

BAB II

TINJAUAN PUSTAKA

2.1 Aksara Sunda

Aksara Sunda merupakan aksara (huruf) yang merupakan warisan budaya masyarakat Sunda. Menurut catatan sejarah aksara Sunda telah digunakan sejak abad ke-14 sampai abad ke-18 oleh masyarakat Sunda (Chaidir et al., 2019). Aksara sunda memiliki bentuk persegi yang menunjukkan ketajaman yang mencolok, dan sebagian kecil yang berbentuk bundar (Farhan et al., 2017). Menurut (Amalia et al., 2020) aksara Sunda dikelompokkan menjadi 5 yaitu aksara swara (vokal), aksara Ngalagéna (konsonan), aksara Sunda khusus, rarangken, dan angka.

2.1.1 Aksara Swara

Aksara swara adalah aksara yang secara silabis memiliki harkat bunyi vokal (Ernawati et al., 2017). Aksara swara berjumlah 7 huruf yaitu a, é, i, o, u, e, dan eu.



Gambar 2.1 Aksara Swara

2.1.2 Aksara Ngalagéna

Aksara Ngalagéna adalah sistem tulisan yang memiliki kemampuan untuk merepresentasikan bunyi fonem konsonan dalam bentuk kata atau suku kata (Ismawan et al., 2020). Terdapat 18 aksara Ngalagéna (konsonan) yaitu ba, ca, da, ga, ha, ja, ka, la, ma, na, nga, nya, pa, ra, sa, ta, wa, dan ya.

ka = ꦏꦏ	ga = ꦒꦒ	nga = ꦒꦏ
ca = ꦕꦕ	ja = ꦗꦗ	nya = ꦒꦚ
ta = ꦠꦠ	da = ꦢꦢ	na = ꦒꦏ
pa = ꦥꦥ	ba = ꦧꦧ	ma = ꦩꦩ
ya = ꦪꦪ	ra = ꦫꦫ	la = ꦭꦭ
wa = ꦮꦮ	sa = ꦱꦱ	ha = ꦲꦲ

Gambar 2.2 Aksara Ngalagéna

2.1.3 Aksara Khusus

Aksara khusus merupakan aksara yang tidak terikat oleh vokal, terdapat 4 aksara khusus dalam aksara sunda yaitu ie atau ieu, re atau reu, ro dan tra.

ie/ieu = ꦲꦶ ꦲꦶꦸ	re/reu = ꦫꦺ ꦫꦺꦸ
ro = ꦫꦺ	tra = ꦠꦫꦲ

Gambar 2.3 Aksara Khusus

2.1.4 Rarangken

Aksara Sunda memiliki sistem penulisan yang dikenal sebagai vokalisasi. Vokalisasi ini menggunakan rangken atau penanda bunyi yang berperan dalam mengubah, menambahkan, atau bahkan menghilangkan bunyi vokal dalam aksara Sunda (Ismawan et al., 2020). Rarangken atau tanda vokalisasi dalam aksara sunda berjumlah 13, dikelompokkan menjadi 3 berdasarkan letak penulisannya yaitu sebagai berikut:

- Rarangken diatas huruf berjumlah 5 buah.
- Rarangken dibawah huruf berjumlah 3 buah.
- Rarangken sejajar dengan huruf berjumlah 5 buah.

2.1.5 Angka Sunda

Angka dalam aksara sunda berjumlah 10 digit, yaitu 0, 1, 2, 3, 4, 5, 6, 7, 8, dan 9.

$\mathbb{G} = 1$	$\mathbb{L} = 2$
$\mathbb{K} = 3$	$\mathbb{7} = 4$
$\mathbb{U} = 5$	$\mathbb{L} = 6$
$\mathbb{M} = 7$	$\mathbb{J} = 8$
$\mathbb{L} = 9$	$\mathbb{O} = 0$

Gambar 2.4 Angka Sunda

2.2 Deep Learning

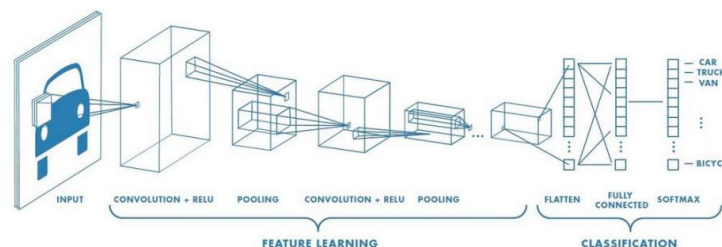
Pembelajaran mendalam atau *Deep Learning* merupakan bagian dari kecerdasan buatan yang banyak digunakan untuk memprediksi peluang, mengenali objek, hingga mendiagnosa penyakit (Purnama et al., 2022). Secara umum metode deteksi dalam *Deep Learning* dibagi menjadi 2 yaitu *region-based* dan *pixel-based*, namun dalam beberapa tahun terakhir terdapat model baru dalam *Deep Learning* yaitu *structural crack detection* (Elghaish et al., 2022).

Deep Learning memiliki keunggulan dalam bidang *computer vision* untuk mengklasifikasikan citra yang cepat dan tepat. Terdapat beberapa algoritma yang dapat menyelesaikan klasifikasi gambar dengan *Deep Learning*, yaitu *Multi-Layer Perceptron* (MLP) dan *Convolutional Neural Network* (Purnama et al., 2022).

2.2.1 Convolutional Neural Network

Convolutional Neural Network (CNN) adalah bagian dari *Deep Learning* karena kedalaman jaringannya (Kirana et al., 2020). CNN merupakan

pengembangan dari *multi-layer perceptrо* (MLP) yang dirancang untuk memproses data citra (Purnama et al., 2022). CNN digunakan untuk mengklasifikasikan data yang telah diberi label menggunakan metode *supervised learning* (Sazqiah et al., 2022).



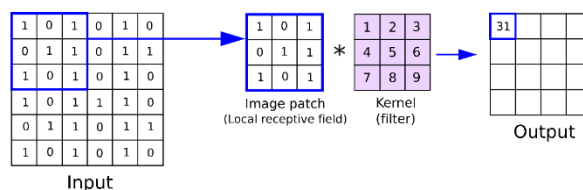
Gambar 2.5 Arsitektur CNN
Sumber: (Prihatiningsih et al., 2019)

Berdasarkan gambar 2.5 yang merupakan ilustrasi dari arsitektur CNN. Arsitektur CNN dibagi menjadi 2 bagian besar yaitu, *convolutional layer* (yang terdiri dari *pooling*, *activation*, dan *hyperparameter*) dan *fully connected layer*.

1. Convolutional Layer

Convolutional layer merupakan merupakan elemen kunci dalam CNN. Pada proses *image classification*, *input* yang diterima berupa satu atau beberapa matriks dan akan menghasilkan *output* beberapa matriks yang jumlahnya dapat berbeda dari *input* (Li et al., 2014). Operasi konvolusi bertujuan untuk mengekstraksi ciri atau *feature* dari *input* yang selanjutnya akan digunakan sebagai parameter pada proses *learning*. Pada prosesnya, operasi konvolusi melibatkan perkalian titik dari dua buah fungsi yang menghasilkan *output filter* atau *kernel*. Kemudian *filter* tersebut digeser pada citra yang disebut *stride* (Rahmawati et al., 2021). Untuk mempertahankan

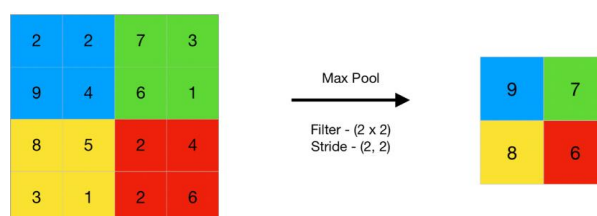
dimensi matriks *input*, maka ditambahkan proses *padding* yang akan mengisi setiap sisi dengan angka nol (Yamashita et al., 2018).



Gambar 2.6 Feature Map
Sumber: <https://anhreynolds.com>

2. Pooling

Pooling berperan penting dalam CNN dalam pengurangan dimensi fitur (Li et al., 2014) untuk mempercepat proses *learning* (Rahmawati et al., 2021). Terdapat 2 metode yang bisa digunakan pada *pooling layer*, yaitu *average pooling* dan *max pooling* (Li et al., 2014). Perbedaannya terdapat pada nilai yang diambil. *Average pooling* mengambil nilai rata-rata sebuah matriks, sedangkan *max pooling* mengambil nilai tertinggi sebuah matriks yang mengalami operasi *pooling* (Yamashita et al., 2018). Berikut ini adalah contoh gambar operasi *max pooling*.

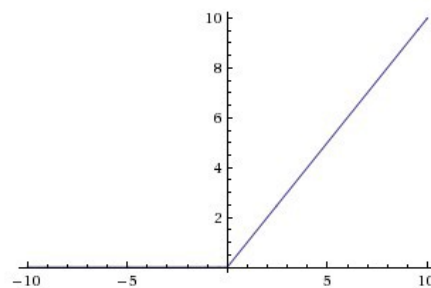


Gambar 2.7 Max Pooling
Sumber: www.geeksforgeeks.org

3. Activation Function: ReLU

Rectified Linear Units (ReLU) bersifat *linear* unit, sehingga apapun yang didapat dari nilai konvolusi itulah yang akan digunakan untuk operasi

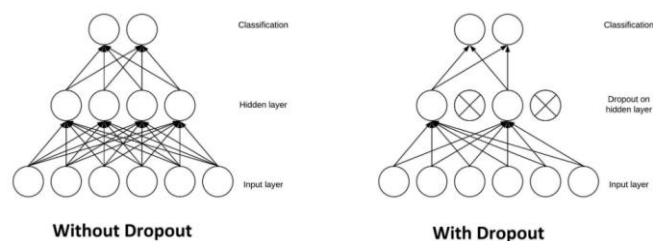
selanjutnya. Penggunaan fungsi aktivasi ReLU mempercepat *learning* dan kinerja klasifikasi dalam CNN (Li et al., 2014). Fungsi ReLU digunakan untuk menghilangkan nilai negatif pada citra dengan cara mengganti nilainya dengan nilai 0.



Gambar 2.8 Grafik ReLU
Sumber: <https://medium.com>

4. Dropout

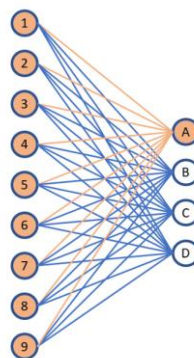
Dropout merupakan proses yang bertujuan untuk menghilangkan beberapa neuron secara acak dan tidak dipakai selama proses *training* (Li et al., 2014). Hal tersebut bertujuan untuk mencegah terjadinya *overfitting* untuk meminimalisir *error* yang dihasilkan (Rahmawati et al., 2021). Berikut perbedaan menggunakan dan tidak menggunakan *dropout*.



Gambar 2.9 Dropout
Sumber: <https://www.baeldung.com/cs/ml-ReLU-dropout-layers>

5. *Fully Connected Layer*

Fully Connected Layer adalah lapisan dimana semua neuron aktivasi dari lapisan sebelumnya terhubung semua ke neuron di lapisan berikutnya. Setiap *input* terhubung sepenuhnya ke *output* akhir yang berjumlah sama dengan kelas yang dilatih (Yamashita et al., 2018).



Gambar 2.10 *Fully Connected Layer*
 Sumber: <https://towardsdatascience.com>

6. *Softmax Classifier*

Softmax classifier digunakan untuk melakukan klasifikasi *multiclass*. Fungsi aktivasi *softmax* mengubah *output* dari *fully connected layer* yang terhubung dengan kelas probabilitas target menjadi angka riil. Setiap angka bernilai antara 0 dan 1, jika dijumlahkan semua bernilai 1 (Yamashita et al., 2018).

2.3 OPENCV

OPENCV (*Open Computer Vision*) merupakan sebuah *library* API (*Application Programming Interface*) yang sering digunakan pada *computer vision*. *Computer vision* adalah cabang ilmu dari bidang pengolahan citra atau *image processing*. Dengan vision komputer dapat melihat layaknya manusia dan bisa

mengambil keputusan, melakukan aksi, dan mengenali suatu objek tertentu. OPENCV merupakan *library open source* untuk *computer vision* yang didesain untuk aplikasi *real time*, dan memiliki fungsi akuisisi yang baik untuk *image/video*. Beberapa implementasi dari *computer vision* adalah *face recognition*, *face/object tracking*, *road tracking*, dan lain-lain (Rido et al., 2015).

2.4 Performansi Sistem

Performansi sistem dapat diukur berdasarkan parameter sebagai berikut:

1. *Inference time*

Inference time adalah waktu yang dibutuhkan sistem untuk memproses *input* data melalui suatu model atau algoritma dan menghasilkan *output* prediksi. *Inference time* menjadi faktor penting dalam menentukan sebuah sistem dalam mendeteksi suatu objek secara *real time*, semakin kecil *inference time* maka sistem semakin bagus. Pada sistem ini *inference time* dapat dihitung menggunakan rumus dibawah:

$$IT = ET - ST \quad (i)$$

Keterangan:

IT : *Inference time*

ET : *End Time*

ST : *Start Time*

2. FPS

Frame per second atau FPS adalah ukuran dari seberapa sering layar pada perangkat menampilkan gambar yang berubah setiap detiknya. FPS dibiasa digunakan untuk mengukur kinerja atau performa ketika menjalankan *game*,

video atau animasi yang bergerak secara halus. FPS dapat dihitung menggunakan rumus dibawah:

$$FPS = \frac{FRAME}{WAKTU} \quad (ii)$$

3. Accuracy

Accuracy adalah ukuran yang digunakan untuk menilai sejauh mana hasil pengukuran mendekati nilai sebenarnya. Nilai akurasi mencerminkan perbandingan keseluruhan dari semua data klasifikasi dengan aturan yang telah ditetapkan (Nurhayati et al., 2022). Jika suatu sampel dianggap positif dan diklasifikasikan dengan benar sebagai positif, maka nilai tersebut adalah *True Positive* (TP). Namun, jika diklasifikasikan sebagai negatif, maka nilai tersebut adalah *False Negative* (FN). Jika sampelnya negatif dan diklasifikasikan dengan benar sebagai negatif, maka nilai tersebut adalah *True Negative* (TN). Namun, jika diklasifikasikan sebagai positif, maka nilai tersebut adalah *False Positive* (FP) (Tharwat, 2021). Untuk menghitung nilai akurasi dapat menggunakan rumus sebagai berikut:

$$AKURASI = \frac{TP+TN}{TP+TN+FP+FN} \quad (iii)$$

2.5 Penelitian Terdekat

Beberapa penelitian mengenai aksara Sunda sudah banyak dilakukan, diantaranya penelitian yang dilakukan oleh (Purnama et al., 2022) mengenai *Implementation Of Deep Learning For Handwriting Imagery Of Sundanese Script Using Convolutional Neural Network Algorithm* (CNN) menghasilkan akurasi 97,5% menggunakan 40 data testing, dan 98% menggunakan 50 data yang berasal dari tulisan tangan pada kertas HVS.

Penelitian dari (Kirana et al., 2020) mengenai pengenalan pola aksara Sunda dengan metode *Convolutional Neural Network*. Dari hasil pengujian menggunakan citra dari buku elektronik mendapatkan akurasi sebesar 72,1%, menggunakan citra huruf komputer mendapatkan akurasi 100%, menggunakan citra yang diambil menggunakan kamera ponsel mendapatkan akurasi 87,1%, dan hasil dari data yang diambil dari pemindai mendapatkan akurasi 85,5%.

Selain itu, penelitian yang dilakukan (Amalia et al., 2020) mengenai pengenalan aksara sunda menggunakan metode jaringan saraf tiruan *Backpropagation* dan deteksi tepi canny. Pengujian menggunakan 70 data latihan menghasilkan rata-rata akurasi 90%, pengujian dengan menggunakan 21 data uji menghasilkan rata-rata akurasi sebesar 76,1905%.

Penelitian yang dilakukan oleh (Rahmawati et al., 2021) mengenai Implementasi *Deep Learning* Pada Pengenalan Aksara Sunda Menggunakan Metode *Convolutional Neural Network*. Hasil akurasi terbesar 98,03% menggunakan *epoch* 500, *learning rate* 0,1 dan menggunakan optimasi ADAM. Pengujian menggunakan kernel 3x3 dan 5x5 menghasilkan akurasi 96,71% dengan *epoch* 100 dan *learning rate* 0,001.

Selain itu penelitian dari (Lestari et al., 2015) mengenai Perancangan Pengenal Kata Dalam Aksara Sunda Menggunakan Metode Deteksi Tepi Dan LVQ Berbasis Pengolahan Citra Pada Android. Pengujian menggunakan 10 kata aksara Sunda mendapatkan akurasi paling tinggi sebesar 60,90% sedangkan menggunakan 30 kata aksara Sunda mendapatkan akurasi paling tinggi sebesar 17,78%. Pada penelitian ini disimpulkan bahwa semakin kecil *learning rate* maka akurasi

semakin besar. Selain itu, dengan melakukan *cropping background* menghasilkan akurasi lebih baik daripada tidak melakukan *cropping background*, yaitu sebesar 61,53% dengan *cropping background* dan 43,07% tanpa *cropping background*.

Pengenalan huruf lainnya seperti pada penelitian (Sazqiah et al., 2022) mengenai Pengenalan Aksara Lampung Menggunakan Metode CNN (*Convolutional Neural Network*). Menghasilkan akurasi sebesar 100% setelah dihitung menggunakan *confusion matrix*.

Penelitian dari (Willyanto et al., 2021) mengenai Identifikasi Tulisan Tangan Aksara Jepang *Hiragana* Menggunakan Metode CNN Arsitektur VGG-16. Pengujian pertama menggunakan *optimizer* ADAM dengan *learning rate* 0,0001 mendapatkan akurasi 97,5%. Pengujian kedua menggunakan *optimizer* SGD dengan *learning rate* 0,0112 mendapatkan akurasi 36,4%. Lalu pada pengujian ketiga menggunakan *optimizer* RMSprop dengan *learning rate* 0,0001 mendapatkan akurasi sebesar 78%.

Selain itu, penelitian dari (Dewa et al., 2018) mengenai *Convolutional Neural Network For Handwritten Javenese Character Recognition*. Hasil akurasi dari CNN lebih baik daripada MLP, namun waktu *training* CNN lebih lama dibandingkan dengan MLP. Akurasi tertinggi dari CNN sebesar 89%, sedangkan MLP sebesar 63%.

Penelitian dari (Prihatiningsih et al., 2019) mengenai Analisa Performa Pengenalan Tulisan Tangan Angka Berdasarkan Jumlah Iterasi Menggunakan Metode *Convolutional Neural Network*. Peningkatan akurasi pada iterasi 0-20 sangat signifikan, akan tetapi pada iterasi 100-1000 perubahan tidaklah signifikan.

Akurasi yang diperoleh pada iterasi 1000 sebesar 100% pada data *training*, 98,7% pada data validasi, dan 98,99% pada data *testing*.

Selain itu, penelitian mengenai Pengenalan Pola Tulisan Tangan Suku Kata Aksara Sasak Menggunakan Metode *Moment Invariant* Dan *Support Vector Machine* yang dilakukan oleh (Yulianti et al., 2019) pada pengujian pertama diperoleh 63 fitur dengan nilai akurasi klasifikasi 89,76%, lalu pada pengujian kedua diperoleh 112 fitur dengan nilai akurasi klasifikasi 92,52%. Proses ekstraksi fitur *moment invariant* dengan penambahan proses *thinning* mampu meningkatkan akurasi sebesar 11,82%. Selain itu, penambahan jumlah fitur dengan menggunakan *zoning* juga mampu meningkatkan akurasi sebesar 15,61%.

2.6 State of Art

Tabel 2.1 State of Art

No	Nama Pengarang	Tahun	Judul	Metode/Algoritma	Masalah	Solusi
1	Arif Purnama, Saeful Bahri, Gunawan, Taufik Hidayatulloh, Satia Suhada	2022	<i>Implementation Of Deep Learning For Handwriting Imagery Of Sundanese Script Using Convolutional Neural Network Algorithm (CNN)</i>	CNN	Saat ini tidak semua masyarakat mengenal aksara Sunda karena adanya pergeseran nilai dan anggapan bahwa aksara Sunda sulit untuk dipelajari karena bentuknya yang unik dan rumit	Menggunakan CNN untuk menenali aksara Sunda.
2	Alif Kirana, Hanny Hikmayanti H, Jamulidin Indra	2020	Pengenalan Pola Aksara Sunda Dengan Metode <i>Convolutional Neural Network</i>	CNN	Sulitnya mempelajari aksara Sunda karena bentuknya yang unik dan rumit.	Pengenalan citra menggunakan CNN
3	Nisa Amalia, Eka Wahyu Hidayat, Aldy Putra Aldya	2020	Pengenalan Aksara Sunda Menggunakan Metode Jaringan Saraf Tiruan <i>Backpropagation</i>	JST <i>Backpropagation</i> , deteksi tepi	Banyak dari masyarakat saat ini yang tidak mengetahui dan mengerti tentang aksara Sunda.	Membuat program pengenalan aksara Sunda menggunakan metode jaringan saraf tiruan <i>Backpropagation</i>

No	Nama Pengarang	Tahun	Judul	Metode/Algoritma	Masalah	Solusi
			Dan Deteksi Tepi Canny			dan deteksi tepi <i>canny</i>
4	Shelvi Nur Rahmawati, Eka Wahyu Hidayat, Husni Mubarok	2021	Implementasi <i>Deep Learning</i> Pada Pengenalan Aksara Sunda Menggunakan Metode <i>Convolutional Neural Network</i>	CNN	Aksara Sunda sudah mulai terlupakan bahkan jarang digunakan oleh masyarakat Sunda dalam kehidupan sehari-hari serta kurangnya memahami bahasa daerah sendiri.	Membuat sistem yang dapat mengidentifikasi aksara Sunda menggunakan metode <i>Convolutional Neural Network</i>
5	Dea Delia Lestari, Dr. Ir. Bambang Hidayar, DEA, Nur Adini, S.T., M.T	2015	Perancangan Pengenalan Kata Dalam Aksara Sunda Menggunakan Metode Deteksi Tepi Dan LVQ Berbasis Pengolahan Citra Pada Android	Deteksi tepi, LVQ	Orang-orang sudah mulai melupakan aksara Sunda yang harus dijaga keberlangsungan dan kelestariannya.	Merancang aplikasi untuk mempermudah pengenalan aksara Sunda menggunakan metode deteksi tepi dan LVQ berbasis Android.
6	Nanda Putri Sazqiah, Yessi Mulyani, Meizano Ardhi Muhammad, Martinus, Irza	2022	Pengenalan Aksara Lampung Menggunakan Metode CNN	CNN	Aksara Lampung terancam punah akibat minat masyarakat akan aksara Lampung yang memprihatinkan.	Membuat sitem yang mampu mengenali aksara Lampung menggunakan metode CNN

No	Nama Pengarang	Tahun	Judul	Metode/Algoritma	Masalah	Solusi
	Sukmana, Gigih Forda Nama, Zulmiftah Huda, Trisyia Septiana, Panji Kurniawan, Gita Paramita Djausal		<i>(Convolutional Neural Network)</i>			
7	Aldo Willyanto, Derry Alamsyah, Hafiz Irsyad	2021	Identifikasi Tulisan Tangan Aksara Jepang Hiragana Menggunakan Metode CNN Arsitektur VGG-16	CNN	Sulitnya mempelajari huruf Hiragana bagi pemula.	Membuat program untuk mengidentifikasi tulisan tangan aksara Hiragana menggunakan metode CNN arsitektur VGG-16
8	Chandra Kusuma Dewa, Amanda Lailatul Fadhillah, Afiahayati	2018	<i>Convolutional Neural Network For Handwriting Javanese Character Recognition</i>	CNN	Masyarakat Jawa saat ini sudah tidak menggunakan aksara Jawa pada kehidupan sehari-hari.	Membuat sistem yang bisa mengenali aksara Jawa secara otomatis menggunakan CNN dan MLP
9	Siwi Prihatiningsih, Nadhiranisa Shafiy	2019	Analisa Performa Pengenalan Tulisan	CNN	Bagaimana cara yang tepat yang dapat	Membuat sistem analisa performa

No	Nama Pengarang	Tahun	Judul	Metode/Algoritma	Masalah	Solusi
	M, Feni Andriani, Nurma Nugraha		Tangan Angka Berdasarkan Jumlah Iterasi Menggunakan Metode <i>Convolutional Neural Network</i>		membuat performa CNN lebih baik.	pengenalan tulisan tangan angka berdasarkan perubahan jumlah iterasi menggunakan CNN
10	Riska Yulianti, I Gede Pasek Suta Wijaya, Fitri Bimantoro	2019	Pengenalan Pola Tulisan Tangan Suku Kata Aksara Sasak Menggunakan Metode <i>Moment Invariant</i> Dan <i>Support Vector Machine</i>	<i>Moment invariant, Support vector machine</i>	Pada penelitian yang mengklasifikasikan aksara Sunda dan Bali memiliki permasalahan yaitu transformasi citra, <i>noise reduction</i> , dan perbedaan skala citra.	Menggabungkan 2 metode yaitu <i>moment invariant</i> dan SVM, untuk memperbaiki permasalahan sebelumnya.

2.7 Matriks Penelitian

Tabel 2.2 Matriks Penelitian

No	Peneliti	Objek penelitian	Metode							Hasil	
			CNN	JST <i>Backpropagation</i>	Deteksi tepi	LVQ	MLP	Moment invariant	SVM	Akurasi	Kecepatan
1	(Purnama et al., 2022)	Aksara Sunda	√							√	
2	(Kirana et al., 2020)	Aksara Sunda	√							√	
3	(Amalia et al., 2020)	Aksara Sunda		√	√					√	
4	(Rahmawati et al., 2021)	Aksara Sunda	√							√	
5	(Lestari et al., 2015)	Aksara Sunda			√	√					√
6	(Sazqiah et al., 2022)	Aksara Lampung	√							√	
7	(Willyanto et al., 2021)	Aksara <i>hiragana</i>	√							√	
8	(Dewa et al., 2018)	Aksara Jawa	√				√			√	
9	(Prihatiningsih et al., 2019)	Tulisan Tangan Angka	√							√	
10	(Yulianti et al., 2019)	Aksara Sasak						√	√	√	

2.8 Relevansi Penelitian

Berikut adalah relevansi dari penelitian terkait:

Tabel 2.3 Relevansi Penelitian

Peneliti	(Rahmawati et al., 2021)	Penelitian yang dilakukan oleh Ilyas Samsudin
Judul	Implementasi <i>Deep Learning</i> pada pengenalan aksara Sunda menggunakan metode <i>Convolutional Neural Network</i>	Pengenalan aksara Sunda secara <i>real time</i> menggunakan metode <i>Convolutional Neural Network</i>
Masalah Penelitian	Aksara Sunda sudah mulai terlupakan bahkan jarang digunakan oleh masyarakat Sunda dalam kehidupan sehari-hari serta kurangnya memahami bahasa daerah sendiri	Belum adanya penelitian terkait pengenalan aksara Sunda atau <i>image classification</i> yang dilakukan secara <i>real time</i>
Objek Penelitian	Aksara Sunda (swara)	Aksara Sunda (Ngalagéna)
Metode	<i>Convolutional Neural Network</i>	<i>Convolutional Neural Network</i>
Implementasi	Mengimplementasikan CNN untuk mengklasifikasikan aksara Sunda (swara)	Menggunakan CNN untuk membuat sistem yang dapat mengenali aksara Sunda secara <i>real time</i>