

RESEARCH ARTICLE

Klasifikasi Aksara Lontara Dari Sulawesi Selatan Menggunakan CNN

Abdul Rahim, Febryanti Sthevanie * and Kurniawan Nur Ramadhani

Fakultas Informatika, Universitas Telkom, Bandung, 40257, Jawa Barat, Indonesia

* Corresponding author: febryantisthevanie@telkomuniversity.ac.id

Abstrak

Sebagai negara kepulauan dengan keberagaman budaya dan bahasa, Indonesia memegang peran penting dalam menyimpan dan merawat warisan budaya. Salah satu warisan tersebut adalah aksara Lontara, sebuah sistem tulisan tradisional yang telah digunakan secara luas di Sulawesi Selatan. Penelitian ini melakukan perbandingan beberapa arsitektur CNN untuk klasifikasi aksara Lontara dalam konteks OCR. Hasil eksperimen menunjukkan bahwa arsitektur CNN yaitu VGG16 mencapai performa terbaik dengan akurasi *training validation* sebesar 97% dan *testing* sebesar 90% dibandingkan dengan arsitektur VGG19, ResNet, dan ResNetV2.

Key words: Aksara Lontara, Citra Digital, OCR, Klasifikasi, CNN.

Pendahuluan

Sebagai negara kepulauan yang kaya akan budaya dan bahasa, Indonesia memiliki peran krusial dalam melestarikan warisan budaya. Salah satu warisan penting tersebut adalah aksara Lontara, sistem tulisan tradisional yang banyak digunakan di Sulawesi Selatan. Aksara Lontara ini merupakan simbol dan memiliki peran signifikan dalam menyimpan catatan sejarah, sastra, dan pengetahuan tradisional masyarakat setempat [1]. Mengingat nilai budaya yang tinggi dan pentingnya pelestarian aksara ini, diperlukan teknologi yang mampu membantu memperkenalkan aksara ini kepada masyarakat luas. Salah satu teknologi tersebut adalah *Optical Character Recognition* (OCR), yang mengubah gambar menjadi teks, sehingga memudahkan pengenalan aksara Lontara oleh Masyarakat. OCR berfungsi untuk memecahkan masalah dalam ekstraksi informasi tulisan dari gambar [2]. Untuk mengenali pola dan mengklasifikasikan model yang telah dilatih, algoritma deep learning seperti *Convolutional Neural Network* (CNN) bisa digunakan. Saat ini, algoritma CNN merupakan solusi dalam pengolahan citra untuk mendeteksi objek pada gambar.

Penelitian menunjukkan bahwa CNN efektif dalam mengklasifikasi aksara Lontara [3]. CNN merupakan pengembangan dari *Multilayer Perceptron* (MLP), dirancang khusus untuk memproses gambar atau data. Dalam *deep learning*, MLP telah dikembangkan lebih lanjut untuk menangani data yang kompleks, menghasilkan model berlapis yang dikenal sebagai CNN. Penggunaan CNN dalam pengenalan citra dapat meningkatkan efisiensi pembelajaran aksara Lontara dengan membangun model yang efektif untuk mengklasifikasikan aksara tersebut. Penelitian ini bertujuan untuk Penelitian ini bertujuan untuk mengklasifikasikan aksara Lontara dan membantu pengenalan serta

pemahaman bentuk setiap karakter aksara. Metode CNN diterapkan dalam penelitian ini untuk mengklasifikasikan aksara Lontara dengan akurat.

Tinjauan Pustaka

Pada penelitian [4] klasifikasi aksara Jawa menggunakan metode *Convolutional Neural Network* (CNN) berhasil mencapai tingkat akurasi sebesar 85%. Hasil menunjukkan kemampuan CNN dalam mengelompokkan aksara Jawa 'Ka' dan 'Nya'. Penelitian [3] menunjukkan bahwa pengenalan tulis tangan dan segmentasi karakter aksara Lontara menggunakan *Convolutional Neural Network* (CNN) memberikan hasil yang baik dengan nilai akurasi sebesar 96% dan untuk kombinasi segmentasi karakter pada kumpulan data terpisah memperoleh akurasi 75%. Penelitian [5] CNN digunakan untuk klasifikasi aksara Lampung dan menunjukkan bahwa model sudah baik yaitu dengan hasil evaluasi nilai *training* akurasi sebesar 57% serta nilai presisi sebesar 87%. Penelitian [6] CNN digunakan untuk pengenalan pola tulis tangan aksara Arab menggunakan CNN dengan nilai akurasi sebesar 78.10%. Hal ini menunjukkan bahwa metode CNN memiliki potensi yang baik untuk dikembangkan sebagai media pembelajaran aksara Arab.

Aksara Lontara

Aksara Lontara 1 merupakan salah satu aksara tradisional dari Indonesia yang berkembang di wilayah Sulawesi Selatan. Aksara ini digunakan untuk menulis bahasa suku Bugis, Makassar, dan Mandar sejak abad 16 sampai 20 masehi. Aksara ini jarang digunakan dan dipelajari

Ina' Sure' atau Anrong Lontara'

ka	ga	nga	ngka
pa	ba	ma	mpa
ta	da	na	nra
ca	ja	nya	nca
ya	ra	ia	wa
sa	a	ha	

Gambar 1. Aksara Lontara

di sekolah untuk saat ini karena fungsinya digantikan oleh bahasa Latin dan penerapannya sangat terbatas di kehidupan sekarang. Aksara Lontara ini memiliki 23 aksara dasar dan 4 aksara yang mewakili suku kata pra-nasal yaitu ngka, mpa, nra, dan nca [7].

Optical Character Recognition

Optical Character Recognition (OCR) adalah suatu sistem komputer yang mampu membaca karakter dari mesin cetak maupun tulis tangan. OCR adalah aplikasi yang mengubah gambar karakter menjadi format teks dengan mencocokkan pola karakter secara baris demi baris dengan pola yang disimpan dalam database aplikasi. Hasil dari proses OCR berupa teks yang sesuai dengan gambar yang dihasilkan oleh scanner. Tingkat keakuratan teks tergantung pada kejelasan gambar dan metode yang digunakan [2].

Convolutional Neural Network

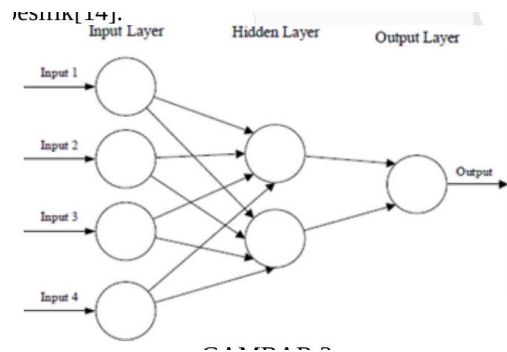
Convolutional Neural Network (CNN) 2 adalah jenis jaringan saraf tiruan yang sering digunakan untuk mengolah data citra. CNN dapat mendeteksi dan mengenali objek dalam bentuk gambar. Teknik ini terinspirasi oleh cara mamalia, termasuk manusia yang menghasilkan persepsi visual. Secara umum, CNN tidak jauh berbeda dari jaringan saraf tiruan pada umumnya. CNN terdiri dari neuron yang memiliki bobot, bias, dan fungsi aktivasi. Lapisan CNN dibentuk oleh *neuron-neuron* yang disusun dalam pola tertentu sehingga membentuk filter dengan panjang dan lebar yang spesifik [8].

Arsitektur Convolutional Neural Network

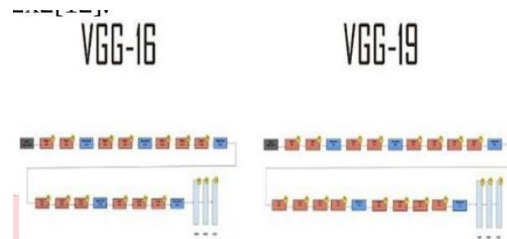
Jenis arsitektur yang dipakai dalam penelitian ini yaitu arsitektur *Visual Geometry Group* (VGG16, VGG19) dan *Residual Network* (ResNet, ResnetV2).

a. *Visual Geometry Group* (VGG16 dan VGG19)

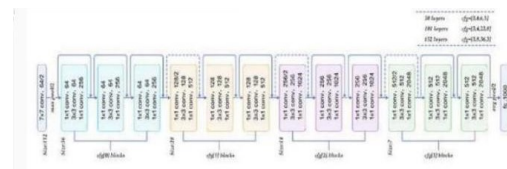
Arsitektur VGGNet[14] adalah pengembangan dari alexnet yang berfokus pada proses ekstraksi fitur di *layer convolution* untuk mendapatkan representasi citra yang banyak sehingga dapat diklasifikasikan dengan baik. Arsitektur VGG16 adalah model *deep learning* yang terdiri dari 16 lapisan. Ini termasuk 13 Lapisan



Gambar 2. Convolutional Neural Network



Gambar 3. VGG16 dan VGG19

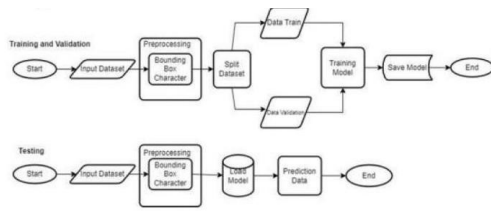


Gambar 4. ResNet

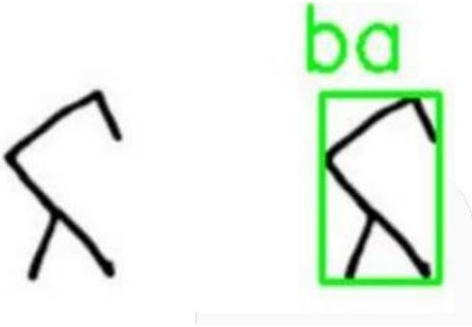
konvolusi dan 3 lapisan *fully connected*. *Layer convolutional* menggunakan kernel 3x3, dan *max pooling* 2x2. Dan ada 5 *layer max pooling* yang mengikuti beberapa *layer convolution*. Sedangkan arsitektur VGG19 mirip dengan VGG16 namun memiliki 19 *layer*. Memiliki 16 *layer convolution* dan 3 *layer max pooling*. Arsitektur ini juga memiliki ukuran kernel 3x3 di *layer convolution* dan *max pooling* 2x2 [9]. Untuk arsitektur VGG16 dan VGG19 3 dengan menambahkan lapisan dense dengan 400 dan 100 *neuron* serta fungsi aktivasi ReLU. Dan lapisan dense dengan 23 *neuron* dengan fungsi aktivasi softmax. Ini digunakan untuk klasifikasi multi kelas yaitu sebanyak 23 kelas atau huruf.

b. *Residual Network* (ResNet dan ResNetV2)

Arsitektur ResNet merupakan model yang sudah dilatih sebelumnya, sehingga tidak memerlukan penyesuaian khusus untuk setiap lapisannya. Prinsip utama ResNet adalah membangun jaringan yang lebih dalam daripada VGG, hal ini berguna untuk mengoptimalkan jumlah lapisan untuk mengatasi masalah hilangnya *gradien*. Ada beberapa tipe arsitektur ResNet salah satunya adalah ResNet dan ResNetV2. Perbedaannya terletak pada jumlah lapisan yang digunakan, serta penambahan lapisan *batch normalization* sebelum lapisan bobot pada setiap lapisan dalam ResNetV2 [10]. Untuk arsitektur ResNet dan ResNetV2 4 dengan menambahkan lapisan dense dengan 400 dan 100 *neuron* serta fungsi aktivasi ReLU. Dan lapisan dense dengan 23 *neuron* dengan fungsi aktivasi softmax. Ini digunakan untuk klasifikasi multi kelas yaitu sebanyak 23 kelas atau huruf.



Gambar 5. Flowchart



Gambar 6. Bounding Box Character

Metodologi Penelitian

Rancangan Sistem

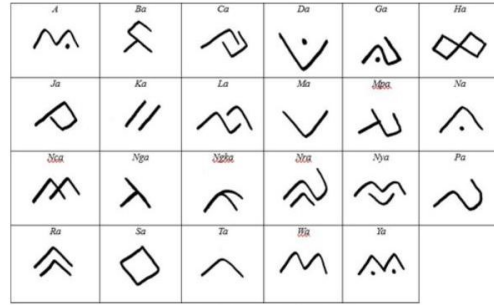
Gambar 5 menunjukkan rancangan sistem di mana penelitian ini melibatkan 2 proses utama yaitu *training validation* dan *testing*. Sebelum memulai kedua proses tersebut, dilakukan tahapan preprocessing dengan *bounding box character*. Proses *training validation* dimulai dengan melatih model CNN sebanyak 100 *epoch* untuk menghasilkan model yang akan digunakan dalam proses *testing*. Pada proses testing, model yang telah dilatih akan digunakan untuk memprediksi huruf dari *data input* serta melakukan klasifikasi untuk setiap huruf.

Bounding Box Character

Bounding box disini digunakan untuk menandai karakter yang ingin diidentifikasi dalam sebuah gambar, memungkinkan komputer untuk melakukan pengenalan objek dengan lebih efektif. Saat komputer digunakan untuk mengklasifikasikan gambar dan mendeteksi objek, *bounding box* membantu visualisasi dengan memberi garis pembatas di sekitar objek yang ditemukan. Ini dilakukan dengan menghasilkan koordinat yang menunjukkan lokasi objek di dalam gambar serta memberi *label* dan batas pada objek tersebut [11].

Dataset

Dataset yang digunakan dalam penelitian ini terdiri dari data yang didapatkan dari website Kaggle.com dan data yang telah dikumpulkan peneliti secara langsung dengan cara mengambil sampel sebanyak 15 orang berbeda lalu digabungkan dengan dataset yang telah diperoleh dari website yang akan dijalankan di proses *testing*. Tujuan adanya dataset tambahan tersebut untuk melakukan variasi tulisan didalam dataset agar meningkatkan proses pembelajaran dari model. Dataset terkumpul sebanyak 12.282 gambar yang dibagi menjadi data *training* 7.452 gambar, data *validation* 3.726 gambar, dan data *testing* 1.104 gambar. Didalam dataset mpa, na, nca, nga, ngka, nra, nya, pa, ra, sa, ta, wa, ya. Masing- masing kelas berisi 324 gambar untuk data *training*, 162 gambar untuk data *validation*, dan 48 gambar untuk data *testing*.



Gambar 7. Dataset

Pelatihan Model

Untuk melatih model yang akan dibangun menggunakan TensorFlow dan Keras untuk model VGG16, VGG19, ResNet, dan ResNetV2. Proses preprocessing data input adalah *bounding box character* untuk memberi garis pembatas pada objek. Lalu mempersiapkan data *training* dan *validation* dengan ukuran gambar 64x64. Model dilatih sebanyak 100 *epoch* pada tahap *training* dan data *validation* digunakan untuk mengevaluasi model selama pelatihan. Model yang telah dilatih akan disimpan sesuai nama dari model tersebut, lalu pada tahap *testing* dilakukan untuk mengevaluasi performa model untuk klasifikasi huruf dengan data baru berupa tulis tangan yang telah peneliti kumpulkan.

Pengujian Model

Pengujian model dimulai dengan membandingkan grafik akurasi saat pelatihan di tahap *training* dan *validation* untuk mengetahui perbedaan antara model tersebut. Model yang digunakan ada 4 yaitu model VGG16, VGG19, ResNet, dan ResNetV2. Kemudian model akan mengklasifikasi dataset yang telah di input dan hasilnya akan dipetakan ke dalam confusion matrix multiclass. Dengan representasi visual tersebut, dapat dibandingkan performa dari semua model yang telah dibuat. Selanjutnya, model tersebut akan dibandingkan dengan menghitung nilai presisi dan recall untuk masing- masing kelas dalam model. Untuk menghitung matrix tersebut diperlukan tiga matrix penilaian, yaitu *True Positive* (TP), *False Positive* (FP), dan *False Negative* (FN). Dari ketiga matrix tersebut dapat ditentukan nilai akurasi, presisi, *recall*, dan *F1-score* yang dapat digunakan untuk menilai kemampuan model secara keseluruhan. Rumus perhitungan untuk masing-masing nilai adalah:

- Akurasi

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- Presisi

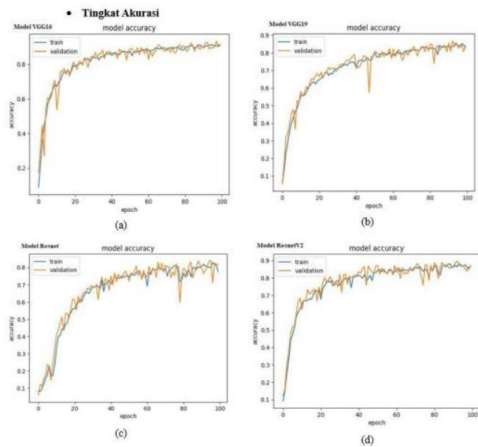
$$Precision = \frac{TP}{TP + FP}$$

- Recall

$$Recall = \frac{TP}{TP + FN}$$

- F1-Score

$$F1-Score = \frac{2 \times precision \times recall}{precision + recall}$$



Gambar 8. Dataset

Table 1. Performansi Model

Model	Akurasi	Presisi	Recall	F1-Score
VGG16	0.90	0.91	0.90	0.90
VGG19	0.83	0.84	0.83	0.83
ResNet	0.85	0.86	0.85	0.85
ResNetV2	0.86	0.88	0.86	0.86

Table 2. Huruf Aksara Lontara Tertinggi dan Terendah

	Huruf	Prediksi
Akurasi tertinggi		
Akurasi terendah		

Hasil dan Pembahasan

Training dan Validation

Pada tahap ini dilakukan skenario *training* dan *validation* pada 100 *epoch*. Dari gambar 8 dapat dilihat bahwa tingkat akurasi dari setiap model terus meningkat dan tidak jauh berbeda. Hal ini dapat dipengaruhi dari jumlah data *training* dan *validation* yang sama. Untuk tingkat akurasi yang paling stabil yaitu model VGG16.

Tabel

Pada tabel 1 dapat dilihat model VGG16 memiliki tingkat akurasi, presisi, *recall*, *f1-score* yang terbaik dibandingkan model lainnya yaitu sebesar 90%.

Analisis Hasil Pengujian

Analisis adalah tahap krusial yang memungkinkan peneliti untuk memilih model terbaik dari berbagai model yang diuji dalam proses klasifikasi aksara Lontara. Model yang terbaik adalah model VGG16 dengan nilai akurasi 97% pada proses *training* dan *validation* serta 90% pada proses *testing*.

Dapat dilihat pada tabel 2 huruf dengan akurasi tertinggi adalah huruf 'ka', sedangkan akurasi terendah adalah huruf 'wa' dikarenakan huruf 'wa' tersebut memiliki kemiripan dengan beberapa huruf seperti huruf 'a' dan huruf 'ya'.

Kesimpulan

Berdasarkan hasil penelitian, disimpulkan bahwa metode CNN efektif dalam merancang dan mengimplementasikan sistem klasifikasi aksara Lontara. Hal ini terbukti dari hasil perbandingan performa beberapa arsitektur atau model CNN. Dari hasil perbandingan tersebut, model VGG16 menunjukkan tingkat akurasi terbaik, dengan nilai akurasi mencapai 97% pada tahap *training* dan *validation*, serta 90% pada tahap *testing*. Saran untuk penelitian selanjutnya adalah mengembangkan model klasifikasi menggunakan metode yang berbeda dan dataset yang lebih beragam. Hal ini akan memperluas kemampuan klasifikasi aksara Lontara.

Daftar Pustaka

- Ahmad AA. Melestarikan Budaya Tulis Nusantara; 2014.
- Setiawan A, Sujaini H, Bijaksana A. Implementasi Optical Character Recognition (OCR) pada Mesin Penerjemah Bahasa Indonesia ke Bahasa Inggris. 2017.
- Hidayat A, Nurtanio I, Tahir Z. Segmentation and Recognition of Handwritten Lontara Characters Using Convolutional Neural Network. 2020.
- Hanindria IS, Hendry. Pengklasifikasian Aksara Jawa Metode Convolutional Neural Network. 2022.
- Mulyanto A, et al. Penerapan Convolutional Neural Network (CNN) pada Pengenalan Aksara Lampung Berbasis Optical Character Recognition (OCR). JEPIN (Jurnal Edukasi dan Penelitian Informatika). 2021;7(1):52-7.
- Kasim N, Nugraha GS. Pengenalan Pola Tulisan Tangan Aksara Arab Menggunakan Metode CNN. 2021.
- Safitri DA. Pengenalan Pola Aksara Lontara dengan Metode Optical Character Recognition. 2018.
- O'Shea K, Nash R. An Introduction to Convolutional Neural Network. 2020.
- Atliha V, Sesok D. Comparison of VGG and ResNet used as Encoders for Image Captioning. 2020.
- Ridhovan A, Suharso A. Penerapan Metode Residual Network (ResNet) Dalam Klasifikasi Penyakit Pada Daun Gandum. 2022.
- Lempitsky V, Kohli P, Rother C, Sharp T. Image Segmentation with A Bounding Box Prior. 2009.