

Pengembangan Sistem Pengenalan Karakter Aksara Suku Simalungun Berbasis Android

Theresia Seftiani Girsang^{a1}, I Dewa Made Bayu Atmaja Darmawan^{a2}, Ngurah Agus Sanjaya ER^{a3},
AAIN Eka Karyawati^{a4}, I Putu Gede Hendra Suputra^{a5}, Cokorda Rai Adi Pramatha^{a6}

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana
Badung, Bali, Indonesia

¹theresiagirsang89@gmail.com

²dewabayu@unud.ac.id

³agus_sanjaya@unud.ac.id

⁴eka.karyawati@unud.ac.id

⁵hendra.suputra@unud.ac.id

⁶cokorda@unud.ac.id

Abstract

Simalungun script is the script used by the Simalungun people to communicate with each other in their time. But over time, this character is rarely used. Therefore, the Simalungun script needs special attention because it is already threatened with extinction due to limited data and information. To overcome this, it is necessary to use the role of information technology. In this study, a system was built that can classify and introduce the Simalungun tribal characters using the Android-based Convolutional Neural Network (CNN) method. In addition, in this study, CNN MobileNetV2 and TensorFlow Lite architectures were used for deploying android needs. From the results of the training using the MobileNetV2 architecture by testing 29 characters, the accuracy results are 81% and the test results are 82%. In testing the feasibility of the application, the author uses the concept of usability testing by involving 15 respondents and giving a percentage of 78%.

Keywords: *Simalungun Script, Image Classification, Convolutional Neural Network, Android, Usability Testing*

1. Pendahuluan

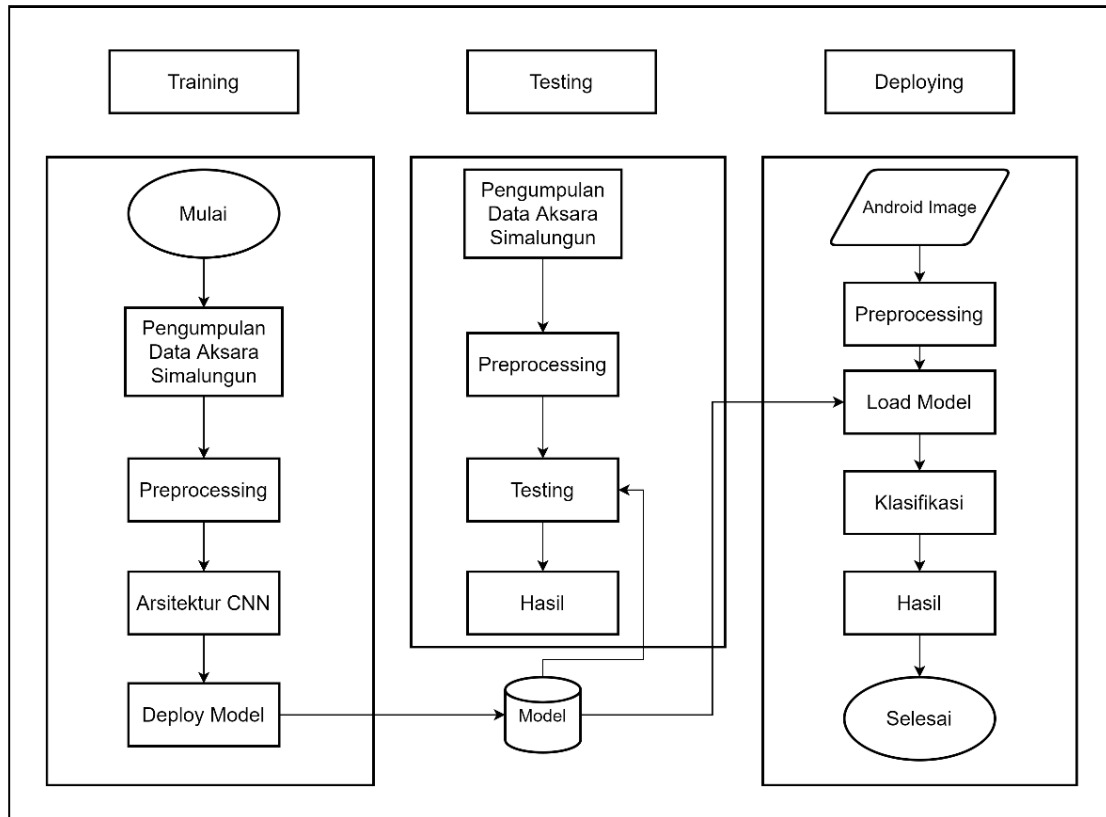
Aksara Simalungun merupakan aksara yang digunakan oleh masyarakat Simalungun untuk saling berkomunikasi. Aksara Simalungun terdiri dari Induk Surat dan Anak Surat. Induk Surat merupakan sembilan belas huruf aksara simalungun. Anak Surat merupakan tanda-tanda atau diakritik yang dapat mengubah nilai atau bunyi dari Induk Surat. Aksara Simalungun ini adalah salah satu rumpun dari Aksara Batak yang memerlukan perhatian khusus [1]. Hal ini dikarenakan perkembangan teknologi informasi yang sangat pesat, sehingga perlahan-lahan aksara ini mulai ditinggalkan. Aksara Simalungun merupakan warisan budaya yang pantas untuk tetap dijaga dan dilestarikan. Untuk mengatasi permasalahan tersebut maka perlu adanya peran teknologi informasi yang digunakan. Machine Learning dan Deep Learning merupakan teknologi kecerdasan buatan yang populer saat ini dan dapat digunakan untuk mengatasi permasalahan tersebut [2]. Convolutional Neural Network merupakan salah satu metode Deep Learning dari pengembangan Multi-Layer Perceptron yang didesain untuk mengolah data dua dimensi, yaitu gambar dan suara.

Penelitian lainnya dilakukan oleh Susilo, dkk pada tahun 2017, dimana pada penelitian tersebut menggunakan metode CNN dalam pengenalan pola karakter bahasa jepang hiragana. Pada penelitian tersebut arsitektur jaringan CNN menghasilkan akurasi yang baik mencapai 96,2 %. Metode *Convolutional Neural Network* dianggap baik dan mampu dalam mengenali berbagai bentuk citra. Penelitian diatas menjadi dasar usulan peneliti untuk membuat suatu teknologi yang dapat digunakan dalam pengenalan karakter Aksara Simalungun [3]. Berdasarkan latar belakang permasalahan di atas, maka penelitian kali ini akan membangun sebuah sistem yang dapat mendeteksi dan mengenalkan aksara suku Simalungun dengan lebih baik. Penelitian ini akan dilakukan dengan menggunakan metode *Convolutional Neural Network* (CNN). Supaya dapat membantu melestarikan warisan budaya

supaya tidak punah. Peneliti menganggap hal ini penting untuk dilakukan. Oleh karena itu peneliti akan membuat suatu sistem berbasis *android* sederhana sebagai aplikasi pengenalan karakter aksara Simalungun.

2. Metode Penelitian

Pada Gambar 1. terdapat tiga proses yang akan dilakukan dalam penelitian ini. Proses pertama yaitu *training* (pelatihan), proses kedua *testing* (pengujian) dan proses yang terakhir *deploying* (penyematan).

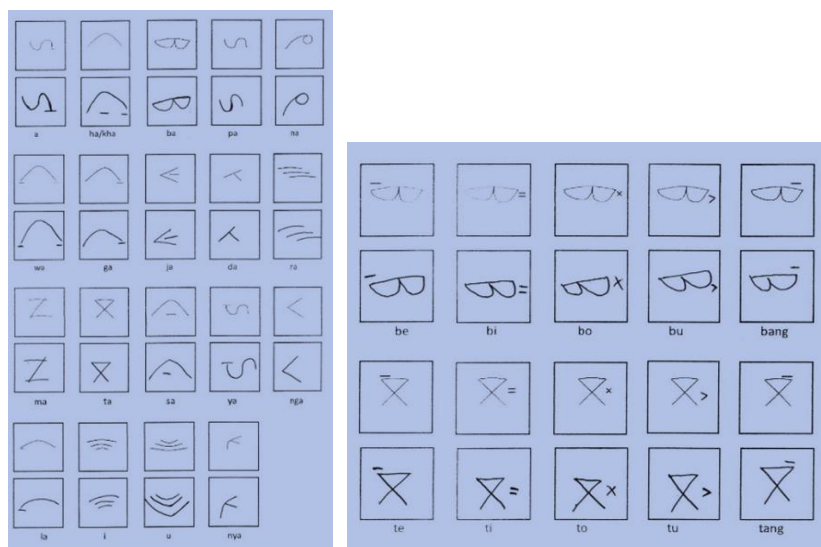


Gambar 1. Alur Sistem Pengenalan Karakter Aksara Suku Simalungun

Pada tahap *training* (pelatihan) terdiri dari pengumpulan data Aksara Simalungun yang diambil langsung dari beberapa responden terkait, proses *preprocessing* data yang siap diolah ke proses klasifikasi, proses klasifikasi menggunakan arsitektur CNN dan menghasilkan model yang siap untuk di *testing*. Kemudian pada tahap *testing* (pengujian) akan dilakukan pengujian terhadap model yang sudah dilatih pada tahap *training*. Jika model yang dihasilkan sudah memberikan akurasi yang cukup baik pada saat *testing*, maka akan dilanjutkan dengan proses *deploying* (penyematan). Tahap *deploying* (penyematan) adalah tahap yang berisi proses konversi klasifikasi CNN pada aplikasi android. Dimana pada tahap ini akan menghasilkan aplikasi yang dapat memberikan prediksi deskripsi karakter aksara.

2.1 Pengumpulan Data

Data primer adalah data yang diperoleh langsung dengan cara mengambil sendiri tanpa adanya perantara. Sedangkan data sekunder adalah data yang sebelumnya sudah dikumpulkan oleh orang lain, sehingga peneliti boleh meminta data yang sudah ada tersebut kepada instansi atau organisasi [4]. Data yang digunakan dalam penelitian ini adalah data primer. Pengumpulan data akan dilakukan dengan memberikan sampel data kepada 33 orang responden. Kemudian responden akan menulis sesuai dengan sampel data yang diberikan. Responden yang dituju adalah siswa/siswi tingkat sekolah dasar kelas VI (enam). Data tersebut merupakan scan citra karakter Aksara Simalungun berjumlah 29 karakter, yang mana terdiri dari 19 karakter induk huruf dan dengan tambahan 10 karakter anak huruf. Berikut dapat dilihat 29 karakter Aksara Simalungun. Contoh data dapat dilihat pada Gambar 2.



Gambar 2. (a) Induk Huruf, (b) Anak Huruf

Gambar 2. menunjukkan data yang akan dikumpulkan dari seluruh responden. Data yang berhasil dikumpulkan terdiri dari 33 variasi jenis tulisan tangan yang berbeda. Setiap responden menuliskan sebanyak lima kali penulisan aksara sesuai sampel yang diberikan. Oleh karena itu, maka total keseluruhan data aksara yang terkumpul adalah sebanyak 4.640 karakter. Kemudian untuk validasi daripada tulisan karakter Aksara Simalungun ini secara detail diperiksa oleh guru aksara yang bersangkutan.

2.2 Pra Proses Citra Input

Pra proses citra input ini dilakukan pada saat implementasi dalam penulisan kode program (*coding*) dengan bahasa pemrograman Python. Tujuan pra proses input adalah untuk menyesuaikan format citra supaya pada saat citra memasuki arsitektur CNN dapat terbaca dengan baik. Tahapannya sebagai berikut:

1. Mengubah data citra (.png) ke dalam bentuk array.
2. Mendapatkan fitur (X) dan label (y) citra, hasilnya menjadi *variable* berupa X_{train} , y_{train} , X_{test} , y_{test} .
3. Normalisasi (*feature scaling*) untuk X_{train} dan X_{test} , yaitu mengubah rentang nilai 0-225 menjadi 0-1.
4. *One-hot-encoding* untuk y_{train} dan y_{test} , yaitu mengubah setiap nilai di dalam kolom menjadi kolom baru dan mengisinya dengan nilai biner yaitu 0 dan 1.
5. Hasil akhir berupa X_{train_norm} (X_{train} yang telah di normalisasi), y_{train_encode} (y_{train} yang telah di *encoding*), X_{test_norm} (X_{test} yang telah di normalisasi), y_{test_encode} (y_{test} yang telah di *encoding*).
6. Selanjutnya variabel yang merepresentasikan citra tersebut sudah siap diolah kedalam algoritma CNN untuk dilakukan proses *training* kemudian *testing*.

2.3 Arsitektur CNN: MobileNetV2

Dalam penelitian ini, peneliti memilih arsitektur CNN MobileNetV2. Dengan alasan selain memang score akurasi cukup tinggi, juga yang menjadi keunggulan utamanya adalah jumlah training parameters yang kecil dibandingkan dengan arsitektur CNN lainnya. Sehingga kebutuhan akan komputasinya lebih ringan. Oleh karena itu, jika model tersebut akan di deploy ke dalam sebuah real app, misalnya aplikasi android ataupun aplikasi berbasis website akan ringan dan berukuran kecil [5]. Pada penelitian ini akan digunakan tipe arsitektur CNN *MobileNetV2* dengan tambahan *dropout*. Arsitektur CNN akan dilatih (*training* dan *validation*) terhadap dataset yang sudah tersedia. Selain itu, juga akan diterapkan *hyperparameter* berupa *learning rate* dengan nilai 0.0003 yang akan menghasilkan model CNN. Model tersebut akan diujikan (*testing*) terhadap data test untuk kemudian dievaluasi hasilnya, jika hasilnya sudah cukup baik maka yang terakhir adalah men-deploynya ke aplikasi siap pakai.

2.4 Perancangan Sistem

Dalam penelitian ini sistem yang akan dikerjakan merupakan sistem berbasis android dengan menggunakan bahasa pemrograman Kotlin. Sistem dibangun menggunakan model tensorflow lite, dimana model ini sudah dikonversi dari bahasa pemrograman Python 3.9.0 dengan tambahan library yang sudah tersedia pada Google Collaboratory. Tujuan utama pembuatan sistem ini adalah untuk

memudahkan pengenalan karakter Aksara Simalungun menggunakan metode Convolutional Neural Network. Hasil dari proses ini adalah berupa deskripsi karakter aksara.

Sistem dirancang untuk mengklasifikasi masing-masing karakter dari Aksara Simalungun berdasarkan citra input menggunakan teknologi Deep Learning metode Convolutional Neural Network. Karakter Aksara Simalungun yang akan diklasifikasikan terdiri dari 19 induk huruf dan 10 anak huruf. Secara keseluruhan, proses klasifikasi citra Aksara Simalungun ini melalui beberapa tahapan. Pertama, dilakukan pengumpulan dataset citra Aksara Simalungun secara langsung. Kedua, akan dilakukan praproses dataset agar data citra sesuai dengan kriteria standar saat memasuki arsitektur CNN.

Praproses dataset ini dilakukan dengan harapan menghasilkan model CNN yang baik. Tahap ketiga, dataset akan dipecah menjadi data train dan data test, dimana data train nantinya akan digunakan untuk melatih model dan data test nantinya akan digunakan untuk menguji model. Untuk mengevaluasi model digunakan K-Fold Cross Validation. Kemudian untuk mengevaluasi model klasifikasi digunakan Confusion Matrix. Setelah model selesai dievaluasi, maka akan dikonversi menggunakan tensorflow lite dan akan menghasilkan model.tflite yang selanjutnya dilakukan deploy ke Android Studio. Proses yang selanjutnya berjalan adalah implementasi aplikasi menggunakan model.tflite pada Android Studio.

2.5 Evaluasi Sistem

Untuk evaluasi performansi hasil klasifikasi dalam penelitian ini, maka digunakan teknik confusion matrix dan classification report. Dalam confusion matrix akan diperlihatkan tabel yang berisi jumlah prediksi benar dan salah dari model klasifikasi. Dalam confusion matrix hasil yang ditampilkan sudah cukup detail, namun jika ditambah dengan classification report akan semakin menambah pemahaman hasil klasifikasi dan dapat mengukur kinerja model.

Pengujian menggunakan *usability testing* yang merupakan tingkat suatu produk yang digunakan oleh user untuk mencapai target yang ditetapkan, seperti efektifitas, efisiensi dan mencapai kepuasan pengguna pada suatu aplikasi [6]. Kriteria yang dapat diukur dalam pengujian *usability* ini yaitu sebagai berikut:

- Kemudahan (*learnability*), mendefinisikan seberapa cepat user dalam menggunakan sistem dan kemudahan dalam penggunaan aplikasi.
- Efisiensi (*efficiency*), mendefinisikan sebagai sumber daya yang dikeluarkan guna mencapai ketepatan dan kelengkapan tujuan.
- Mudah diingat (*memorability*), mendefinisikan bagaimana kemampuan user dapat mempertahankan pengetahuannya setelah jangka waktu tertentu.
- Kepuasan (*satisfaction*), mendefinisikan sikap positif user terhadap aplikasi.

Dari keempat aspek *usability* tersebut, diambil 15 responden dengan kriteria responden seperti berikut:

- Responden berasal dari daerah Simalungun.
- Responden berusia 18 – 30 tahun.
- Responden dengan jenis kelamin laki-laki dan perempuan.
- Responden yang mengerti dan memahami tentang Aksara Simalungun.
- Responden yang memahami teknologi dan informasi.

Ketika seluruh kriteria responden terpenuhi, maka 15 responden tersebut akan diarahkan untuk mengikuti sesi latihan dalam menggunakan aplikasi. Peneliti akan memberikan panduan dalam menggunakan aplikasi kepada seluruh responden. Hal ini diberikan supaya responden dapat lebih mengerti dalam menggunakan aplikasi dan dapat mengisi kuesioner yang telah disediakan dengan penilaian yang valid.

3. Hasil dan Pembahasan

3.1 Implementasi CNN

Proses pelatihan (training) dilakukan dengan menggunakan dataset yang sudah diolah pada tahap pra proses sebelumnya. Menggunakan metode CNN dengan arsitektur MobileNetV2 + Dropout. Selain itu, proses training dilakukan menggunakan teknik K-Fold Cross Validation dengan nilai K=3, dimana dataset yang digunakan 80% data train (3712) dan 20% data testing (928). Digunakan juga beberapa hyperparameter sebagai berikut:

- Input shape citra : 128x128x3
- Batch size : 32
- Epoch : 20
- Optimizer : Adam
- Learning rate : 0.0003

Berikut adalah hasil *accuracy* dan *loss* dari Fold ke-1, Fold ke-2 dan Fold ke-3.

Tabel 1. Hasil Training dengan K-Fold

No.	Accuracy	Loss	Accuracy	Loss	Accuracy	Loss
-----	----------	------	----------	------	----------	------

	<i>Fold 1</i>	<i>Fold 1</i>	<i>Fold 2</i>	<i>Fold 2</i>	<i>Fold 3</i>	<i>Fold 3</i>
1.	81.6	0.538	80.8	0.522	82.4	0.506

Berdasarkan hasil *training* pada Tabel 1. yang dilakukan maka didapatkan total rata-rata akurasi sebesar 81.6%. Dapat dikatakan bahwa akurasi ini sudah cukup baik dan dapat dilanjutkan ke tahap *testing*.

3.2 Implementasi Android

a. Splash Screen Aplikasi

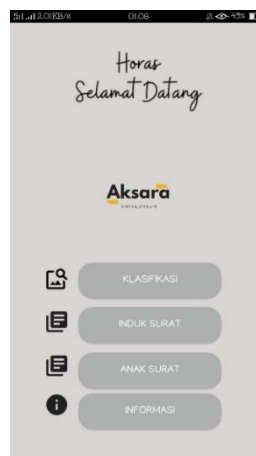
Halaman ini merupakan halaman splashscreen aplikasi. *Splash Screen* muncul pada saat aplikasi mulai dijalankan dengan memperlihatkan logo aplikasi. Saat aplikasi pertama kali dibuka, maka akan ditampilkan logo aplikasi sebagai pengantar ke halaman selanjutnya. Halaman *splash screen* ini akan berdurasi 3 detik, setelah 3 detik berlalu maka akan ditampilkan halaman berikutnya, yaitu halaman utama. Berikut dapat dilihat *splash screen* aplikasi.



Gambar 3. Splash Screen Aplikasi

b. Halaman Utama Aplikasi

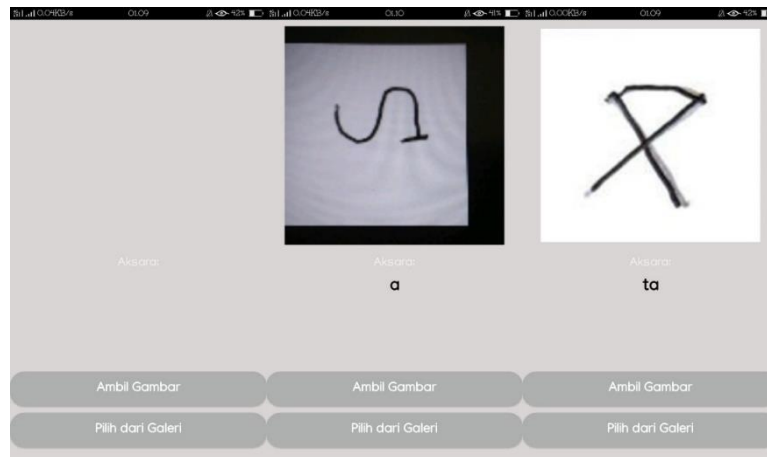
Halaman utama aplikasi, dimana terdapat beberapa menu yang dapat dijalankan. Pada halaman ini juga terdapat tulisan yang merupakan sambutan saat masuk ke dalam halaman utama aplikasi. Terdapat logo aplikasi juga pada halaman ini. Halaman utama tentunya dilengkapi dengan empat menu, yaitu menu klasifikasi, menu induk surat, menu anak surat dan menu informasi. Setiap menu dapat dijalankan dengan menekan *button* masing-masing menu, yang selanjutnya akan diarahkan ke halaman menu yang dipilih.



Gambar 4. Halaman Utama Aplikasi

c. Halaman Klasifikasi

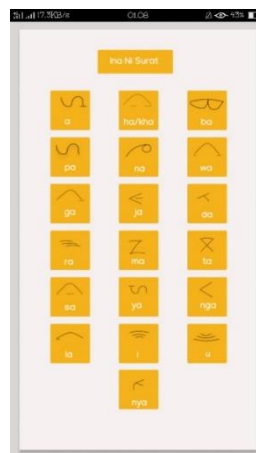
Halaman klasifikasi adalah halaman yang digunakan untuk memproses pengambilan gambar melalui kamera dan galeri, yang selanjutnya akan di prediksi hasil dari masing-masing karakter yang diinput. Pada halaman ini terdapat 2 (dua) pilihan, pengguna dapat memilih button ambil gambar untuk mengambil gambar aksara secara real time sebagai inputan untuk diprediksi. Kemudian terdapat pilihan pilih dari galeri, dimana pengguna dapat memasukkan inputan dari kumpulan foto aksara yang sudah ada atau sudah diambil sebelumnya. Inputan dari kedua pilihan tersebut kemudian akan diprediksi oleh aplikasi, aplikasi akan memberikan output berupa deskripsi aksara.



Gambar 5. Halaman Klasifikasi Aplikasi

d. Halaman Induk Surat

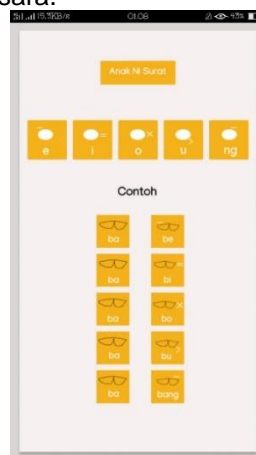
Halaman ini merupakan halaman induk surat pada aplikasi, yang didalamnya berisi seluruh karakter induk surat Aksara Simalungun. Terdapat image dan label ke-23 induk surat Aksara Simalungun pada halaman ini. Halaman ini bertujuan untuk memperlihatkan image dan label induk surat aksara yang benar kepada pengguna.



Gambar 6. Halaman Induk Surat Aplikasi

e. Halaman Anak Surat

Halaman ini merupakan halaman anak surat pada aplikasi, yang didalamnya berisi lima pola/tanda dalam anak surat. Halaman ini bertujuan untuk memperlihatkan image dan label anak surat aksara yang benar kepada pengguna. Dalam hal ini, anak surat bertujuan sebagai tanda/pola yang dapat digunakan untuk mengubah bunyi suatu aksara.



Gambar 7. Halaman Anak Surat Aplikasi

f. Halaman Informasi

Halaman ini merupakan halaman informasi pada aplikasi yang berisi informasi singkat mengenai Aksara Simalungun. Dalam halaman ini juga ditampilkan satu image yang merupakan sejarah Aksara

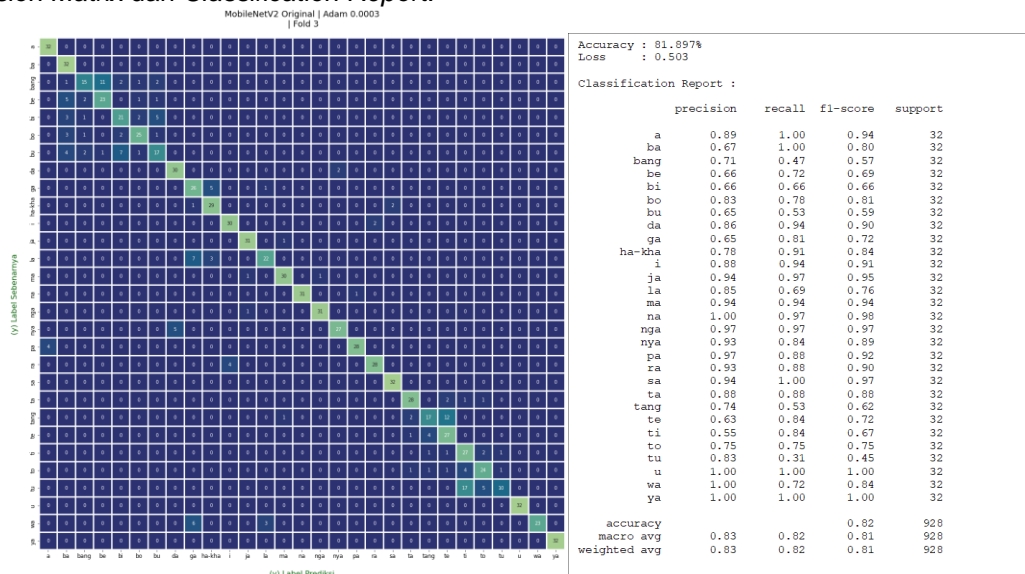
Simalungun, yaitu aksara yang ditulis pada media bambu. Pengguna dapat membaca secara singkat sejarah dari Aksara Simalungun pada halaman informasi.



Gambar 8. Halaman Informasi Aplikasi

3.3 Evaluasi Confusion Matrix dan Classification Report

Dari hasil pelatihan (*training*) dan pengujian (*testing*), dilakukan analisis terhadap model dengan menggunakan *Confusion Matrix* dan *Classification Report*. Berikut ditampilkan untuk detail hasil dari *Confusion Matrix* dan *Classification Report*.



Gambar 9. (a) Confusion Matrix, (b) Classification Report

Berdasarkan informasi pada Gambar 9. *Confusion Matrix* dan *Classification Report* diatas, beberapa hal yang dapat disimpulkan yaitu:

1. Hasil *testing* dengan data *test* menghasilkan paling banyak salah prediksi yaitu pada aksara be, bi, bu, bang, te, ti, tu dan tang. Jika diperhatikan secara seksama kesalahan prediksi beberapa aksara ini disebabkan oleh kemiripan bentuk huruf dan beberapa bentuk huruf yang sama, namun hanya dibedakan oleh simbol atau tanda tertentu.
2. Model *MobileNetV2 + dropout* (Adam 0.0003) dengan total data *test*, secara keseluruhan menghasilkan *score* akurasi dan *f1-score* yang cukup baik.
3. Pengujian dengan *confusion matrix* dan *classification report* menunjukkan akurasi yang didapatkan sebesar 82% dan dengan akurasi seperti ini dapat dikatakan sudah cukup baik.

3.4 Analisis Kepuasan Pengguna

Analisis kepuasan pengguna bertujuan untuk menilai tanggapan dari pengguna sistem, yaitu siapa saja yang penasaran dan mau belajar tentang Aksara Simalungun. Untuk mencapai target yang ditetapkan dengan efektifitas, efisiensi dan mencapai kepuasan pengguna pada suatu aplikasi, maka dibutuhkan pengujian usability yang digunakan oleh pengguna. Evaluasi kepuasan pengguna dilakukan dengan menggunakan data kuesioner yang berasal dari 15 responden yang sudah memenuhi kriteria. Berdasarkan hasil kuesioner yang dikumpulkan maka didapatkan hasil sebagai berikut:

$$\text{Skor maksimum} = 15 (\text{responden}) \times 12 (\text{pertanyaan}) \times 5 (\text{score}) = 900$$

Skor maksimum yang didapat yaitu sebesar 900, maka hasil kuesioner usability ini sudah bisa ditentukan persentasenya sebagai berikut:

$$\frac{702}{900} \times 100\% = 78\%$$

Dapat disimpulkan bahwa hasil dari persentase menunjukkan bahwa keempat aspek usability yang didapatkan dari 15 responden yang langsung menjadi pengguna aplikasi Aksara Simalungun, memberikan penilaian aplikasi sebesar 78%.

4. Kesimpulan

Penelitian ini menghasilkan sebuah sistem pengenalan karakter Aksara Simalungun berdasarkan citra input yang menggunakan teknologi *deep learning* dengan metode *Convolutional Neural Network* (CNN) berbasis android. Berikut beberapa kesimpulan yang dapat diambil dari penelitian ini yaitu:

1. Berdasarkan metode *Convolutional Neural Network* menggunakan arsitektur *MobileNetV2 + Dropout* dengan *hyperparameter* berupa *Adam Optimizer* dan *Learning Rate* 0.0003 di *fold* ke-3. Memberikan *score* akurasi *training* sebesar 81% dan *score* akurasi *testing* sebesar 82%. Dengan *score* akurasi seperti berikut, arsitektur *MobileNetV2* teruji menghasilkan *score* akurasi *testing* yang cukup tinggi.
2. Parameter pelatihan yang digunakan pada penelitian ini seperti *Adam Optimizer*, *Learning Rate* (0.0003) dan *epoch* (20) memberikan pengaruh yang cukup signifikan terhadap *score* akurasi yang dihasilkan oleh metode *Convolutional Neural Network*. Dengan adanya parameter pelatihan ini dapat mengoptimalkan kinerja metode, mempercepat waktu komputasi dan menghasilkan *score* akurasi yang tinggi.
3. Dari hasil pengujian dengan analisis kepuasan pengguna menggunakan skala *likert* pada pengujian *usability* yang berguna untuk mengetahui tingkat kelayakan aplikasi. Peneliti melibatkan 15 orang responden dan pengujian ini dilakukan dengan cara menyebarkan kuesioner yang berisi pertanyaan seputar aplikasi. Hasil dari kuesioner tersebut menentukan kelayakan aplikasi dari 4 aspek *usability* dengan hasil persentase sebesar 78%.

Daftar Pustaka

- [1] Kozok, U. Warisan Leluhur: sastra lama dan aksara Batak (Vol. 17). Kepustakaan Populer Gramedia. 1999.
- [2] Nurhikmat, Triakno. "Implementasi deep learning untuk image classification menggunakan algoritma Convolutional Neural Network (CNN) pada citra wayang golek." (2018).
- [3] Susilo, M. M., Wonohadidjojo. D. M., & Sugianto, N. (2017). Pengenalan Pola Karakter Bahasa Jepang Hiragana Menggunakan 2D Convolutional Neural Network. *J. Inform. dan Sist. Inf. Univ. Ciputra*, 3(02), 28-36.
- [4] Khasanah. Dqlab.id. 07 Juli 2021. <https://dqlab.id/perbedaan-data-primer-dan-data-sekunder>. Diakses pada tanggal 12 Juli 2022.
- [5] Afif. Medium.id. 28 April 2020. <https://medium.com/@hafizhan.aliady/membuat-klasifikasi-gambar-images-menggunakan-keras-tensorflow-tf-keras-dan-python-53f7ae953cea>. Diakses pada 12 Juli 2022.
- [6] Rahadi, Dedi Rianto. "Pengukuran usability sistem menggunakan use questionnaire pada aplikasi android." *JSI: Jurnal Sistem Informasi (E-Journal)* 6.1 (2014).