Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) p-ISSN: 2355-7699
Vol. 12, No. 5, Oktober 2025, hlm. 1151-1162 e-ISSN: 2528-6579

PERANCANGAN MODEL *CONVOLUTIONAL NEURAL NETWORK* PADA APLIKASI PENGENALAN AKSARA SUNDA BERBASIS *MOBILE*

Dede Kurniadi[*1], Ade Iskandar Zulkarnaen[2], Asri Mulyani[3]

[1,2,3]Institut Teknologi Garut, Kabupaten Garut
Email: [1]dede.kurniadi@itg.ac.id, [2]2006043@itg.ac.id, [3]asrimulyani@itg.ac.id
[*]Penulis Korespondensi

(Naskah masuk: 6 September 2025, diterima untuk diterbitkan: 30 Oktober 2025)

Abstrak

Mendikbudristek pada tahun 2022 menjadikan 38 bahasa daerah menjadi objek revitalisasi, salah satunya yaitu bahasa Sunda. Hal tersebut dikarenakan sebagian besar bahasa daerah di Indonesia kondisinya terancam punah dan kritis. Hilangnya bahasa daerah juga mengancam keberadaan aksara lokal yang menjadi bagian integral dari warisan budaya. Dalam upaya memperkenalkan serta mendukung revitalisasi di bidang kebahasaan, pengembangan aplikasi media pembelajaran berbasis *mobile* dapat menjadi solusi yang efektif. Hal ini didasarkan bahwa perangkat *smartphone* memiliki pengguna yang luas di kalangan masyarakat. Penelitian ini bertujuan untuk merancang dan mengimplementasikan model *Convolutional Neural Network* (CNN) untuk pengenalan aksara Sunda pada perangkat berbasis *mobile*. Model CNN diterapkan pada fitur belajar menulis untuk mengenali tulisan tangan aksara Sunda dan memberikan *feedback* kepada pengguna. Penelitian ini menggunakan metode *Machine Learning Lifecycle* (MLLC) dimana tahapan yang dilakukan meliputi *problem definition*, *data*, *model*, dan *production system*. Penelitian dimulai dengan pembuatan dataset tulisan tangan digital, yang kemudian digunakan untuk melatih model klasifikasi menggunakan arsitektur CNN VGG-16. Dataset yang berhasil dibuat sebanyak 7500 gambar yang terdiri dari aksara Sunda *swara*, *ngalagena*, dan *ngalagena serapan*. Model yang dihasilkan dari proses pelatihan dengan total 10 *epoch* memperoleh akurasi sebesar 99%, sementara pada data *testing* memperoleh akurasi rata-rata sebesar 83%. Pada tahap akhir pengujian, model diimplementasikan pada *prototype* aplikasi pengenalan aksara Sunda berbasis *mobile* dan dapat dengan baik mengklasifikasi aksara Sunda. Hasil dari penelitian ini yaitu berupa model pengenalan aksara Sunda yang dapat diterapkan pada aplikasi berbasis *mobile*. Melalui pembuatan model dan *prototype* aplikasi pengenalan aksara Sunda, penelitian ini ikut berkontribusi pada digitalisasi aksara Sunda serta menyediakan landasan untuk pengembangan dan penelitian lanjutan dalam penerapan CNN pada aplikasi berbasis *mobile*.

Kata kunci: *CNN VGG-16, MLLC, mobile application, aksara Sunda*

DESIGN OF A CONVOLUTIONAL NEURAL NETWORK MODEL FOR A MOBILE-BASED SUNDANESE SCRIPT RECOGNITION APPLICATION

Abstract

In 2022, the Minister of Education, Culture, Research and Technology made 38 regional languages the object of revitalization, one of which is Sundanese. This is because most regional languages in Indonesia are endangered and critical. The loss of regional languages also threatens the existence of local scripts, which are an integral part of cultural heritage. To introduce and support revitalization in the language field, the development of mobile-based learning media applications can be an effective solution. This is based on the fact that smartphone devices have a wide user base in society. This study aims to design and implement a Convolutional Neural Network (CNN) model for Sundanese script recognition on mobile-based devices. The CNN model is applied to the learning-to-write feature to recognize Sundanese handwriting and provide user feedback. This study uses the Machine Learning Lifecycle (MLLC) method, where the stages include problem definition, data, models, and production systems. The study began with creating a digital handwriting dataset, which was then used to train a classification model using the CNN VGG-16 architecture. The successfully created dataset was 7500 images consisting of Sundanese swara, ngalagena, and ngalagena absorption scripts. The model produced from the training process with 10 epochs obtained an accuracy of 99%, while the testing data obtained an average accuracy of 83%. In the final stage of testing, the model was implemented on a mobile-based Sundanese script recognition application prototype and could classify Sundanese script well. The results of this study are in the form of a Sundanese script recognition model that is applied to a mobile-based application prototype. By creating a model and prototype of a Sundanese script recognition application, this research contributes to the

digitalization of Sundanese script and provides a foundation for further development and research in the application of CNN to mobile base applications.

Keywords: *CNN VGG-16, MLLC, mobile application, Sundanese script*

1. PENDAHULUAN

Indonesia merupakan negara dengan keanekaragaman budaya yang sangat kaya, termasuk dalam hal bahasa daerah. Mengingat Indonesia memiliki bahasa daerah terbanyak kedua di dunia setelah Papua Nugini, dengan jumlah 718 bahasa daerah yang masing-masing memiliki keunikan dan kekayaan tersendiri (Pun, 2024). Bahasa daerah juga merupakan salah satu aspek penting dari identitas budaya suatu daerah yang mencerminkan kearifan lokal, nilai-nilai luhur budaya dan pengetahuan lokal yang telah diwariskan dari generasi ke generasi. Pelestarian bahasa daerah menjadi penting tidak hanya untuk menjaga keberagaman budaya, tetapi juga sebagai identitas dan warisan budaya yang tidak ternilai dari masing-masing daerah. Mendikbudristek Nadiem Anwar Makarim (kemdikbud, 2022) pada siaran pers merdeka belajar episode ke-17 mengungkapkan bahwa "revitalisasi bahasa daerah perlu dilakukan mengingat 718 bahasa daerah Indonesia, sebagian besar kondisinya terancam punah dan kritis". Pada tahun 2022 sebanyak 38 bahasa daerah dijadikan objek revitalisasi oleh kemdikbud, salah satunya yaitu bahasa Sunda dan Jawa (kemdikbud, 2022). Sasaran utama dari revitalisasi bahasa daerah tersebut antara lain komunitas tutur, guru, kepala sekolah, pengawas dan para siswa. Hal tersebut bertujuan untuk menjaga kelangsungan hidup bahasa dan sastra daerah serta menciptakan ruang kreativitas bagi para penutur bahasa daerah untuk mempertahankan bahasanya. Pada saat ini, penutur asli bahasa daerah cenderung tidak lagi menggunakan dan mewariskan bahasa tersebut kepada generasi berikutnya. Hal ini menyebabkan kekayaan budaya, pemikiran, dan pengetahuan yang terkandung dalam bahasa daerah tersebut menghadapi risiko kepunahan (kemdikbud, 2022). Apabila proses kepunahan tersebut terus berlanjut tanpa ada upaya pelestarian maka bukan hanya berarti kehilangan sistem komunikasi tetapi juga kehilangan sejarah, tradisi, dan identitas budaya suatu bangsa. "Hal ini disebabkan karena melalui bahasa dapat diketahui cara pandang suatu masyarakat tentang sesuatu dan melalui bahasa pula dapat diketahui aturan, tradisi, serta kepercayaan sebuah kelompok etnik"(Tondo, 2015). Selain itu, hilangnya bahasa daerah juga mengancam keberadaan aksara-aksara lokal yang menjadi bagian integral dari warisan budaya. Aksara-aksara ini bukan hanya sekadar simbol-simbol tulisan, melainkan juga merupakan representasi dari nilai-nilai, pengetahuan, dan identitas yang telah diwariskan dari generasi ke generasi. Saat ini, Indonesia memiliki 12 sistem penulisan aksara lokal yang dikenal juga sebagai Aksara Nusantara (Setiawan, 2020). Sistem-sistem penulisan aksara tersebut meliputi aksara Sunda, Jawa, Lampung, Bali, Bugis (lontara), Rejang, Pakpak, Karo, Simalungun, Toba, Mandailing, dan Kerinci (Setiawan, 2020). Beberapa aksara daerah seperti Sunda, Jawa, Bugis, Batak, Rejang, dan Pegon telah mengalami proses digitalisasi dan terdaftar di *unicode*, sehingga memungkinkan untuk dapat diakses dan dimanfaatkan secara luas di internet (Setiawan, 2020).

Dalam upaya memperkenalkan serta mendukung pelestarian dan revitalisasi kebudayaan Indonesia di bidang kebahasaan, pengembangan aplikasi media pembelajaran berbasis *mobile* dapat menjadi salah satu solusi. Hal ini didasarkan pada fakta bahwa perangkat *mobile* seperti *smartphone* memiliki pengguna yang luas di kalangan masyarakat, yang mana menurut survei BPS pada tahun 2022 mencatat bahwa 67,88% penduduk Indonesia telah memiliki telepon seluler (BPS, 2022). Aplikasi berbasis *mobile* memungkinkan pengguna untuk mengakses aplikasi tersebut dimana saja dan kapan saja, meningkatkan keterjangkauan dan kemudahan penggunaan. Pengembangan aplikasi media pembelajaran yang lebih lanjut, dapat memanfaatkan teknologi terkini seperti kecerdasan buatan (AI) untuk memberikan pengalaman belajar yang lebih interaktif. Sebagai contoh, aplikasi dapat menggunakan teknologi pengenalan suara untuk membantu pengguna mempraktikkan pengucapan bahasa daerah, atau menggunakan teknologi jaringan syaraf tiruan untuk mengenali dan menganalisis aksara tulisan tangan pengguna dalam bahasa daerah. Permasalahan yang terjadi saat ini yaitu belum tersedianya aplikasi media pembelajaran berbasis *mobile* yang mampu mengenali tulisan tangan aksara daerah secara otomatis. Secara umumnya aplikasi pembelajaran aksara yang telah dikembangkan masih terbatas pada penyajian materi yang masih bersifat statis melalui materi teks dan gambar, tanpa fitur interaktif berbasis kecerdasan buatan. Hal ini menjadi tantangan dalam menciptakan media pembelajaran yang adaptif dan responsif terhadap masukan pengguna. Untuk menjawab permasalahan tersebut, penelitian ini mengembangkan model *Convolutional Neural Network* (CNN) yang difokuskan pada pengenalan tulisan tangan aksara daerah sebagai bagian dari upaya menghadirkan media pembelajaran yang lebih interaktif yang menerapkan kecerdasan buatan. Melalui pemanfaatan CNN, penelitian ini berupaya menjembatani kesenjangan antara pelestarian aksara daerah dan teknologi modern. *Convolutional Neural*

Network (CNN) merupakan salah satu arsitektur *deep learning* yang sangat efektif dalam mengenali pola dan gambar, serta telah banyak digunakan dalam berbagai aplikasi mulai dari pengenalan wajah hingga analisis citra medis. CNN dapat diterapkan dalam aplikasi pembelajaran bahasa daerah untuk memproses dan mengenali pola aksara. Sebagai contoh, CNN dapat digunakan untuk mengenali tulisan tangan pengguna dalam aksara daerah dan memberikan *feedback* langsung mengenai kesalahan atau area yang perlu diperbaiki. Ini tidak hanya membantu pengguna dalam proses pembelajaran, tetapi juga membuat proses belajar menjadi lebih menarik.

Penelitian ini didasari oleh beberapa penelitian sebelumnya yang relevan untuk dijadikan sebagai rujukan. Penelitian pertama yang dilakukan oleh (Habibah, Sholihaningtias and Pratiwi, 2020) telah berhasil merancang dan mengembangkan aplikasi *mobile* berbasis android untuk memfasilitasi pembelajaran aksara Sunda dengan menggabungkan materi teks, audio, dan visual. Aplikasi yang dikembangkan dilengkapi dengan fitur pembelajaran berupa materi aksara Sunda dan fitur *game* pilihan ganda mengenai materi aksara Sunda. Namun pada aplikasi tersebut belum tersedia fitur belajar menulis yang dapat mengenali dan menganalisis aksara tulisan tangan pengguna. Oleh karena itu, aplikasi ini dapat dikembangkan lebih lanjut guna meningkatkan fungsionalitas media pembelajaran.

Penelitian kedua dilakukan oleh (Purnama et al., 2022) yang bertujuan untuk mengevaluasi tingkat akurasi algoritma CNN dalam mengklasifikasikan gambar aksara Sunda. Penelitian ini berhasil menunjukkan efektivitas penerapan algoritma CNN pada klasifikasi gambar tulisan tangan aksara Sunda dengan mencapai tingkat akurasi sebesar 97,5%. Hasil dari penelitian tersebut mengemukakan bahwa model klasifikasi yang di bangun dengan menggunakan arsitektur CNN mampu mengenali aksara Sunda dengan baik sehingga model tersebut dapat diimplementasikan pada aplikasi praktis.

Penelitian ketiga dilakukan oleh (Rahmawati, Hidayat and Mubarok, 2021) yang bertujuan untuk memperoleh model terbaik dalam klasifikasi tulisan tangan aksara Sunda dengan menerapkan metode CNN. Penelitian ini mengemukakan bahwa penggunaan optimasi ADAM dengan *epoch* 500 dan *learning rate* sebesar 0.1 menghasilkan akurasi tinggi sebesar 98,03%.

Penelitian keempat dilakukan oleh (Lorentius, Adipranata and Tjondrowiguno, 2021) yang bertujuan untuk mengambangkan model yang mampu mengenali aksara Jawa dengan menggunakan metode CNN. Hasil dari penelitian tersebut mengemukakan bahwa model klasifikasi yang di bangun dengan menggunakan arsitektur CNN VGG-16 mampu mengenali aksara Jawa dengan memperoleh tingkat akurasi sebesar 97,55%.

Penelitian kelima dilakukan oleh (Anggraeny, Via and Mumpuni, 2023) bertujuan untuk membandingkan berbagai metode *preprocessing* dalam meningkatkan akurasi pengenalan tulisan tangan aksara Jawa menggunakan *Artificial Neural Network* (ANN). Hasil penelitian ini mengungkapkan bahwa model yang dibangun berhasil mengenali tulisan tangan aksara Jawa dengan akurasi tertinggi sebesar 98%. Penelitian tersebut menegaskan bahwa pemilihan latar belakang dan metode *preprocessing* yang tepat dapat meningkatkan performa sistem pengenalan karakter secara signifikan.

Penelitian keenam dilakukan oleh (Kurniadi et al., 2022) untuk mengembangkan aplikasi *text-to-speech* (TTS) dalam bahasa Indonesia berbasis android. Penelitian ini bertujuan mengambangkan aplikasi yang dapat mengenali teks tulisan tangan dengan cara memindai teks tulisan tangan dan mengonversi teks tersebut menjadi suara. Hasil penelitian ini menunjukkan bahwa aplikasi yang dibangun dan diuji menghasilkan akurasi pengenalan teks yang baik, dengan akurasi pengenalan tulisan tangan sebesar 85, 25%, sedangkan dari tulisan mesin sebesar 87,35%.

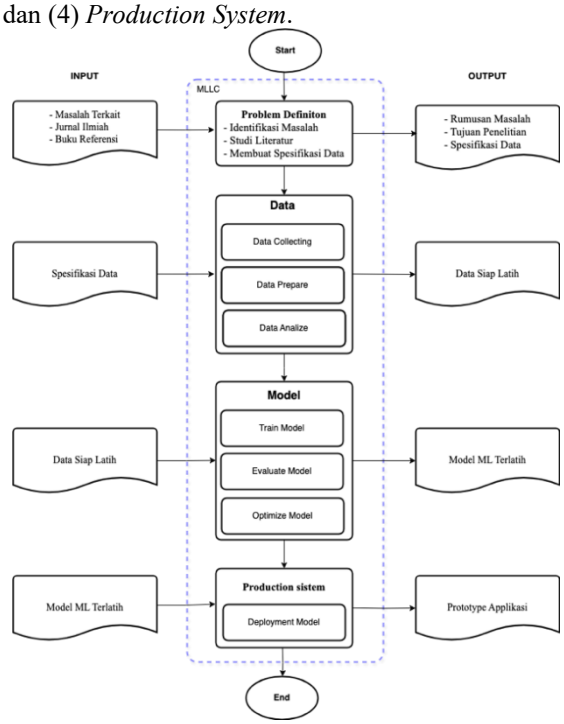
Berdasarkan penelitian sebelumnya yang telah diuraikan, dapat disimpulkan bahwa algoritma CNN dapat mengklasifikasikan aksara lokal seperti aksara Sunda dan Jawa dengan sangat baik. Sebagaimana diungkapkan juga dalam penelitian yang dilakukan oleh (Dewa, Fadhilah and Afiahayati, 2018) dengan melakukan perbandingan efektivitas antara algoritma *Convolutional Neural Network* (CNN) dan algoritma *Multilayer Perceptron* (MLP) pada pengenalan aksara Jawa dari segi akurasi dan waktu pelatihan. Hasil penelitian tersebut diperoleh capaian akurasi lebih tinggi diperoleh dengan menggunakan algoritma CNN meskipun proses *training* yang relatif lebih lama. Pada proses pemodelan, arsitektur CNN VGG dipilih karena dapat memberikan akurasi yang baik dalam klasifikasi gambar tulisan tangan dibandingkan dengan arsitektur lainnya. Hal tersebut diungkapkan dalam penelitian yang dilakukan oleh (Prameswari, Dwi Sulistiyo and Ihsan, 2023) dengan membandingkan beberapa arsitektur CNN seperti ResNet-50, MobileNet, dan VGG.

Berdasarkan rujukan penelitian dan permasalahan yang telah diuraikan, tujuan dari penelitian ini adalah untuk merancang dan mengimplementasikan model CNN pengenalan aksara Sunda yang mampu mengenali tulisan tangan pengguna secara otomatis pada perangkat berbasis *mobile*. Model ini akan diintegrasikan dalam sebuah *prototype* aplikasi pembelajaran berbasis *mobile*, khususnya pada fitur interaktif untuk belajar menulis aksara Sunda. Aplikasi ini memungkinkan pengguna menuliskan aksara dengan *stylus* atau jari dan memperoleh umpan balik terhadap tingkat kesesuaian tulisannya. Kontribusi utama dari penelitian ini adalah menghadirkan solusi

pembelajaran berbasis kecerdasan buatan yang adaptif serta memperkuat upaya pelestarian aksara Sunda melalui teknologi modern. Selain itu, model yang dikembangkan juga dapat menjadi fondasi untuk penelitian lanjutan dalam pengenalan karakter tulisan tangan berbasis aksara lokal lainnya.

2. METODE PENELITIAN

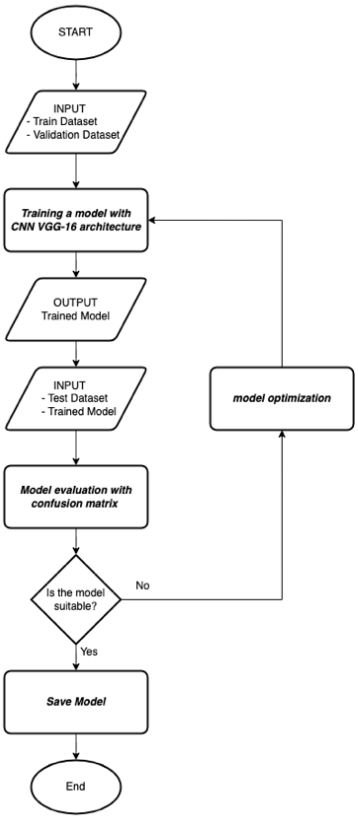
Metode yang digunakan pada penelitian ini adalah *Machine Learning Lifecyle* (MLLC). MLLC merupakan sebuah siklus *machine learning* yang bersifat *iterative* dan *incremental* dengan instruksi dan praktik terbaik untuk digunakan di seluruh fase yang sudah ditentukan pada saat mengembangkan proyek *machine learning* (amazon, 2024). MLLC memberikan kejelasan dan struktur yang diperlukan untuk keberhasilan proyek *machine learning*. Tahapan yang dilakukan dalam penelitian ini merujuk pada (Maskey et al., 2019) yang menjelaskan bahwa penerapan model ML ke dalam sistem produksi melibatkan empat tahapan utama, yaitu: (1) *Problem Definition*; (2) *Data*; (3) *Model*; dan (4) *Production System*.



Gambar 1. Tahapan Penelitian

Gambar 1 merupakan tahapan penelitian yang dilakukan. Berikut merupakan penjelasan dari setiap tahapan dari penelitian:

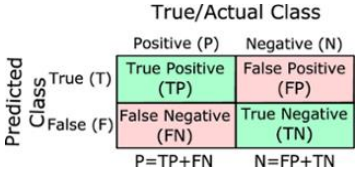
1. *Problem Definition*, pada tahap ini terdapat tiga aktivitas yaitu identifikasi masalah, melakukan studi literatur, dan membuat spesifikasi data.
   a. Identifikasi masalah, dilakukan untuk mengetahui permasalahan dan menentukan tujuan spesifik yang dapat diselesaikan dengan menggunakan *machine learning*;
   b. Studi Literatur, dilakukan untuk meninjau penelitian sebelumnya baik dari metode yang telah digunakan dan hasil-hasil yang telah dicapai. Hal ini bertujuan untuk membantu dalam mengenali kekurangan dari metode yang ada, sehingga dapat menentukan pendekatan yang lebih tepat untuk menyelesaikan masalah yang dihadapi.
   c. Membuat spesifikasi data, dilakukan untuk menyusun spesifikasi rinci mengenai kebutuhan model, batasan, dan ruang lingkup proyek, untuk memastikan solusi yang diusulkan sesuai dengan tujuan yang telah ditetapkan.
2. *Data*, pada tahap ini terdapat tiga aktivitas yaitu *data collecting, data preparation,* dan *data analyze*.
   a. *Data collecting,* dilakukan untuk membuat atau mengumpulkan dataset sesuai dengan spesifikasi data yang telah ditentukan;
   b. *Data preparation*, dilakukan untuk mempersiapkan data menjadi fitur yang koheren dan siap digunakan pada proses pelatihan. Tahap ini melibatkan proses *data labeling*, *splitting*, dan *normalization.*
   c. *Data analyze*, dilakukan untuk mendapatkan wawasan yang lebih baik mengenai data telah dipersiapkan. Langkah yang dilakukan meliputi eksplorasi data, visualisasi data dan validasi data.
3. *Model*, pada tahap ini dilakukan proses *modeling* menggunakan arsitektur CNN VGG-16. Proses di tahap modeling dapat dilihat pada *flowchart* yang disajikan pada Gambar 2.



Gambar 2. *Flowchart* tahap *Modeling*

Pada tahap modeling terdapat tiga aktivitas yaitu *train model, evaluate model,* dan *optimize model*.

a. *Train model,* bertujuan untuk melatih model *machine learning* dengan menggunakan dataset yang telah dipersiapkan.
b. *Evaluate model*, dilakukan dengan menggunakan *confusion matrix* untuk menilai seberapa baik model dalam mengklasifikasikan data dan membantu dalam memahami distribusi kesalahan yang dibuat oleh model.



Gambar 3. *Confusion Matrix* (Tharwat, 2018)

Gambar 3 menunjuk empat kemungkinan hasil prediksi, dimana diagonal yang berwarna hijau menunjukkan prediksi benar dan diagonal yang berwarna merah menunjukkan prediksi salah. Berdasarkan hasil yang diperoleh dari *confusion matrix* dapat dihitung beberapa metrik evaluasi utama, seperti akurasi, presisi, recall, dan F1-score, yang memberikan gambaran lebih komprehensif tentang performa model. Berikut merupakan persamaan untuk menghitung metrik evaluasi tersebut

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (1)$$

$$precision = \frac{TP}{TP+FT} \quad (2)$$

$$recall = \frac{TP}{TP+FN} \quad (3)$$

$$f1score = 2 \times \frac{(precision \times recall)}{(precision + recall)} \quad (4)$$

c. *Optimize model*, dilakukan untuk meningkatkan performa model yang telah dilatih, terutama jika model yang dilatih menunjukkan akurasi yang buruk. Optimalisasi model dilakukan dengan menyesuaikan *hyperparameter*, atau menerapkan teknik regularisasi untuk meningkatkan kinerja model. Tujuan dari tahap ini adalah untuk memaksimalkan akurasi dan generalisasi model tanpa mengorbankan kinerja.
4. *Production System,* pada tahap ini model yang telah dioptimalkan diimplementasikan ke dalam sebuah prototipe aplikasi berbasis *mobile*. Proses ini menghasilkan model yang telah terintegrasi dalam sistem dan siap digunakan untuk pengenalan aksara Sunda pada perangkat *mobile*.

3. HASIL DAN PEMBAHASAN

Pada bagian ini disampaikan empat tahapan sesuai metode penelitian yang disampaikan pada bagian 2.

3.1. *Problem Definition*

1. Identifikasi masalah

Berdasarkan latar belakang yang telah diuraikan pada bagian 1, peneliti menemukan bahwa rendahnya penggunaan bahasa daerah serta kecenderungan penutur asli bahasa daerah yang tidak lagi mewariskan bahasa daerah ke generasi berikutnya menyebabkan sebagian besar bahasa daerah terancam punah dan kritis. Selain itu, hilangnya bahasa daerah juga mengancam keberadaan aksara-aksara lokal yang menjadi bagian integral dari warisan budaya.

Berdasarkan permasalahan tersebut peneliti menetapkan tujuan untuk membangun *prototype* aplikasi *mobile* yang dapat mendukung pembelajaran aksara lokal khususnya aksara Sunda dengan memanfaatkan teknologi CNN. CNN diterapkan pada fitur belajar menulis yang memungkinkan pengguna menulis aksara dengan *stylus* atau jari pada layar perangkat mereka dan aplikasi akan memberikan *feedback* berupa nilai keakuratan tulisan tersebut.

2. Studi Literatur

Berdasarkan hasil studi literatur yang dipaparkan pada bagian 1, sebagian besar penelitian sebelumnya yang terkait pengenalan aksara daerah umumnya hanya sampai tahap pemodelan dan belum diimplementasikan pada aplikasi praktis. Hal tersebut menunjukkan perlunya penelitian lebih lanjut yang berfokus pada integrasi teknologi klasifikasi aksara ke dalam aplikasi praktis.

Melalui tahapan ini diperoleh wawasan dan pengetahuan yang berguna untuk merancang solusi, antara lain: Pertama, penelitian yang dilakukan oleh (Rahmawati, Hidayat and Mubarok, 2021) mengungkapkan bahwa penggunaan *optimizer* Adam memberikan hasil paling optimal dalam mengklasifikasikan aksara Sunda dengan tingkat akurasi yang tinggi. Kedua, penelitian yang dilakukan oleh (Lorentius, Adipranata and Tjondrowiguno, 2021) menunjukkan bahwa arsitektur CNN VGG-16 mampu mengklasifikasikan aksara dengan baik. Ketiga, penelitian yang dilakukan (Anggraeny, Via and Mumpuni, 2023) menyoroti pentingnya teknik *preprocessing* dalam meningkatkan akurasi pengenalan aksara dan hasil

ini menegaskan bahwa pemilihan latar belakang serta metode *preprocessing* yang tepat dapat secara signifikan meningkatkan performa sistem pengenalan karakter.

3. Membuat Spesifikasi Data

Spesifikasi data yang dibutuhkan meliputi gambar aksara Sunda dengan resolusi 224x224 *pixel* dan berlatar belakang putih. Aksara Sunda yang dicakup meliputi aksara Sunda *Swara*, *Ngalagena*, dan *Ngalagena Serapan*. Gambar tidak memiliki latar belakang yang kompleks karena penulisan dilakukan pada kanvas di *smartphone*, yang juga memiliki latar belakang putih. Oleh karena itu, latar belakang putih dipilih untuk menjaga konsistensi dengan pengalaman menulis pada perangkat *smartphone*. Spesifikasi ini disesuaikan dengan kebutuhan aplikasi yang difokuskan pada prediksi kesesuaian penulisan aksara, sehingga tidak mencakup aspek lain seperti pengenalan kalimat atau paragraf secara keseluruhan.

3.2. Data

1. *Data Collecting*

Pada tahap ini peneliti membuat dataset tulisan tangan digital aksara Sunda yang akan digunakan untuk data pelatihan dan validasi. Proses pembuatan dataset dilakukan menggunakan aplikasi menggambar pada *smartphone* atau desktop yang kemudian disimpan dalam format .png. Dataset yang berhasil dibuat dapat diakses pada link *github* berikut. Rincian hasil pembuatan dataset aksara Sunda dapat dilihat pada Tabel 1.

Tabel 1. Rincian dataset yang dibuat

No.	Label	Jenis Aksara Sunda	Jumlah
1	a	Swara	250
2	é	Swara	250
3	i	Swara	250
. . .	. . .	. . .	. . .
29	xa	Ngalagena Serapan	250
30	za	Ngalagena Serapan	250

Tabel 1 menunjukkan dataset yang berhasil dibuat yang mencakup aksara 30 kelas aksara sunda dengan total 7500 gambar.

2. *Data preparation*

Pada tahap persiapan data, peneliti mengolah data yang telah dikumpulkan agar siap digunakan dalam pelatihan model. Adapun tahapan *data preparation* yang dilakukan disajikan dalam bentuk *pseudocode*.

Data dikelompokkan ke dalam kategori atau kelas yang kemudian diatur dalam struktur direktori terpisah, memungkinkan pelabelan otomatis yang lebih efisien.

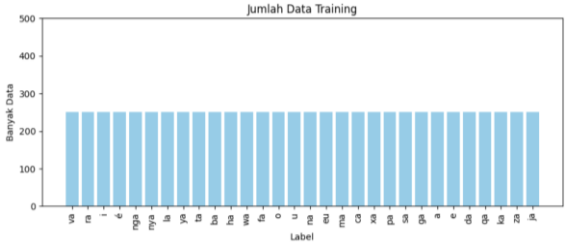
Name:
Data Preprocessing
In:
- Dataset directory containing folders of images.
Out:
- Labeled, split, and normalized training and validation datasets.
Step:
1. Automatic Labeling:
***For each** folder **in** the dataset directory:*
- Assign the folder name as the label for all images in that folder.
2. Data Splitting:
Divide the dataset into two sets:
- 80% of the images for the training set.
- 20% of the images for the validation set.
Randomly select images to ensure a representative split.
3. Data Normalization:
***For each** image **in** both training and validation sets:*
- Convert pixel values to a range between 0 and 1.
- This is done by dividing each pixel value by 255.

Dataset dibagi menjadi subset untuk pelatihan dan validasi, dimana 80% data digunakan untuk pelatihan dan 20% untuk validasi. Selain itu, data testing yang terpisah digunakan untuk evaluasi akhir model, memastikan pengujian pada data yang belum pernah dilihat sebelumnya.

Normalisasi data dilakukan untuk menjaga konsistensi skala dan intensitas gambar, sehingga mengurangi variabilitas yang tidak diperlukan dan memastikan *input* model berada dalam rentang yang konsisten. Langkah-langkah ini mempersiapkan data secara komprehensif untuk mendukung proses pelatihan model yang optimal.

3. *Data analyze*

Pada tahap analisis data, peneliti melakukan pemeriksaan menyeluruh terhadap dataset yang telah disiapkan untuk memahami karakteristik data sebelum digunakan dalam pelatihan model. Pada tahap ini dilakukan eksplorasi data untuk meneliti distribusi karakter aksara dalam dataset untuk memastikan bahwa semua kelas diwakili dengan baik dan tidak ada ketidakseimbangan yang signifikan. Distribusi karakter aksara Sunda dapat dilihat pada Gambar 4.



Gambar 4. Distribusi Dataset aksara Sunda

Pada tahap ini dilakukan validasi ulang pada dataset untuk memastikan bahwa tidak ada gambar yang rusak atau tidak relevan. Dengan melalui tahap ini, data dipastikan siap untuk digunakan untuk melatih model klasifikasi aksara Sunda.

3.2. Model

1. *Train Model*

Pada tahap pelatihan model, penelitian ini mengadopsi pendekatan *transfer learning* dengan menggunakan arsitektur CNN VGG-16, yang sebelumnya telah dilatih menggunakan dataset ImageNet. Perancangan struktur model CNN dapat dilihat pada Tabel 2.

Tabel 2. Struktur Model CNN

Lapisan	**Konfigurasi Lapisan**
InputLayer	*input* (224x224x3)
VGG-16 Base Model	*pre-trained*, 3x3 *kernel*, *ReLU*
GlobalAveragePooling2D	-
Flatten	-
Dense (Fully Connected)	512 Neuron, *ReLU*
Dropout	20%
Dense (Output Layer)	30 Neuron, *Softmax*

Setelah model dirancang, proses pelatihan model dilakukan menggunakan *data training* dan *validation*. Pelatihan model berlangsung selama 10 *epoch* dan menerapkan optimasi Adam dengan *learning rate* 0.001. Perbandingan dari proses pelatihan dapat dilihat pada Tabel 3.

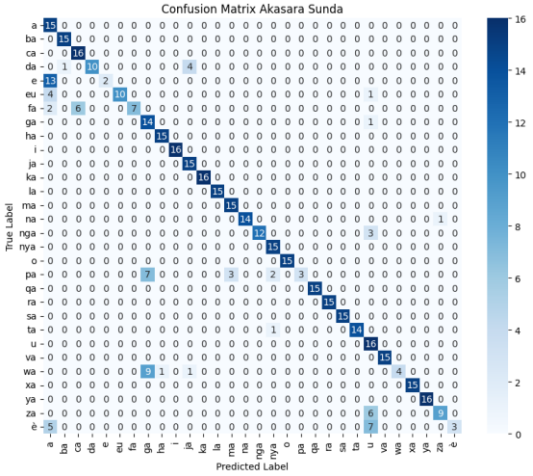
Tabel 3. Hasil *Training* Model

Epoch	***Training***		***Validation***	
	Accuracy	***Loss***	***Accuracy***	***Loss***
1	0.3419	2.6005	0.8540	0.8003
2	0.8957	0.6124	0.9313	0.3376
3	0.9634	0.2495	0.9627	0.1936
4	0.9811	0.1359	0.9713	0.1348
5	0.9835	0.0963	0.9727	0.1065
6	0.9898	0.0643	0.9653	0.1214
7	0.9889	0.0542	0.9800	0.0781
8	0.9935	0.0385	0.9827	0.0656
9	0.9947	0.0328	0.9920	0.0489
10	0.9941	0.0315	0.9940	0.0304

Hasil pelatihan model CNN VGG-16 menunjukkan peningkatan kinerja yang signifikan seiring bertambahnya jumlah *epoch*, dengan akurasi pelatihan meningkat dari 34.19% pada *epoch* pertama menjadi 99.41% pada *epoch* kesepuluh. Akurasi validasi juga menunjukkan peningkatan yang konsisten, mencapai 99.40% pada akhir pelatihan. Penurunan nilai *loss* pada data validasi menunjukkan bahwa model berhasil mengurangi kesalahan prediksi dan mencapai generalisasi yang baik, sehingga dapat diandalkan untuk pengenalan aksara Sunda.

2. *Evaluate Model*

Pada proses ini data testing digunakan pada model akhir untuk memberikan gambaran yang lebih jelas tentang keakuratan dan kelemahan model dalam mengenali aksara-aksara tersebut. Data testing ini diperoleh dari mahasiswa Institut Teknologi Garut yang secara *random* dibuat dari aplikasi menggambar. Proses evaluasi model pada penelitian ini menggunakan *confusion matrix* disajikan pada Gambar 5.



Gambar 5. Hasil Evaluasi *Confusion Matrix*

Gambar 5 menunjukkan kinerja model klasifikasi huruf aksara Sunda, dimana sebagian besar prediksi model akurat dengan nilai yang tinggi pada diagonal utama, yang merepresentasikan jumlah prediksi yang benar untuk setiap kelas. Namun, model ini masih menunjukkan kelemahan dalam membedakan beberapa huruf tertentu, seperti huruf `e` yang sering kali diprediksi sebagai `a`, serta huruf `o` yang cenderung salah diklasifikasikan sebagai `pa`. Kesalahan-kesalahan ini mengindikasikan bahwa ada kemiripan atau kesalahan representasi fitur antara beberapa huruf yang mempengaruhi akurasi model. Setelah *confusion matrix* pada model aksara sunda didapatkan kemudian dilakukan perhitungan evaluasi perhitungan seperti *precision*, *recall*, *f1-score* dan *accuracy*. Dengan menggunakan persamaan 1, 2, 3, dan 4. Hasil dari perhitungan tersebut disajikan dalam Tabel 4 sebagai *classification report*.

Tabel 4. Classification Report

Label	***Precision***	***Recall***	***F1-score***	***Accuracy***
a	0,38	1,00	0,56	1,00
ba	0,94	1,00	0,97	1,00
ca	0,73	1,00	0,84	1,00
da	1,00	0,67	0,80	0,67
e	1,00	0,13	0,24	0,13
eu	1,00	0,67	0,80	0,67
fa	1,00	0,47	0,64	0,47
ga	0,47	0,93	0,62	0,93
ha	0,94	1,00	0,97	1,00
i	1,00	1,00	1,00	1,00
ja	0,75	1,00	0,86	1,00
ka	1,00	1,00	1,00	1,00
la	1,00	1,00	1,00	1,00
ma	0,83	1,00	0,91	1,00
na	1,00	0,93	0,97	0,93
nga	1,00	0,80	0,89	0,80
nya	0,83	1,00	0,91	1,00
o	1,00	1,00	1,00	1,00
pa	1,00	0,20	0,33	0,20
qa	1,00	1,00	1,00	1,00
ra	1,00	1,00	1,00	1,00
sa	1,00	1,00	1,00	1,00
ta	1,00	0,93	0,97	0,93
u	0,47	1,00	0,64	1,00
va	1,00	1,00	1,00	1,00

Label	*Precision*	*Recall*	*F1-score*	*Accuracy*
wa	1,00	0,27	0,42	0,27
xa	1,00	1,00	1,00	1,00
ya	1,00	1,00	1,00	1,00
za	0,90	0,60	0,72	0,60
é	1,00	0,20	0,33	0,20

Hasil pelatihan model yang ditampilkan pada Tabel 4 menunjukkan variasi performa di antara berbagai kelas yang diklasifikasikan. Beberapa kelas seperti "*a*", "*ba*", "*ca*", dan "*ha*" menunjukkan tingkat akurasi yang sangat baik, dengan nilai F1-score yang tinggi, menandakan bahwa model mampu mengenali pola-pola dalam kelas-kelas tersebut dengan baik. Namun, terdapat pula beberapa kelas seperti "*e*", "*pa*" dan "*é*" yang memiliki nilai F1-score yang lebih rendah, menunjukkan adanya tantangan dalam membedakan pola-pola tertentu dalam beberapa kelas.

Meskipun terdapat perbedaan performa di antara kelas-kelas tersebut, model secara keseluruhan telah menunjukkan kemampuannya untuk mengenali mayoritas kelas dengan tingkat akurasi yang memadai. Rata-rata akurasi yang dicapai oleh model adalah 83%, yang menunjukkan bahwa model ini memiliki potensi yang baik dalam melakukan klasifikasi aksara Sunda. Hasil ini dapat digunakan sebagai dasar untuk melakukan optimasi lebih lanjut pada model, dengan fokus pada kelas-kelas yang memiliki performa lebih rendah, untuk meningkatkan hasil klasifikasi secara keseluruhan.

3.2. *Production System*

Tahap *production system* dimulai setelah evaluasi model selesai, dengan fokus pada proses *deployment* model CNN VGG-16 ke dalam prototipe aplikasi berbasis *mobile*. Implementasi model *machine learning* ke dalam aplikasi mobile dilakukan dengan menggunakan *framework* Flutter, yang memungkinkan pengembangan aplikasi lintas platform untuk Android dan iOS. Flutter menyediakan lingkungan pengembangan yang efisien dan fleksibel, serta dukungan untuk integrasi dengan berbagai pustaka dan alat untuk menambahkan fungsionalitas *machine learning* ke aplikasi. Proses implementasi model dilakukan melalui beberapa langkah utama:

1. Pemilihan *Framework* dan Konversi Model

Model *machine learning* dibangun menggunakan TensorFlow, yang dipilih karena kemampuannya dalam menangani tugas-tugas *deep learning* dan fleksibilitasnya dalam konversi model. Setelah model dikembangkan, model dikonversi ke format TensorFlow Lite (.tflite) menggunakan *TensorFlow Lite Converter*. Konversi ini mengoptimalkan model untuk berjalan pada perangkat *mobile* dengan mengurangi ukuran model dan meningkatkan efisiensi inferensi. Model yang telah dibuat dan dikonversi dapat diakses pada tautan berikut github.

2. Integrasi dengan Flutter:

Model yang telah dikonversi menjadi format .tflite, kemudian diintegrasikan ke dalam *prototype* aplikasi menggunakan *framework* flutter. Flutter memungkinkan pengembangan aplikasi dengan antarmuka pengguna yang responsif dan *native-like*, serta mendukung integrasi dengan pustaka machine *learning* seperti *flutter_tflite*. Dengan menggunakan pustaka ini, model .tflite dapat diakses dan digunakan dalam aplikasi untuk melakukan prediksi atau inferensi berdasarkan *input* dari pengguna. *Prototype* aplikasi yang telah dibuat dapat diakses pada tautan github.

3. Pengujian Aplikasi

Pada tahap ini, peneliti menguji sistem *prototype* aplikasi yang telah dibuat untuk memastikan bahwa sistem berfungsi sesuai harapan dan memenuhi kebutuhan pengguna. Pada tahapan ini, dilakukan percobaan kepada beberapa mahasiswa terhadap sistem rekomendasi yang telah dibuat. Hasil pengujian ditampilkan pada Gambar 6.

Gambar 6. Hasil Pengujian *prototype* aplikasi

Detail Hasil pengujian dapat dilihat pada Tabel 6.

Tabel 5. Hasil Training Model

No	Fitur Uji	Skenario Uji	Hasil yang diharapkan	Keterangan
1	Huruf a	Pengguna Menggambar huruf a	Menampilkan hasil klasifikasi huruf a dan nilai *confidence*	Diterima
2	Huruf é	Pengguna Menggambar huruf é	Menampilkan hasil klasifikasi huruf é dan nilai *confidence*	Diterima
3	Huruf i	Pengguna Menggambar huruf i	Menampilkan hasil klasifikasi huruf i dan nilai *confidence*	Diterima
4	Huruf o	Pengguna Menggambar huruf o	Menampilkan hasil klasifikasi huruf o dan nilai *confidence*	Diterima

No	Fitur Uji	Skenario Uji	Hasil yang diharapkan	Keterangan
5	Huruf u	Pengguna Menggambar huruf u	Menampilkan hasil klasifikasi huruf u dan nilai *confidence*	Diterima
6	Huruf e	Pengguna Menggambar huruf e	Menampilkan hasil klasifikasi huruf e dan nilai *confidence*	Diterima
7	Huruf eu	Pengguna Menggambar huruf eu	Menampilkan hasil klasifikasi huruf eu dan nilai *confidence*	Diterima
8	Huruf ka	Pengguna Menggambar huruf ka	Menampilkan hasil klasifikasi huruf ka dan nilai *confidence*	Diterima
9	Huruf ga	Pengguna Menggambar huruf ga	Menampilkan hasil klasifikasi huruf ga dan nilai *confidence*	Diterima
10	Huruf nga	Pengguna Menggambar huruf nga	Menampilkan hasil klasifikasi huruf nga dan nilai *confidence*	Diterima
11	Huruf ca	Pengguna Menggambar huruf ca	Menampilkan hasil klasifikasi huruf ca dan nilai *confidence*	Diterima
12	Huruf ja	Pengguna Menggambar huruf a	Menampilkan hasil klasifikasi huruf ja dan nilai *confidence*	Diterima
13	Huruf nya	Pengguna Menggambar huruf nya	Menampilkan hasil klasifikasi huruf nya dan nilai *confidence*	Diterima
14	Huruf ta	Pengguna Menggambar huruf ta	Menampilkan hasil klasifikasi huruf ta dan nilai *confidence*	Diterima
15	Huruf da	Pengguna Menggambar huruf da	Menampilkan hasil klasifikasi huruf da dan nilai *confidence*	Diterima
16	Huruf na	Pengguna Menggambar huruf na	Menampilkan hasil klasifikasi huruf na dan nilai *confidence*	Diterima
17	Huruf pa	Pengguna Menggambar huruf pa	Menampilkan hasil klasifikasi huruf pa dan nilai *confidence*	Diterima
18	Huruf ba	Pengguna Menggambar huruf ba	Menampilkan hasil klasifikasi huruf ba dan nilai *confidence*	Diterima
19	Huruf ma	Pengguna Menggambar huruf ma	Menampilkan hasil klasifikasi huruf ma dan nilai *confidence*	Diterima
20	Huruf ya	Pengguna Menggambar huruf ya	Menampilkan hasil klasifikasi huruf ya dan nilai *confidence*	Diterima
21	Huruf ra	Pengguna Menggambar huruf ra	Menampilkan hasil klasifikasi huruf ra dan nilai *confidence*	Diterima
22	Huruf la	Pengguna Menggambar huruf la	Menampilkan hasil klasifikasi huruf la dan nilai *confidence*	Diterima
23	Huruf wa	Pengguna Menggambar huruf wa	Menampilkan hasil klasifikasi huruf wa dan nilai *confidence*	Diterima
24	Huruf sa	Pengguna Menggambar huruf sa	Menampilkan hasil klasifikasi huruf sa dan nilai *confidence*	Diterima
25	Huruf ha	Pengguna Menggambar huruf ha	Menampilkan hasil klasifikasi huruf ha dan nilai *confidence*	Diterima
26	Huruf fa	Pengguna Menggambar huruf fa	Menampilkan hasil klasifikasi huruf fa dan nilai *confidence*	Diterima
27	Huruf va	Pengguna Menggambar huruf va	Menampilkan hasil klasifikasi huruf va dan nilai *confidence*	Diterima
28	Huruf qa	Pengguna Menggambar huruf qa	Menampilkan hasil klasifikasi huruf qa dan nilai *confidence*	Diterima
29	Huruf xa	Pengguna Menggambar huruf xa	Menampilkan hasil klasifikasi huruf xa dan nilai *confidence*	Diterima
30	Huruf za	Pengguna Menggambar huruf za	Menampilkan hasil klasifikasi huruf za dan nilai *confidence*	Diterima
31	Huruf secara acak	Pengguna Menggambar huruf acak	Menampilkan hasil klasifikasi huruf acak dan nilai *confidence*	Menampilkan hasil dari huruf Sunda

4. KESIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan model CNN pengenalan aksara Sunda pada perangkat berbasis *mobile.* Model yang dibangun menunjukkan performa dengan akurasi sebesar 99% pada data pelatihan dan 83% pada data pengujian. Meskipun model mampu mengenali mayoritas kelas aksara dengan baik, diperlukan optimasi lebih lanjut pada kelas-kelas tertentu yang memiliki tingkat akurasi rendah agar hasil klasifikasi lebih merata dan andal. Penelitian ini memberikan kontribusi terhadap pelestarian aksara daerah melalui pengembangan sistem pengenalan tulisan tangan aksara Sunda berbasis CNN yang diintegrasikan ke dalam aplikasi *mobile*. Selain itu, penelitian ini turut menyediakan dataset tulisan tangan aksara Sunda sebagai sumber data pelatihan dan pengujian, serta menghasilkan model awal yang dapat digunakan kembali dalam pengembangan lebih lanjut pada studi serupa berbasis kecerdasan buatan.

Meskipun telah menunjukkan hasil yang menjanjikan, model yang dikembangkan masih memiliki keterbatasan dalam membedakan antara karakter aksara Sunda dengan gambar atau teks non-aksara. Beberapa *input* non-aksara tetap terdeteksi sebagai salah satu kelas aksara Sunda. Oleh karena itu, pengembangan selanjutnya disarankan untuk

menambahkan kelas khusus yang berfungsi mendeteksi *input* tidak dikenal (*unknown class*), sehingga model dapat membedakan antara aksara yang *valid* dan *input* yang tidak relevan, serta meningkatkan akurasi dan kinerja sistem secara keseluruhan.

DAFTAR PUSTAKA

AMAZON, 2024. *Well-Architected machine learning lifecycle*. [online] aws. Available at: <https://docs.aws.amazon.com/wellarchitected/latest/machine-learning-lens/well-architected-machine-learning-lifecycle.html>.

ANGGRAENY, F.T., VIA, Y.V. AND MUMPUNI, R., 2023. Image preprocessing analysis in handwritten Javanese character recognition. *Bulletin of Electrical Engineering and Informatics*, [online] 12(2), pp.860–867. https://doi.org/10.11591/eei.v12i2.4172.

BPS, 2022. *Statistik Telekomunikasi Indonesia 2022*. [online] bps.go.id. Available at: <https://www.bps.go.id/id/publication/2023/08/31/131385d0253c6aae7c7a59fa/statistik-telekomunikasi-indonesia-2022.html>.

DEWA, C.K., FADHILAH, A.L. AND AFIAHAYATI, A., 2018. Convolutional Neural Networks for Handwritten Javanese Character Recognition. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, [online] 12(1), p.83. https://doi.org/10.22146/ijccs.31144.

HABIBAH, A., SHOLIHANINGTIAS, D.N. AND PRATIWI, N.K., 2020. Aplikasi Media Pembelajaran Aksara Sunda Berbasis Android. *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, [online] 4(3), p.256. https://doi.org/10.30998/string.v4i3.4850.

KEMDIKBUD, 2022. *Mendikbudristek Luncurkan Merdeka Belajar 17: Revitalisasi Bahasa Daerah*. [online] kemdikbud.go.id. Available at: <https://www.kemdikbud.go.id/main/blog/2022/02/mendikbudristek-luncurkan-merdeka-belajar-17-revitalisasi-bahasa-daerah>.

KURNIADI, D., NURAENI, F., RAHARJA, I.T. AND MULYANI, A., 2022. Perancangan Aplikasi Text To Speech Dalam Bahasa Indonesia Menggunakan Firebase Machine Learning Kit Berbasis Android. *Jurnal Teknologi Informasi dan Ilmu Komputer*, [online] 9(6), pp.1281–1288. https://doi.org/10.25126/jtiik.2022965985.

LORENTIUS, C., ADIPRANATA, R. AND TJONDROWIGUNO, A.N., 2021. Pengenalan Aksara Jawa dengan Menggunakan Metode Convolutional Neural Network. *Jurnal Infra*. [online] Available at: <https://publication.petra.ac.id/index.php/teknik-informatika/article/view/8075>.

MASKEY, M., MOLTHAN, A., HAIN, C., RAMACHANDRAN, R., GURUNG, I., FREITAG, B., MILLER, J.J., RAMASUBRAMANIAN, M., BOLLINGER, D., MESTRE, R. AND CECIL, D., 2019. Machine Learning Lifecycle for Earth Science Application: A Practical Insight into Production Deployment. In: *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. pp.10043–10046. https://doi.org/10.1109/IGARSS.2019.8899031.

PRAMESWARI, M.A., DWI SULISTIYO, M. AND IHSAN, A.F., 2023. Classification of Handwritten Sundanese Script via Transfer Learning on CNN-Based Architectures. In: *2023 3rd International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS)*. [online] IEEE. pp.401–406. https://doi.org/10.1109/ICE3IS59323.2023.10335382.

PUN, R., 2024. *UNESCO: Setiap Dua Minggu, Satu Bahasa Daerah Punah di Dunia*. [online] portal jabarprovgoid. Available at: <https://jabarprov.go.id/berita/unesco-setiap-dua-minggu-satu-bahasa-daerah-punah-di-dunia-12944>.

PURNAMA, A., BAHRI, S., GUNAWAN, G., HIDAYATULLOH, T. AND SUHADA, S., 2022. Implementation of Deep Learning for Handwriting Imagery of Sundanese Script Using Convolutional Neural Network Algorithm (CNN). *ILKOM Jurnal Ilmiah*, [online] 14(1), pp.10–16. https://doi.org/10.33096/ilkom.v14i1.989.10-16.

RAHMAWATI, S.N., HIDAYAT, E.W. AND MUBAROK, H., 2021. Implementasi Deep Learning Pada Pengenalan Aksara Sunda Menggunakan Metode Convolutional Neural Network. *Journal Insert*. [online] Available at: <ttps://ejournal.undiksha.ac.id/index.php/insert/article/view/37405>.

SETIAWAN, A., 2020. *Digitalisasi Aksara Nusantara agar Lestari*. [online] indonesia.go.id. Available at: <https://indonesia.go.id/ragam/budaya/kebudayaan/digitalisasi-aksara-nusantara-agar-lestari>.

THARWAT, A., 2018. Classification assessment methods. *Applied Computing and Informatics*, [online] 17(1), pp.168–192. https://doi.org/10.1016/j.aci.2018.08.003.

TONDO, F.H., 2015. Kepunahan Bahasa-Bahasa

Daerah: Faktor Penyebab Dan Implikasi Etnolinguistis. *Jurnal Masyarakat dan Budaya (JMB)*, [online] 11. Available at: <https://jmb.lipi.go.id/jmb/article/view/245>.

Halaman ini sengaja dikosongkan