

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

2.1.1 Aksara Sunda Berbasis *Computer Vision* Dan *Speech Recognition*

Aksara Sunda (ᮘᮞᮧᮒ ᮘᮞᮧᮒ) pada waktu jaman dahulu lebih dikenal dengan sebuah sebutan aksara Sunda kuno atau aksara Ratu Pakuan ada juga yang lebih mengenal dengan nama aksara Kaganga. Disebut aksara Sunda kuno, secara sederhana, karena sebagai pembeda dengan aksara Sunda pada sekarang ini sudah telah di berikan standard sekaligus sudah diresmikan penggunaannya untuk kepentingan masyarakat banyak. (Candra T. Munawar, 2012)

Pada Aksara Sunda tersusun sebanyak 30 aksara dasar, yaitu ada 7 aksara swara (aksara vokal): a, é, i, o, u, e, dan eu, dan ada juga 23 aksara ngalagena (konsonan bervokal /a/): ka-ga-nga, ca-ja-nya, ta-da-na, pa-ba-ma, ya-ra-la, dan wa-sa-ha. (Candra T. Munawar, 2012)

2.1.1.1 Aksara Swara

Tabel 2.1 Paper penelitian terkait (Sumber : Candra T. Munawar, 2012)

ᮘ = a	ᮘ = é	ᮞ = i	ᮞ = o
ᮞ = u	ᮘ = e	ᮘ = eu	

2.1.1.2 Aksara Ngalagena

Tabel 2.2 Paper penelitian terkait (Sumber : Candra T. Munawar, 2012)

Tempat Pelafalan	Nirsuara	Bersuara	Sengau	Semivokal	Sibilan	Celah
Velar	ᶑ	ᶒ	ᶓ			ᶑ
	ka	ga	nga			ha
Palatal	ᶔ	ᶕ	ᶖ	ᶗ		
	ca	ja	nya	ya		
Retrofleks				ᶙ		
				ra		
Dental	ᶛ	ᶜ	ᶛ	ᶝ	ᶞ	
	ta	da	na	la	wa	
Labial	ᶞ	ᶟ	ᶠ	ᶡ		
	pa	ba	ma	wa		

Terdapat sebuah lima aksara ngalagena yang di tambahkan untuk menyempurnakan fungsi aksara sunda moderen. Sebagai contoh yaitu: aksara "fa" dan "va" merupakan sebuah modifikasi dari aksara "pa" sebelumnya, kemudian

aksara "qa" dan "xa" merupakan sebuah modifikasi dari aksara "ka" dan aksara "za" merupakan modifikasi dari aksara "ja". (Candra T. Munawar, 2012)

2.1.1.3 Aksara Ngalagena Untuk Kata Serapan

Tabel 2.3 Paper penelitian terkait (Sumber : Candra T. Munawar, 2012)

Ƶ = fa	ƹ = qa	ƶ = va	Ƹ = xa	Ʒ = za	ƹƹ = kha	ƹƶ = sya
--------	--------	--------	--------	--------	----------	----------

Ada pula rarangkén yang berfungsinya untuk mengubah, menghilangkan, atau menambah bunyi atau tanda bunyi pada aksara dasar. Tiga belas rarangkén menurut posisinya dapat dikelompokkan menjadi 3 (tiga) yaitu: (1) lima rarangkén di atas huruf, (2) tiga rarangkén di bawah huruf, dan (3) lima rarangkén sejajar dengan huruf. Untuk menuliskan angka, aksara ini memiliki 10 angka dasar (dari 0 sampai 9). (Candra T. Munawar, 2012)

2.1.1.4 Rarangkén Di Atas Huruf

Tabel 2.4 Paper penelitian terkait (Sumber : Candra T. Munawar, 2012)

	<p>panghulu, membuat vokal aksara Ngalagena dari [a] menjadi [i].</p> <p>Contoh: ƹ̄ (ki)</p>
	<p>pamepet, membuat vokal aksara Ngalagena dari [a] menjadi [ə].</p> <p>Contoh: ƹ̄̄ (ke)</p>

	paneuleung, membuat vokal aksara Ngalagena dari [a] menjadi [i]. Contoh: $\check{\text{7}}$ (keu)
	panglayar, menambah konsonan [r] pada akhir suku kata. Contoh: $\check{\text{7}}$ (kar)
	panyecek, menambah konsonan [ŋ] pada akhir suku kata. Contoh: $\check{\text{7}}$ (kang)

2.1.1.5 Rarangkén Di Bawah Huruf

Tabel 2.5 Paper penelitian terkait (Sumber : Candra T. Munawar, 2012)

	panyuku, membuat vokal aksara Ngalagena dari [a] menjadi [u]. Contoh: $\check{\text{7}}$ (ku)
	panyakra, menambah konsonan [r] di tengah suku kata. Contoh: $\check{\text{7}}$ (kra)
	panyiku, menambah konsonan [l] di akhir suku kata. Contoh: $\check{\text{7}}$ (kla)
	pamintel, menambah konsosnan [m] di tengah suku kata. Contoh: $\check{\text{7}}$ (kma)
	papasangan, menambah konsosnan [w] di tengah suku kata. Contoh: $\check{\text{7}}$ (kwa)

2.1.1.6 Rarangkén Sejajar Huruf

Tabel 2.6 Paper penelitian terkait (Sumber : Candra T. Munawar, 2012)

	<p>panéléng, membuat vokal aksara Ngalagena dari [a] menjadi [ɛ].</p> <p>Contoh: z77 (ké)</p>
	<p>panolong, membuat vokal aksara Ngalagena dari [a] menjadi [o].</p> <p>Contoh: 77z (ko)</p>
	<p>pamingkal, menambah konsonan [j] di tengah suku kata.</p> <p>Contoh: 77j(kya)</p>
	<p>pangwisad, menambah konsonan [h] di akhir suku kata.</p> <p>Contoh: 77h (kah)</p>
	<p>patén atau pamaéh, meniadakan vokal pada suku kata.</p> <p>Contoh: 77½ (k)</p>
	<p>avagraha, memisahkan bunyi vokal dari konsonan di akhir suku kata.</p> <p>Contoh: 77 7 (k'a)</p>

Dilihat dari tampilan, huruf ngalagena termasuk rarangkén memiliki sudut 45°–75°. Umumnya, rasio dimensi huruf (tinggi:lebar) adalah 4:4, kecuali untuk huruf ngalagena ra (4:3), ba dan nya (4:6), dan aksara swara i (4:3). Rarangkén

memiliki rasio dimensi 2:2, kecuali untuk panyecek (1:1), panglayar (4:2), panyakra (2:4), pamaéh (4:2) dan pamingkal (2:4 sisi bawah, 3:2 sisi kanan). Angka memiliki rasio dimensi 4:4, kecuali untuk angka 4 dan 5 (4:3). (Candra T. Munawar, 2012)

2.1.2 *Computer Vision Dan metode*

Dalam dunia *computer vision*, perkembangan terbaru telah mengarah pada penerapan teknologi *transformers* dalam berbagai tugas, termasuk deteksi objek. *Transformer*, yang dikenal dalam pemrosesan bahasa alami, membuka pintu untuk pendekatan baru yang lebih efisien dan akurat untuk deteksi huruf-huruf aksara dalam Aksara Sunda, teknologi *transformer* dapat digunakan secara *end-to-end*, menghadirkan pendekatan yang komprehensif dan efisien. (Han et al., 2023)

Berikut adalah poin-poin utama dalam pendekatan ini:

1. **Deteksi Objek dengan *Transformers*:** Model *transformer*, yang telah dilatih dengan dataset yang luas, digunakan untuk melakukan deteksi objek secara langsung pada gambar-gambar tersebut. *Transformer* mampu mengidentifikasi lokasi dan batas-batas dari huruf-huruf aksara dengan akurasi tinggi. (Carion et al., 2020)
2. **Klasifikasi dan *OCR*:** Setelah deteksi objek dilakukan, langkah klasifikasi dilakukan menggunakan metode *transformer* untuk mengidentifikasi jenis huruf yang terdeteksi. Selain itu, teknologi *OCR (Optical Character Recognition)* juga dapat diterapkan untuk mengubah teks dalam gambar menjadi teks yang dapat diakses dan dicari. (Kim et al., 2022)

Dengan pendekatan ini, deteksi huruf-huruf aksara dalam Aksara Sunda dapat dilakukan secara komprehensif dan efisien, memanfaatkan kekuatan transformer dalam memahami konteks visual dan bahasa.

Namun pada prakteknya penggunaan dua model untuk deteksi huruf-huruf aksara dalam Aksara Sunda diperlukan karena kompleksitas tugas yang berbeda. Pertama, deteksi aksara satu huruf mengharuskan model untuk mengenali dan membedakan setiap huruf secara individu. Kedua, deteksi aksara dalam bentuk kalimat membutuhkan model yang mampu memahami konteks dan hubungan antara huruf-huruf untuk menghasilkan hasil yang lebih akurat yaitu:

2.1.2.1 Model untuk Deteksi Aksara Satu Huruf

1. Model

Model ini mendeteksi aksara atau huruf (deteksi objek) akan fokus pada identifikasi dan klasifikasi setiap huruf secara individual. Model tersebut akan menggunakan arsitektur *deep learning*, seperti *transformer*, untuk melakukan deteksi objek dan klasifikasi huruf-huruf aksara. (Purnama et al., 2022)

2. Fungsi Loss

Untuk mengukur kesalahan dalam klasifikasi huruf-huruf aksara, fungsi *loss* yang umum digunakan adalah *Cross-Entropy Loss*. Rumusnya adalah sebagai berikut:

$$\text{CrossEntropyLoss} = -\frac{1}{N} \sum_{i=1}^N y_i \log(\hat{y}_i, c) \quad (2.1)$$

y_i adalah *one-hot encoded* dari kelas huruf yang sebenarnya,

\hat{y}_i, c adalah probabilitas prediksi untuk kelas huruf tersebut,

N adalah jumlah sampel,

C adalah jumlah kelas huruf.

2.1.2.2 Model untuk Deteksi Aksara dalam Bentuk Kalimat

1. Model

Model ini mendeteksi aksara dalam bentuk kalimat akan berfokus pada pemahaman konteks dan hubungan antara huruf-huruf dalam sebuah kalimat. Model ini akan dilatih dengan menggunakan pendekatan pemrosesan bahasa alami transformer dan mungkin juga penggunaan OCR untuk memahami struktur kalimat serta hubungan antara huruf-huruf dalam kalimat tersebut. (Li et al., 2023)

2. Fungsi Loss

Fungsi loss yang umum digunakan untuk mengukur kesalahan dalam prediksi lokasi dari *bounding box* (kotak pembatas) huruf-huruf aksara adalah *Mean Squared Error* (MSE). Rumusnya adalah sebagai berikut:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2.2)$$

y_i adalah koordinat *ground truth* dari *bounding box*,

\hat{y}_i, c adalah koordinat prediksi dari *bounding box*,

N adalah jumlah sampel.

Dengan menggunakan dua pendekatan yang berbeda untuk dua tugas yang berbeda ini, kita dapat mengoptimalkan kinerja dalam mendeteksi huruf-huruf aksara dalam berbagai konteks dan dengan tingkat akurasi yang lebih tinggi.

Pendekatan ini memungkinkan untuk memahami teks dalam bahasa Sunda secara lebih menyeluruh dan akurat, baik dalam level huruf maupun dalam level kalimat.

2.1.3 *Speech Recognition dan Metode*

Dalam dunia pengenalan ucapan, terobosan terbaru datang melalui penerapan teknologi transformer dengan pendekatan *wav2vec2* untuk bahasa Sunda. *Wav2vec2*, dikembangkan oleh Facebook AI, memanfaatkan representasi suara mentah dan dilatih untuk mengenali pola dalam data audio tanpa transkripsi manual. Pendekatan ini meningkatkan akurasi dan efisiensi pengenalan ucapan, terutama untuk bahasa dengan sumber daya terbatas seperti bahasa Sunda. (Jain et al., 2023)

2.1.3.1 *Pengenalan Ucapan Dengan Metode Wav2Vec*

Dalam pendekatan *wav2vec* untuk pengenalan ucapan bahasa Sunda, prosesnya melibatkan beberapa langkah penting:

1. **Perekaman Suara:** Langkah awal tetap sama, yaitu perekaman suara. Suara dalam bahasa Sunda diambil menggunakan mikrofon atau sumber *audio* lainnya.
2. **Preprocessing:** Suara yang direkam kemudian diproses menggunakan teknik *wav2vec2* dengan merubah *sampling rate* dan untuk membersihkan *noise*. Hal ini penting untuk memastikan hasil pengenalan yang akurat.
3. **Feature Extraction:** Berbeda dengan pendekatan tradisional, dalam *wav2vec2*, fitur-fitur penting dari suara diekstraksi secara otomatis menggunakan model *transformer* yang telah dilatih sebelumnya.

4. **Model Speech Recognition:** Model *wav2vec2* yang telah dilatih dengan data bahasa Sunda kemudian digunakan untuk mengenali dan menerjemahkan pola suara menjadi teks.

2.1.3.2 Perhitungan *Loss*, *WER*, dan *BLUE* dalam *Wav2Vec2*

1. **Loss (Kehilangan):** *Loss* merupakan ukuran seberapa baik model *neural network* melakukan prediksi terhadap target yang sebenarnya. Dalam *wav2vec*, *loss* biasanya dihitung menggunakan *Mean Squared Error* (MSE) atau *Cross-Entropy Loss* (CE).

- Rumus *Mean Squared Error* (MSE):

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2.4)$$

- Rumus *Cross-Entropy Loss* (CE):

$$\text{CE} = -\frac{1}{N} \sum_{i=1}^N y_i \log(\hat{y}_i) \quad (2.5)$$

2. **Word Error Rate (WER):** WER adalah metrik yang mengukur perbedaan antara teks hasil prediksi dengan teks sebenarnya dalam bentuk jumlah kata yang salah atau tertukar, diukur sebagai persentase.

- Rumus *Word Error Rate* (WER):

$$\text{WER} = \frac{S+D+I}{N} \quad (2.6)$$

di mana:

S = Jumlah kata yang disubstitusi,

D = Jumlah kata yang dihapus,

I = Jumlah kata yang dimasukkan, dan

N = Jumlah total kata dalam teks referensi.

3. **BLEU (Bilingual Evaluation Understudy)**: BLEU adalah metrik evaluasi yang digunakan untuk mengukur kualitas teks hasil prediksi dengan teks referensi, dengan memperhitungkan kesamaan *n-gram* antara keduanya.

- Rumus BLEU:

$$BLUE = BP \times \text{EXP}(\sum_{n=1}^N W_n \log(P_n)) \quad (2.7)$$

di mana:

BP = *Brevity Penalty*,

W_n = Bobot *n-gram*,

P_n = Presisi *n-gram*,

N = Panjang maksimum *n-gram*.

2.1.4 **Software dan Pemrograman Terkait**

Software dan pemrograman, bersama dengan *virtual environment* dan *library*, adalah fondasi yang sangat penting dalam pengembangan sistem seperti *speech recognition* dan *computer vision*. *Software* dan pemrograman memberikan kerangka kerja lalu *Virtual environment* memungkinkan pengembang untuk mengisolasi proyek-proyek yang dikerjakan selain itu, *library* adalah kumpulan alat dan fungsi yang tersedia untuk digunakan dalam proyek. Dengan menggunakan pengembang ini dapat memaksimalkan produktivitas mereka dan fokus pada pengembangan inti dari sistem yang ingin dibuat. (Hunt, 2023)

2.1.4.1 Library untuk Speech Recognition

1. **Torch**: *Library* utama untuk pengembangan *deep learning* di *Python*, termasuk dalam pembuatan dan pelatihan model-model *speech recognition*.
2. **Torchaudio**: *Library* khusus untuk pengolahan *audio* dengan *PyTorch*, yang membantu dalam perekaman suara dan *preprocessing audio*.
3. **Numpy**: *Library* untuk operasi numerik, sering digunakan dalam manipulasi data *audio*.
4. **Pandas**: *Library* untuk analisis data, berguna untuk mengorganisir dan memanipulasi data yang terkait dengan *speech recognition*.
5. **Jiwer**: *Library* untuk menghitung *Word Error Rate* (WER), membantu dalam evaluasi kualitas hasil pengenalan.
6. **Evaluate**: *Library* untuk evaluasi kinerja model, yang dapat digunakan untuk mengukur berbagai metrik performa *speech recognition*.
7. **Ipython.display**: *Library* untuk menampilkan *audio* dalam *notebook Jupyter*.
8. **Nltk**: *Library* untuk pemrosesan bahasa alami, dapat digunakan untuk *preprocessing* teks hasil pengenalan.
9. **Typing**: *Library* untuk *type hinting*, membantu dalam mengatur tipe data.
10. **Sklearn**: *Library* untuk *machine learning*, yang digunakan untuk membangun model klasifikasi atau melakukan evaluasi performa.
11. **Transformers**: *Library* untuk model-model *transformer*, digunakan dalam implementasi pendekatan *wav2vec*.

2.1.4.2 Library untuk Computer Vision Aksara

1. **Torch**: *Library* utama untuk pengembangan *deep learning* di *Python*, digunakan dalam pembuatan dan pelatihan model-model *computer vision*.
2. **Torchvision**: *Library* yang berisi dataset, model-model prapembelajaran, dan algoritma-algoritma umum untuk *computer vision* di *pytorch*.
3. **Sklearn**: *Library* untuk *machine learning*, digunakan dalam pembangunan model klasifikasi atau dalam evaluasi performa.
4. **Numpy**: *Library* untuk operasi numerik, sering digunakan dalam manipulasi data gambar.
5. **Pandas**: *Library* untuk analisis data, digunakan untuk mengorganisir dan memanipulasi data yang terkait dengan *computer vision*.
6. **Seaborn, matplotlib**: *Libraries* untuk visualisasi data, berguna untuk menampilkan gambar atau grafik visualisasi hasil eksperimen.
7. **Transformers**: *Library* untuk model-model *transformer*, digunakan dalam implementasi pendekatan *transformer* untuk *computer vision*.

Dengan menggunakan *library-library* ini dalam lingkungan pemrograman *Python* dengan aplikasi *Anaconda Jupyter*, dapat mengembangkan sistem *speech recognition* dan *computer vision* untuk pengenalan huruf-huruf aksara dalam bahasa Sunda dengan lebih efisien dan efektif.

2.1.4.3 Streamlit: Framework untuk Pengembangan Aplikasi

Streamlit adalah framework open-source yang memudahkan pengembang untuk membuat aplikasi web interaktif untuk data science dan machine learning. Dengan menggunakan *Python*, Streamlit memungkinkan pengembang untuk

mengubah skrip data menjadi aplikasi web yang dapat digunakan tanpa perlu pengetahuan mendalam tentang front-end development. Pengembang dapat dengan cepat menampilkan hasil analisis data, model prediksi, dan visualisasi interaktif hanya dengan beberapa baris kode. Hal ini membuat Streamlit menjadi pilihan populer di kalangan data scientist dan engineer yang ingin membangun dan membagikan aplikasi data-driven dengan mudah dan cepat. (Treuille, 2023)

2.1.4.4 SQLite: Database Relasional Terintegrasi

SQLite adalah sebuah library database relasional yang sangat ringan dan self-contained, yang diimplementasikan sebagai sebuah file tunggal. SQLite tidak memerlukan server terpisah untuk beroperasi, sehingga mudah diintegrasikan langsung ke dalam aplikasi. Karena sifatnya yang embedded, SQLite sering digunakan dalam aplikasi mobile, desktop, dan perangkat IoT yang memerlukan penyimpanan data lokal yang andal dan efisien. SQLite merupakan pilihan populer karena kemudahannya dalam penggunaan, performa yang baik, dan dukungan yang luas dalam berbagai bahasa pemrograman. (Pawlaszczyk, 2022)

2.2 Kajian Literatur

Tabel di bawah ini menyajikan ringkasan referensi penelitian sebelumnya yang terkait dengan pembuatan computer vision aksara sunda dan speech recognition.

Tabel 2.7 Paper penelitian terkait.

No.	Penulis, Tahun	Publikasi dan Akreditasi	Judul	Metode
1.	Muhammad Fitroh Amrilla, 2023	Institutional Repository UIN Syarif Hidayatullah Jakarta Q1	<i>Speech Recognition</i> pada Dialek Bahasa Sunda Menggunakan Metode <i>Convolutional Neural Network</i> dengan Ekstraksi Ciri <i>Mel-Spectrogram</i>	<i>Convolutional Neural Network (CNN)</i>
<p>Hasil:</p> <ul style="list-style-type: none"> • Penelitian ini membahas penggunaan metode (CNN) dengan ekstraksi ciri <i>mel-spectrogram</i> dalam melakukan <i>speech recognition</i> pada dialek bahasa Sunda. • Penggunaan dataset yang terdiri dari 50 data suara dialek Banten dan 50 data suara dialek Selatan, dengan model CNN yang memiliki jumlah filter 32-32-64-64, menghasilkan nilai akurasi training dan testing sebesar 93.75% dan 95.00%. 				
<p>Keterbatasan penelitian:</p> <p>Model ini terbatas hanya membandingkan sebuah dialek, yang berarti hanya berfokus pada <i>waveform</i> dan <i>spectrogram</i> sebagai bahan penelitian untuk membandingkan dialek bahasa sunda</p>				

2.	Shelvi Nur Rahmawati, Eka Wahyu Hidayat, Husni Mubarok, 2020	Prodi Sistem Jurusan Teknik Informatika Universitas Pendidikan Ganesha Q1	Implementasi <i>Deep Learning</i> pada Pengenalan Aksara Sunda Menggunakan Metode <i>Convolutional Neural Network</i>	<i>Convolutional Neural Network</i> (CNN),
<p>Hasil:</p> <ul style="list-style-type: none"> • Penelitian ini bertujuan untuk mengimplementasikan <i>deep learning</i>, khususnya metode <i>Convolutional Neural Network</i> (CNN), dalam pengenalan aksara Sunda. • Hasil penelitian menunjukkan bahwa dengan menggunakan optimasi <i>ADAM</i> dan variasi jumlah <i>epoch</i> (20, 50, 100, dan 500), nilai akurasi tertinggi yang dicapai adalah 98.03% dengan penggunaan <i>epoch</i> 500 dan <i>learning rate</i> 0.1. Selain itu, pada nilai <i>epoch</i> 100 dan <i>learning rate</i> 0.001, akurasi yang dicapai adalah 96.71% untuk data <i>training</i> dan 92.02% untuk data <i>testing</i>. 				
<p>Keterbatasan penelitian:</p> <p>Peneelitian ini hanya berfokus pengenalan Aksara Swara yang berjumlah 5 Huruf dan dan Aksara Ngalagena yang hanya untuk huruf a berjumlah 26. serta terbatas pada 2325 Citra saja untuk bahan penelitian.</p>				
3.	Alif Kirana,	Journal	Pengenalan Pola	<i>Convolutional</i>

<p>Hanny Hikmayanti H, Jamaludin Indra, 2021</p>	<p>Ubpkarawang Google Scholar No-S</p>	<p>Aksara Sunda dengan Metode <i>Convolutional</i> <i>Neural Network</i></p>	<p><i>Neural</i> <i>Network</i> (CNN)</p>
<p>Hasil:</p> <ul style="list-style-type: none"> • Pengujian dilakukan menggunakan empat jenis citra: citra dari buku elektronik, citra dari font komputer, citra dari tulisan tangan yang diambil menggunakan kamera ponsel cerdas, dan citra dari tulisan tangan responden yang diambil menggunakan pemindai. • Hasil pengujian menunjukkan bahwa metode CNN mampu mengklasifikasi citra aksara Sunda dengan akurasi yang bervariasi: <ul style="list-style-type: none"> ○ Citra dari buku elektronik: 72.4% akurasi. ○ Citra dari font komputer: 100.0% akurasi. ○ Citra dari tulisan tangan responden dengan kamera ponsel cerdas: 84.4% akurasi. ○ Citra dari tulisan tangan responden dengan pemindai: 85.5% akurasi. 			
<p>Keterbatasan penelitian:</p> <p>Penelitian ini hanya berfokus pada Aksara Swara yang berjumlah 5 Huruf. Dengan keterbatasan 847 data aksara sunda swara serta dataset ini hanya berisi aksara sunda digital sehingga input yang diberikan harus memiliki kondisi tertentu.</p>			