusi dan *kernel_size* 2x2, 3x3 dan 5x5 untuk proses pooling. Hasil terbaik dan terburuk dalam pengujian;

1. *kernel_size* tipe 3A: Hasil terbaik dengan 10 benar dari keseluruhan 10 data uji, ditunjukan oleh dua kelompok data citra uji kata aksara lampung kelas ‘Abadi’ dan ‘Abu’. Sementara hasil terburuk dengan 4 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Acara’.
2. *kernel_size* tipe 3B: Hasil terbaik dengan 10 benar dari keseluruhan 10 data uji, ditunjukan oleh dua kelompok data citra uji kata aksara lampung kelas ‘Abadi’ dan ‘Abu’. Sementara hasil terburuk dengan 5 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Acara’.
3. *kernel_size* tipe 3C: Hasil terbaik dengan 9 benar dari keseluruhan 10 data uji, ditunjukan oleh dua kelompok data citra uji kata aksara lampung kelas ‘Abadi’ dan ‘Abu’. Sementara hasil terburuk dengan , 2 benar dari keseluruhan 10 data uji, ditunjukan oleh kelompok data citra uji kata aksara lampung kelas ‘Ada’.

TABEL 9
SELISIH HASIL PENGUJIAN KERNEL KONVOLUSI 2X2

Tipe	Hasil Pengujian		Selisih
	Terbaik	Terburuk	
3A	10/10	4/10	6/10
3B	10/10	5/10	5/10
3C	9/10	2/10	7/10

Tabel 9 merupakan selisih hasil pengujian terhadap ukuran *kernel_size* 5x5 untuk proses konvolusi dan *kernel_size* 2x2, 3x3 dan 5x5 untuk proses pooling. Makin kecil nilai selisih pengujian antar kelas, maka makin kecil ketimpangan data pengujian dan makin baik sistem pembelajaran tersebut. Dikarnakan, dapat disimpulkan bahwa keakuratan (akurasi) dan ketepatan (presisi) sistem pengujian berlaku merata untuk setiap kategori/kelas. Tabel 9 menunjukan bahwa selisih pengujian terbaik ditunjukan oleh *kernel_size* tipe 3B dengan selisih sebesar 5/10, sementara selisih pengujian terburuk ditunjukan oleh *kernel_size* tipe 3C dengan selisih sebesar 7/10.

Hasil selisih pengujian terbaik (Tipe 3B) memiliki persebaran jumlah hasil pengujian benar yang cukup merata antar kelasnya, yaitu 10/10, 10/10, 5/10, dan 8/10. Dari data tersebut menunjukkan bahwa sistem dengan tipe ini sangat besar untuk menghasilkan pengujian dengan benar untuk setiap kategori/kelas sesuai dengan kategori/kelas-nya, kecuali kategori/kelas 3 (Abadi) (5/10). Sementara hasil selisih pengujian terburuk (Tipe 3C) memiliki persebaran jumlah hasil pengujian benar yang sangat tidak merata antar kelasnya, yaitu 9/10, 9/10, 4/10, dan 2/10. Dari data tersebut menunjukkan bahwa sistem dengan tipe ini sangat besar (9/10) untuk menghasilkan pengujian antara kategori/kelas 1 (Abadi) dan 2 (Abu) dengan benar, dan sangat kecil menghasilkan pengujian dengan kategori/kelas 3 (Acara) (4/10) dan 4 (Ada) (2/10), meskipun data yang diujikan merupakan kategori/kelas tersebut.

E. Evaluasi dan Perbandingan Sistem

TABEL 10
HASIL PEMBANGUNAN DAN PENGUJIAN SISTEM PEMBELAJARAN CNN

Tipe	Pembangunan Sistem				Sistem Pengujian				
	Pengujian		Validasi		Kelas	Presisi	Recall	<i>F1-Score</i>	Akurasi
Akurasi	Loss	Akurasi	Loss						
1A	0.97	0.16	0.99	0.01	Abadi	0.56	1.00	0.72	0.6
					Abu	0.56	1.00	0.72	
					Acara	1.00	0.30	0.46	
					Ada	1.00	0.10	0.18	
1B	0.98	0.12	0.95	0.13	Abadi	1.00	0.80	0.89	0.73
					Abu	0.48	1.00	0.65	
					Acara	1.00	0.80	0.89	
					Ada	1.00	0.3	0.46	
1C	0.89	0.39	0.88	0.31	Abadi	0.56	0.50	0.53	0.45
					Abu	0.37	1.00	0.54	
					Acara	0.75	0.30	0.43	
					Ada	0.00	0.00	0.00	
2A	1.00	0.01	1.00	0.01	Abadi	0.58	0.70	0.63	0.65
					Abu	0.56	1.00	0.72	
					Acara	0.87	0.70	0.77	
					Ada	1.00	0.20	0.33	
2B	0.99	0.01	0.99	0.01	Abadi	1.00	1.00	1.00	0.9
					Abu	0.71	1.00	0.83	
					Acara	1.00	0.80	0.89	
					Ada	1.00	0.80	0.89	
2C	0.97	0.13	0.95	0.11	Abadi	0.50	0.60	0.54	0.65
					Abu	0.57	0.80	0.67	
					Acara	0.83	0.50	0.62	
					Ada	0.87	0.70	0.78	
3A	0.99	0.06	0.98	0.16	Abadi	0.62	1.00	0.76	0.77
					Abu	0.77	1.00	0.87	
					Acara	1.00	0.40	0.57	
					Ada	1.00	0.70	0.82	
3B	0.99	0.12	0.96	0.16	Abadi	0.67	1.00	0.80	0.82
					Abu	0.83	1.00	0.91	
					Acara	1.00	0.50	0.67	
					Ada	1.00	0.80	0.89	
3C	0.55	0.93	0.99	0.05	Abadi	0.56	0.90	0.69	0.62
					Abu	0.53	0.90	0.64	
					Acara	1.00	0.40	0.57	
					Ada	1.00	0.20	0.33	

Tabel 10 menunjukkan bahwa hasil pembangunan dan pengujian terbaik ditunjukan oleh kernel tipe 2B dengan akurasi pengujian sebesar 0.9. Nilai terbaik kernel tipe 2B untuk setiap parameter evaluasi pengujian lainnya (presisi, recall, dan f1-score) secara keseluruhan bernilai sama, yaitu sebesar 1.00. Sementara Nilai terburuk kernel tipe 2B untuk setiap parameter evaluasi pengujian lainnya (presisi, recall, dan f1-score) secara keseluruhan bernilai, yaitu sebesar 0.71 untuk presisi, 0.80 untuk recall, dan 0.83 untuk f1-score. Hasil-hasil tersebut menunjukkan bahwa kernel tipe 2B memperoleh hasil yang tidak hanya terbaik dari segi hasil perbandingan dengan variasi ukuran kernel lainnya namun juga hasil yang baik secara keseluruhan. Secara spesifik Tabel 10 menunjukkan bahwa hasil yang diperoleh di setiap tipe kernel konvolusi, tipe 1 (2x2), tipe 2 (3x3) dan tipe 3 (5x5) secara bertingkat menunjukkan pola yang sama, yaitu hasil terbaik ke terburuk diperoleh oleh kernel konvolusi yang diterapkan bersama kernel pooling tipe B (3x3), A (2x2), C (5x5). Namun pola tersebut tidak berlaku kebalikan, mengingat ada perbedaan nilai stride yang diterapkan pada tiap kernel konvolusi dan kernel pooling.

Penelitian ini menunjukkan bahwa pengaruh *kernel_size* sebagai operasi pengurangan ekstraksi ciri dalam menentukan nilai keluaran piksel citra, menunjukkan bahwa semakin besar ukuran kernel yang digunakan, maka semakin banyak ciri yang akan hilang (ditunjukan oleh Tabel 3, terutama pada *kernel_size* tipe 3C yang menghasilkan keluaran citra berdimensi 1x1. Secara prinsip hanya berupa citra dengan 1 nilai piksel/warna) meskipun algoritma pembelajaran sistem lebih ringan, dikarenakan sedikitnya *feature/ciri* yang harus dikenali sistem. Hal tersebut juga berlaku kebalikan, semakin kecil ukuran kernel yang digunakan, maka semakin sedikit ciri citra yang hilang (ditunjukan oleh Tabel 3, terutama pada *kernel_size* tipe 1A yang menghasilkan keluaran citra berdimensi 7x7. Secara prinsip berupa citra dengan 49 nilai piksel) namun algoritma pembelajaran sistem menjadi lebih kompleks, dikarenakan banyaknya *feature/ciri* yang harus dikenali sistem. Maka, dalam mempertimbangkan ukuran kernel yang akan digunakan perlu terlebih dahulu mempertimbangkan seberapa banyak ciri citra yang ingin dideteksi dan seberapa banyak perbedaan ciri antara citra.

Pengaruh atribut CNN dan metric pembangunan sistem juga mengambil peranan penting dalam menentukan kinerja algoritma pembelajaran sistem, sebagaimana penelitian terdahulu menunjukkan :

1. *Nanang Kasim dan Gibran Satya Nugraha* [7].: dalam upaya melakukan perbandingan atribut CNN dan metrik pembangunan sistem pada pengenalan pola tulisan aksara Arab menghasilkan hasil terbaik dengan *kernel_size* 5x5, ukuran citra 64x64, jumlah bobot dropout sebesar 30%, jumlah epoch sebanyak 80 dan pembagian dataset citra pembangunan sistem sebesar 70:30.
2. *Muhammad Alham Aqli Dilan* [8].: dalam upaya melakukan perbandingan atribut CNN dan metrik pembangunan sistem pada klasifikasi citra jenis sepatu lari berdasarkan lintasan menghasilkan hasil terbaik dengan *kernel_size* 2x2, ukuran citra 128x128, dan jumlah epoch sebanyak 80.

IV. SIMPULAN

Melalui Hasil yang telah didapat, didapatkan kesimpulan bahwa tipe kernel 2B (kernel konvolusi 3x3 dan kernel pooling 3x3) menunjukkan kinerja terbaik dalam melakukan pengenalan dan transliterasi kata aksara Lampung dengan akurasi pengujian sebesar 0.9 dan kesetimpangan data hasil pengujian yang kecil, yaitu 2/10 atau 0.2. Hasil tersebut menunjukkan bahwa kernel tersebut, memiliki hasil pengujian yang tidak hanya tepat dan akurat namun juga memiliki pendistribusian data yang baik. Sehingga, hasil pengujian tersebut tidak hanya terbaik dalam beberapa kategori/kelas saja, tapi terhadap masing-masing kategori/kelas. Ukuran Kernel 3x3 menunjukkan kondisi yang ideal untuk digunakan terutama dalam penelitian ini yang menggunakan citra dengan perbedaan ciri/features yang sangat sedikit. Selain *kernel_size*, perbedaan kompleksitas ciri citra yang diuji, atribut CNN dan metrik pembangunan sistem yang diterapkan memberikan dampak dan mempunyai peranan masing-masing dalam menentukan keberhasilan kinerja algoritma sistem pembelajaran. Diharapkan kedepannya selain dapat diaplikasikan kedalam aplikasi atau perangkat dengan kelas data yang lebih banyak lagi. Penelitian ini juga diharapkan dapat diteruskan terhadap citra yang memiliki ciri/features yang lebih kompleks serta penggunaan atribut CNN dan metrik pembangunan sistem yang lebih beragam.

UCAPAN TERIMA KASIH

Penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada Departemen Teknik Elektro, Fakultas Teknik, Universitas Lampung, atas dukungan dan bimbingannya dalam penelitian ini.

DAFTAR PUSTAKA

- [1] W. Setiawan, "Media Digital dan Tantangannya," *Semin. Nas. Pendidik.*, pp. 1–9, 2017.
- [2] T. Pudjiastuti, *Aksara dan Naskah Kuno Lampung dalam Pandangan Masyarakat Lampung Kini*, 1996th ed. Jakarta: Proyek Pengkajian dan Pembinaan Nilai-nilai Budaya Direktorat Sejarah dan Nilai Tradisional Direktorat Jenderal Kebudayaan Departemen Pendidikan dan Kebudayaan, 1996.
- [3] R. I. Indrayati and N. Migotuwio, "Identifikasi Anatomi Aksara Lampung," *AKSA J. Desain Komun. Vis.*, vol. 4, no. 1, pp. 541–551, 2020.
- [4] N. Sazqiah *et al.*, "Pengenalan Aksara Lampung Menggunakan Metode CNN (Convolutional Neural Network)," *Semin. Nas. Ins. Prof.*, vol. 2, no. 1, pp. 1–5, 2022.
- [5] D. S. Silalahi, M. M. Santoni, and A. Muliawati, "Implementasi Convolutional Neural Network Untuk Klasifikasi Kata Pada Citra Teks," *Semin.*

- Nas. Inform. Sist. Inf. dan Keamanan Siber*, vol. 4, no. 1, pp. 163–171, 2020.
- [6] A. Mulyanto, E. Susanti, F. Rossi, W. Wajiran, and R. I. Borman, “Penerapan Convolutional Neural Network (CNN) pada Pengenalan Aksara Lampung Berbasis Optical Character Recognition (OCR),” *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 1, p. 52, 2021.
 - [7] N. Kasim and G. S. Nugraha, “Pengenalan Pola Tulisan Tangan Aksara Arab Menggunakan Metode Convolution Neural Network,” *J. Teknol. Informasi, Komputer, dan Apl. (JTIKA)*, vol. 3, no. 1, pp. 85–95, 2021.
 - [8] M. I. A. Dilan, “Implementasi Algoritma Convolutional Neural Network untuk Klasifikasi Citra Jenis Sepatu Lari berdasarkan Permukaan Lintasan,” *Sienna*, vol. 4, no. 2, pp. 137–151, 2023.
 - [9] A. Fadjeri, A. Setyanto, and M. P. Kurniawan, “Pengolahan Citra Digital Untuk Menghitung Ekstrasi Ciri Greenbean Kopi Robusta dan Arabika (Studi Kasus : Kopi Temanggung),” *IJAI (Indonesian J. Appl. Informatics)*, vol. 4, pp. 92–99, 2020.
 - [10] A. Kadir and A. Susanto, *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: Andi, 2012.
 - [11] E. Paulus, S. Hadi, Y. Damayanti, F. R. Natsir, and M. Suryanti, *Analisa dan Pengolahan Citra Digital, Study Kasus Tulisan Tangan Sunda Kuno*. Bitread, 2019.
 - [12] N. Z. Munantri, H. Sofyan, and M. Y. Florestiyanto, “Aplikasi Pengolahan Citra Digital Untuk Identifikasi Umur Pohon,” *Telematika*, vol. 16, no. 2, p. 97, 2019.
 - [13] J. Hidayat, Usman, A. Faisal, and Syafriwel, “Perbandingan Metode Perbaikan Kualitas Citra Berbasis Histogram Equalization Pada Citra Satelit,” *J. Electr. Technol.*, vol. 4, no. 3, pp. 111–115, 2019.
 - [14] N. Sharma, V. Jain, and A. Mishra, “An Analysis of Convolutional Neural Networks for Image Classification,” *Procedia Comput. Sci.*, vol. 132, pp. 377–384, 2018.
 - [15] R. Shanmugamani, *Deep Learning for Computer Vision*, First. Packt Publishing Ltd, 2018.
 - [16] I. Gunawan, “Optimasi Model Artificial Neural Network untuk Klasifikasi Paket Jaringan,” *Simetris*, vol. 14, no. 2, pp. 1–5, 2020.
 - [17] H. C. Mayana and D. Leni, “Deteksi Kerusakan Ban Mobil Menggunakan Convolutional Neural Network dengan Arsitektur ResNet-34,” *J. Surya Tek.*, vol. 10, no. 2, pp. 842–851, 2023.