

Optimasi Optical Character Recognition Tulisan Tangan Aksara Sunda Menggunakan Model Berbasis EfficientNet

Tugas Akhir
diajukan untuk memenuhi salah satu syarat
memperoleh gelar sarjana
pada Program Studi Informatika
Fakultas Informatika Universitas Telkom

103012380509
Gid Achmad Ahlul Fadli



Program Studi Sarjana Informatika
Fakultas Informatika
Universitas Telkom
Bandung

2025

LEMBAR PENGESAHAN

**Optimasi Optical Character Recognition Tulisan Tangan Aksara Sunda Menggunakan
Model Berbasis EfficientNet**

*Optimizing Handwritten Aksara Sunda Optical Character Recognition using
EfficientNet-based Model*

NIM: 103012380509

Gid Achmad Ahlul Fadli

Tugas akhir ini telah diterima dan disahkan untuk memenuhi sebagian syarat memperoleh
gelar pada Program Studi Sarjana Informatika

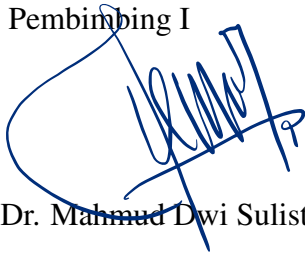
Fakultas Informatika

Universitas Telkom

Bandung, 13 Agustus 2025

Menyetujui

Pembimbing I



Dr. Mahmud Dwi Sulistiyo, S.T., M.T

NIP: 13880017

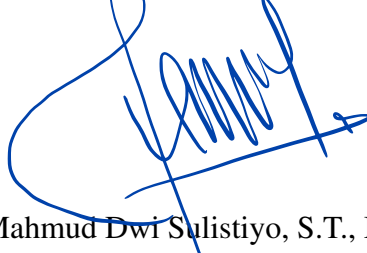
Pembimbing II

Donni Richasdy, S.T., M.T.

NIP: 15900040

Ketua Program Studi

Sarjana Informatika,



Dr. Mahmud Dwi Sulistiyo, S.T., M.T

NIP: 13880017

LEMBAR PERNYATAAN

Dengan ini saya, Gid Achmad Ahlul Fadli, menyatakan sesungguhnya bahwa Tugas Akhir saya dengan judul "**Optimasi Optical Character Recognition Tulisan Tangan Aksara Sunda Menggunakan Model Berbasis EfficientNet**" beserta dengan seluruh isinya adalah merupakan hasil karya sendiri, dan saya tidak melakukan penjiplakan yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan. Saya siap menanggung resiko/sanksi yang diberikan jika dikemudian hari ditemukan pelanggaran terhadap etika keilmuan dalam buku TA atau jika ada klaim dari pihak lain terhadap keaslian karya.

Bandung, 13 Agustus 2025

Yang Menyatakan,



Gid Achmad Ahlul Fadli

Optimasi Optical Character Recognition Tulisan Tangan Aksara Sunda Menggunakan Model Berbasis EfficientNet

Gid Achmad Ahlul Fadli¹, Mahmud Dwi Sulistiyo², Donni Richasdy³

^{1,2,3}Fakultas Informatika, Universitas Telkom, Bandung

¹gidachmad@students.telkomuniversity.ac.id,

²mahmuddwis@telkomuniversity.ac.id, ³donnir@telkomuniversity.ac.id

Abstrak

Aksara daerah merupakan sistem ortografi hasil masyarakat daerah yang meliputi aksara dan sistem pengaksaraan untuk menuliskan bahasa daerah, salah satunya adalah Aksara Sunda. Berbagai penelitian sebagai usaha digitasi dalam pelestarian Aksara Sunda dengan berbagai metode dalam machine learning dengan fokus Optical Recognition Character (OCR) terutama *Convolutional Neural Network* (CNN) sudah dilakukan. Namun, data yang ada pada saat ini terbatas pada karakter yang menjadi bagian swara, ngalagena, rarangkén dan angka. Masih sedikit sistem OCR yang menggunakan dataset kata Aksara Sunda yang berupa gabungan antara ngalagena dan swara dengan rarangkén. Dengan penambahan dataset kata tersebut dapat menghasilkan dua model, yaitu model ensemble dan model *Connectionist Temporal Classification* (CTC). Model ensemble dengan ensemble weight sebesar 0,5 berhasil mencapai akurasi sebesar 0,912, presisi sebesar 0,889, sensitifitas sebesar 0,701, dan *F1-score* sebesar 0,755. Model CTC berhasil mencapai nilai *Character Error Rate* (CER) sebesar 0,069 dan nilai *Word Error Rate* (WER) sebesar 0,087.

Kata kunci : *Optical Character Recognition, Aksara Sunda, Convolutional Neural Network, Recurrent Neural Network, Connectionist Temporal Classification, EfficientNet.*

Abstract

Aksara is an orthographic system resulting from regional communities which includes characters and pronunciation systems for writing regional languages, one of which is Sundanese script. Various studies as an effort to digitize in the preservation of Sundanese script with various methods in machine learning with a focus on Optical Recognition Character (OCR) especially *Convolutional Neural Network* (CNN) have been carried out. However, the current data is limited to characters that are part of swara, ngalagena, rarangkén and numbers. There is only a few of OCR using data in the form of a combination of ngalagena and swara with rarangkén into a word dataset. With the addition of the word dataset, it can produce two models, namely the ensemble model and the *Connectionist Temporal Classification* (CTC) model. The ensemble model with an ensemble weight of 0.5 achieved an accuracy of 0.912, precision of 0.889, sensitivity of 0.701, and *F1-score* of 0.755. While, the CTC model successfully achieved a *Character Error Rate* (CER) value of 0.069 and a *Word Error Rate* (WER) value of 0.087.

Keywords : *Optical Character Recognition, Aksara Sunda, Convolutional Neural Network, Recurrent Neural Network, Connectionist Temporal Classification, EfficientNet.*

1. Pendahuluan

Latar Belakang

Aksara daerah merupakan sistem ortografi hasil masyarakat daerah yang meliputi aksara dan sistem pengaksaraan untuk menuliskan bahasa daerah[1]. Aksara Sunda termasuk ke dalam kategori Aksara dan Naskah Ka Ga Nga yang diakui sebagai warisan budaya tak benda pada tahun 2013[2]. Upaya pelestarian aksara daerah sudah marak dilakukan oleh berbagai lapisan masyarakat. Pemerintah daerah provinsi Jawa Barat (Jabar) mengeluarkan Peraturan Daerah (Perda) Provinsi Jabar nomor 5 tahun 2003, lalu diubah oleh Perda Provinsi Jabar nomor 14 tahun 2014 tentang Pemeliharaan Bahasa, Sastra, dan Aksara Daerah[3]. Maka dari itu, kalangan civitas akademik, khususnya bidang informasi dan teknologi, sudah banyak melakukan penelitian untuk melestarikan aksara daerah menggunakan media digital.

Banyak penelitian dengan fokus *Optical Character Recognition* (OCR) melalui metode *Convolutional Neural Network* (CNN) sudah dilakukan sebagai bentuk proses digitasi Aksara Sunda. Penelitian sebelumnya oleh Hayati K, *et al*[3] dilakukan eksplorasi metode CNN klasik untuk OCR Aksara Sunda. Fitur pada Aksara Sunda diekstraksi menggunakan *Histogram of Oriented Gradients* (HOG). Penelitian ini membandingkan metode *Support Vector Machine* (SVM), *K-Nearest Neighbors* (*k*-NN), *Decision Tree*, *Naïve Bayes*, *Random Forest*, dan *Artificial Neural Network* (ANN) dalam proses OCR. Penelitian lainnya oleh Prameswari M, *et al*[4] dengan fokus transfer learning

Dilihat dari penelitian-penelitian diatas, penelitian yang sudah dilakukan masih berfokus pada klasifikasi karakter. Hal ini menunjukan masih sedikit penelitian yang berfokus pada klasifikasi kata untuk OCR Aksara Sunda. Salah satu dari sedikitnya penelitian yang ada tentang OCR Aksara Sunda untuk klasifikasi kata dilakukan oleh Gerhana Yana, *et al* [6] yang menggunakan metode *Template Matching Algorithm* dan *Feature Extraction Algorithm*. Performa model pada penelitian ini untuk klasifikasi kata dari tulisan tangan secara spesifik mencapai akurasi dengan nilai 75% dari 30 sample kata. Oleh karena itu, diperlukan penelitian lebih lanjut pada pengklasifikasian kata sebagai langkah pengembangan berikutnya dalam OCR khususnya tulisan Aksara Sunda.

Permasalahan yang diangkat pada penelitian ini adalah belum adanya sistem *Optical Character Recognition* (OCR) pada tingkat kata. Kriteria yang menjadi fokus dalam permasalahan ini adalah menghasilkan model klasifikasi kata tulisan tangan Aksara Sunda tanpa menggunakan model segmentasi. Maka penelitian ini merumuskan dua masalah utama, yaitu bagaimana penerapan model berbasis EfficientNet untuk klasifikasi dalam *Optical Character Recognition* tingkat kata tanpa segmentasi pada tulisan tangan Aksara Sunda, serta bagaimana performa model yang dihasilkan dari penerapan model berbasis EfficientNet pada model klasifikasi tersebut.

Tujuan

2. Studi Terkait

Menurut Batra P, *et al* [7], *Optical Character Recognition* (OCR) adalah ranah penelitian yang aktif mencari cara untuk menghasilkan sebuah sistem otomatis yang dapat mengekstrak dan memproses teks dari gambar. Tujuan dari OCR adalah untuk menghasilkan teks yang dapat disunting dalam bentuk digital sebagai hasil dari gambar yang telah dipindai.

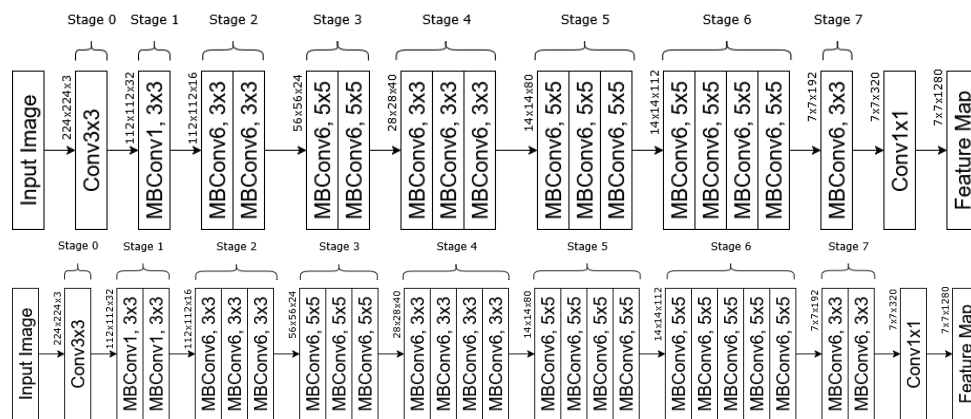
Dataset Aksara Sunda yang akan digunakan pada penelitian ini merupakan hasil standarisasi dari buku Direktori Aksara Sunda untuk Unicode[8], sebagai upaya standarisasi Aksara Sunda dalam pelestarian salah satu budaya suku Sunda. Aksara Sunda terdiri dari swara, ngalagena, angka, dan rarangkén. Aksara Sunda ngalagena berjumlah 25 karakter dengan pemabagian 23 karakter dasar dengan tambahan karakter serapan sebanyak 2 karakter yaitu sya dan kha. Sementara itu, aksara swara pada Aksara Sunda berjumlah 7 karakter. Contoh kalimat dalam aksara sunda diperlihatkan oleh gambar 1.



EfficientNet

Model *Convolutional Neural Network* (CNN) yang telah banyak berkembang menjadikan ada banyak pilihan menarik dibandingkan model-model populer yang telah disebutkan. Seperti dilansir pada situs web keras.io[9]. Poin menarik dari EfficientNet adalah mudahnya melakukan fine-tuning dalam skalabilitas model yang digunakan.

Skalabilitas tersebut dimulai dari B0 sampai dengan B7, dilakukan dengan menambahkan satu blok proses pada setiap satu tahap peningkatan seperti terlihat pada gambar 2. Hal tersebut menjadikan EfficientNet menjadi salah satu pilihan model yang adaptif mengikuti jumlah sumber daya yang ada.



Gambar 2. Perbandingan Arsitektur EfficientNetB0 [10] dengan EfficientNetB1 [11]

Recurrent Neural Network

Recurrent Neural Network (RNN) adalah salah satu tipe ANN yang dirancang secara khusus untuk menangani data secara sekuensial dengan bantuan timestep [12]. Hal tersebut menjadikan RNN sebagai metode yang populer dalam mengolah data yang memiliki pola data secara sekuensial, baik dalam bentuk teks, audio, dan time series. Beberapa penelitian juga menyarankan gabungan CNN dengan RNN dengan tujuan saling melengkapi dalam implementasi OCR [13-15].

Bidirectional Long-Short Term Memory

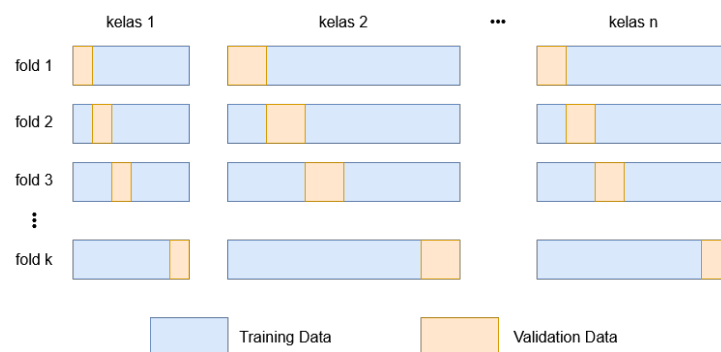
Bidirectional Long-Short Term Memory (BiLSTM) merupakan bentuk pengembangan lainnya dari RNN dan LSTM. Berbeda dengan kedua metode tersebut yang hanya bisa memproses data sekuensial dalam satu arah, BiLSTM dirancang agar bisa memproses data sekuensial dalam dua arah [16]. Metode ini digunakan dalam beberapa penelitian mengenai OCR untuk bahasa arab [17].

Connectionist Temporal Classification

Connectionist Temporal Classification (CTC) Loss adalah teknik evaluasi yang digunakan sebagai pengganti dalam pengolahan data sekuensial dengan panjang input dan target output yang tidak selaras [18]. CTC Loss akan mengevaluasi data input sekuensial dengan menambahkan token “blank” sebagai pengganti data yang berulang dalam sekuensial tersebut [19].

Stratified K-Fold Cross Validation

Stratified K-Fold Cross Validation adalah variasi pengembangan dari K-fold cross validation biasa yang memungkinkan data terbagi dengan persentase yang sama untuk masing-masing kelas [20]. Hal ini memastikan model melihat semua kelas walaupun ada perbedaan jumlah data setiap kelas.



Gambar 3. Stratified K-Fold Cross Validation

Confussion Matrix

Confusion matrix merupakan metrik yang umum digunakan dalam pengevaluasian machine learning tipe klasifikasi[21]. Metrik-metrik yang dihasilkan berupa akurasi (*accuracy*), presisi (*precision*), sensitifitas (*recall*), dan nilai F1 (*F1-score*).

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Sensitifitas = \frac{TP}{TP + FN} \quad (2)$$

$$Presisi = \frac{TP}{TP + FP} \quad (3)$$

$$F1 - score = \frac{2(sensitifitas \times presisi)}{sensitifitas + presisi} \quad (4)$$

Keterangan :

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

Character Error Rate and Word Error Rate

Character Error Rate (CER) dan *Word Error Rate* (WER) adalah metrik yang bagus digunakan untuk mengukur akurasi OCR terutama yang berbasis kata. Kedua metrik ini menggunakan *levenshtein distance*, yang mengukur jumlah minimum perubahan karakter/kata agar dua string menjadi sama baik dalam tingkat karakter atau kata[14].

$$CER = \frac{CI + CD + CS}{CC + CD + CS} \quad (5)$$

Keterangan :

CI = Jumlah karakter yang ditambahkan.

CD = Jumlah karakter yang dihapus.

CS = Jumlah karakter yang disubstitusi.

CC = Jumlah karakter yang benar.

$$CER = \frac{WI + WD + WS}{WC + WD + WS} \quad (6)$$

Keterangan :

WI = Jumlah kata yang ditambahkan.

WD = Jumlah kata yang dihapus.

WS = Jumlah kata yang disubstitusi.

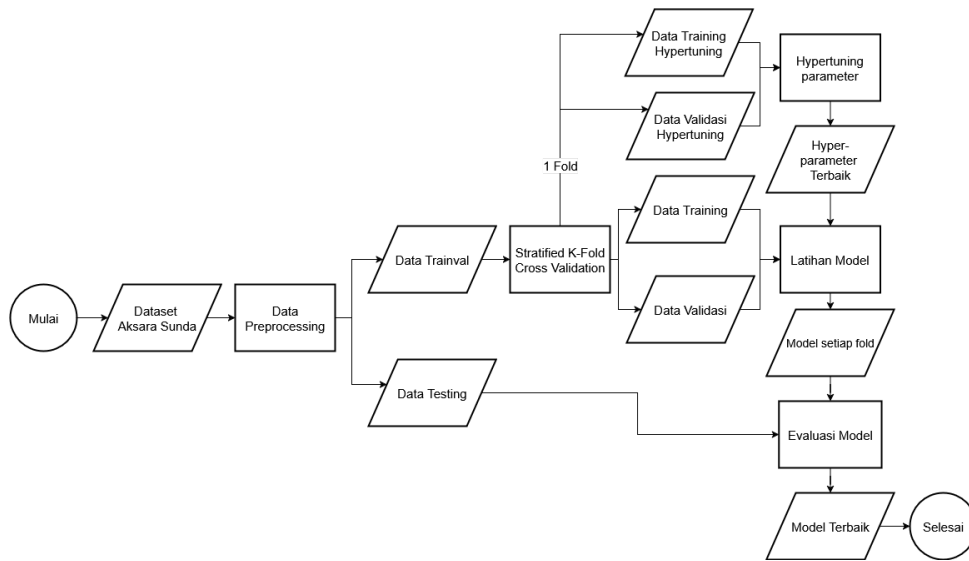
WC = Jumlah kata yang benar.

3. Skema yang Dibangun

Penelitian ini bertujuan untuk membangun sistem OCR yang mengenali kata dalam tulisan tangan Aksara Sunda dengan model berbasis EfficientNet. Pada proses prenelitian ini, EfficientNetB0 dipilih sebagai model dasar dari model yang akan diimplementasikan. Arsitektur ini memiliki kemampuan komputasi yang cukup memadai untuk pelatihan model transfer learning dengan dataset berukuran kecil hingga sedang[22]. Arsitektur MobileNetV2 tidak dipilih karena berdasarkan penelitian oleh Permana, *et al*[23], EfficientNetB0 lebih unggul daripada MobileNetV2. Sementara itu, EfficientNetV2 tidak dipilih dengan alasan ukuran dataset yang masih tergolong sedang. Hal ini menjadikan keunggulan utama dari EfficientNetV2 dibanding EfficientNetB0, yaitu waktu pelatihan yang lebih singkat dan parameter yang lebih efisien tidak terlalu signifikan dalam penelitian ini[24]. Model akan diimplementasikan menjadi dua model yang berbeda, yaitu Model ensemble yang melakukan klasifikasi menggunakan *ensemble learning* dan model CTC yang melakukan klasifikasi menggunakan fungsi *Connectionist Temporal Classification Loss*.

Implementasi kedua model tersebut dilatih melalui proses seperti ditunjukkan gambar 4. Proses pelatihan dimulai dengan proses pembagian dataset. Proses pembagian dataset dilakukan dengan ratio 80:20 untuk data trainval dan data testing. Proses Stratified K-Fold Cross Validation dilakukan dua kali, yaitu untuk proses *Hypertuning Parameter* dan pelatihan model. Pada proses *Hypertuning Parameter* menggunakan 1 fold, sementara untuk pelatihan model menggunakan 5 fold. *Hypertuning parameter* ini yang digunakan adalah learning rate dan dropout. Rentang nilai *learning rate* yang dipakai adalah dari 0,001; 0,0001; dan 0,00005, sedangkan rentang nilai *dropout* adalah 0,2; 0,3; dan 0,4[25] [26]. Proses pelatihan model dilakukan setelah mendapatkan *hyperparameter* terbaik.

Pada proses pelatihan secara menyeluruh, *Stratified K-Fold Cross Validation* membagi data trainval dengan ratio 80:20 untuk data training dan data validation.

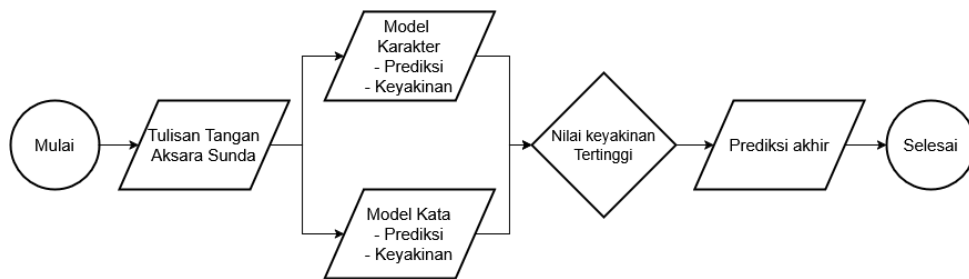


Gambar 4. Diagram Alir dari pelatihan model yang digunakan

Perbedaan antara kedua implementasi model ini terletak pada proses evaluasi model pada saat pelatihan dan evaluasi model untuk menentukan model terbaik. Model ensemble dievaluasi menggunakan *confussion matrix*, lebih tepatnya metrik akurasi pada saat pelatihan dan *F1-score* saat penentuan model terbaik. Sementara itu, model CTC menggunakan *CTC Loss* sebagai fungsi evaluasi pada saat pelatihan dan metrik *Character Error Rate* (CER) serta *Word Error Rate* (WER) pada saat evaluasi model terbaik. *CTC Loss* akan mengevaluasi output dense layer per timestep dengan menjumlahkan probabilitas semua kemungkinan urutan prediksi terhadap label target.

3.1 Model Ensemble

Model pertama yang diimplementasikan adalah Model Ensemble. Model yang digunakan dalam model ensemble berjumlah dua model klasifikasi berbasis EfficientNetB0 seperti terlihat pada gambar 5, yaitu model 1 untuk klasifikasi karakter, dan model 2 untuk klasifikasi kata. Dengan begitu, dataset yang digunakan dalam pelatihan model ini terbagi menjadi dua dataset, yaitu dataset karakter dan dataset kata. Prediksi akhir model ensemble dihasilkan melalui *confidence (weighted)* voting dengan cara membandingkan nilai *confidence (weighted)* dari kedua model.



Gambar 5. Diagram alir model ensemble

Nilai *confidence (weighted)* masing-masing model didapatkan dengan melakukan perhitungan menggunakan nilai α sebagai ensemble weight. Nilai α merepresentasikan bobot tingkat kepercayaan terhadap masing-masing model. Perhitungan *confidence (weighted)* dapat dilihat sebagai berikut.

$$confidence(weighted)_{karakter} = \alpha \times confidence_{karakter} \quad (7)$$

$$confidence(weighted)_{kata} = \alpha \times confidence_{kata} \quad (8)$$

Nilai α yang digunakan dalam rumus 7 dan 8 akan diuji coba dengan rentang 0,4; 0,5; dan 0,6. Rentang ini dipilih dengan mempertimbangkan pengaruh model sebagai dasar dari model ensemble. Nilai $\alpha = 0,4$ memberikan bobot lebih kepada model karakter, nilai $\alpha = 0,6$ memberikan bobot lebih kepada model kata, sementara itu nilai $\alpha = 0,5$ memberikan bobot yang sama antara dua model.

Pada model ensemble ini, model EfficientNetB0 yang digunakan sebagai model dasar diimplementasikan seperti terlihat pada gambar 6. Gambar Aksara Sunda diterima oleh input layer pada EfficientNetB0 yang menghasilkan output berupa *feature map*. EfficientNetB0 dimodifikasi dengan proses *unfreeze layer* dari layer 163 sampai layer terakhir. Proses ini digunakan dalam beberapa penelitian agar mencapai performa yang lebih baik agar model bisa melakukan fine-tuning terhadap dataset khusus, dalam penelitian ini tulisan Aksara Sunda yang berbeda dengan tulisan latin [27, 28].

Proses penentuan model terbaik yang dipakai pada model ensemble dari masing-masing model dilakukan menggunakan evaluasi metrik *F1-score (weighted)*. Hal ini dilakukan dengan tujuan untuk memberikan gambaran komprehensif terhadap performa model terhadap masing-masing dataset. Pemilihan *F1-score (weighted)* dibandingkan dengan metrik akurasi dilakukan karena metrik akurasi cenderung bias terhadap kelas mayoritas. Sehingga penggunaan metrik ini diyakini lebih tepat dan representatif.

3.2 Model Connectionist Temporal Classification Loss

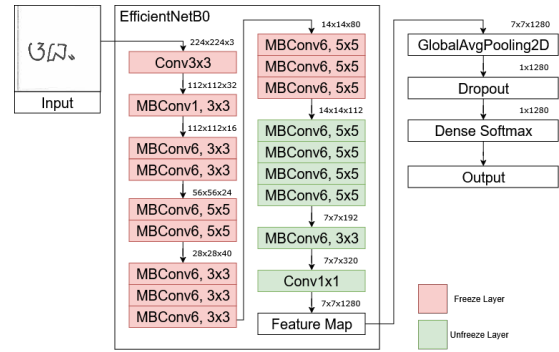
Model kedua yang diimplementasikan adalah Model *Connectionist Temporal Classification (CTC)*. Pada penelitian ini, implementasi dari model ini dilakukan secara sederhana karena tidak menjadi fokus utama pembahasan. Model CTC ini menggunakan EfficientNetB0 sebagai ekstraktor fitur dari input gambar seperti ditunjukkan oleh gambar 7. Selanjutnya, fitur gambar diproses melalui layer *conv2D* dan *reshape* agar menjadi data sekuensial yang memiliki *timestep*. Data ini selanjutnya diproses oleh dua layer *Bidirectional Long Short Term-Memory (BiLSTM)* untuk menghasilkan logits. Logits sequence merupakan representasi prediksi karakter pada setiap langkah waktu. Logits inilah yang akan dievaluasi oleh fungsi *CTC Loss* untuk menyelaraskan antara hasil prediksi dan label target.

4. Evaluasi

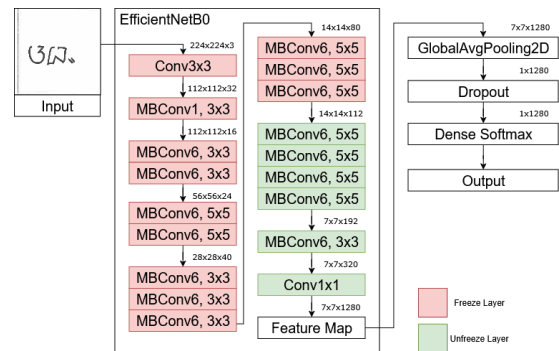
4.1 Skenario Model Ensemble

Hyperparameter yang digunakan pada proses *hypertuning parameter* adalah *dropout* dan *learning rate*. Proses ini dilakukan dengan 5 epochs sebanyak 8 kali untuk masing-masing model. Rentang nilai parameter *dropout* yang dipakai adalah 0,2; 0,3; dan 0,4. Sementara untuk parameter *learning rate* adalah 0,001; 0,0001; dan 0,00005. Tabel 1 menampilkan hasil dari tahap *hypertuning parameter* pada masing-masing model.

Tabel 1 memperlihatkan hasil dari proses *hypertuning parameter* pada kedua model. Model karakter mendapatkan hasil terbaik dalam 5 epochs dengan nilai *dropout* sebesar 0,4 dan nilai *learning rate* sebesar 0,001 dengan akurasi 0,966. Sementara itu, model kata mendapatkan hasil terbaik dalam 5 epochs dengan nilai *dropout* sebesar 0,2 dan nilai *learning rate* sebesar 0,001 dengan nilai akurasi 0,496. Perbedaan parameter terdapat pada nilai



Gambar 6. Implementasi EfficientNetB0 sebagai model dasar dari model ensemble.



Gambar 7. Implementasi EfficientNetB0 sebagai model dasar dari model CTC.

Tabel 1. Hasil pengaturan hyperparameter untuk model karakter dan model kata.

Percobaan	Dropout		Learning Rate		Akurasi	
	Model Karakter	Model Kata	Model Karakter	Model Kata	Model Karakter	Model Kata
1	0,2	0,2	0,001	0,001	0,959	0,496
2	0,3	0,3	0,001	0,001	0,945	0,284
3	0,4	0,4	0,0001	0,001	0,966	0,153
4	0,2	0,2	0,0001	0,0001	0,880	0,005
5	0,3	0,3	0,0001	0,0001	0,861	0,004
6	0,2	0,2	0,00005	0,00005	0,774	0,004
7	0,3	0,3	0,00005	0,00005	0,760	0,004
8	0,4	0,4	0,00005	0,00005	0,805	0,004

dropout yang bertugas untuk menjaga model agar tidak terjadi overfitting. Model karakter dengan dataset kelas yang lebih kecil dan merata memerlukan nilai *dropout* lebih besar agar menjaga model tidak *overfit* dan tetap general. Sedangkan, model kata dengan dataset kelas yang banyak (220 kelas) dan jumlah data setiap kelas sedikit cenderung sensitif akan kehilangan informasi sehingga memerlukan *dropout* lebih kecil. Setelah diketahui hyperparameter terbaik untuk masing-masing model, maka berlanjut ke pelatihan menyeluruh dengan hasil pelatihan ditunjukkan oleh tabel 2.

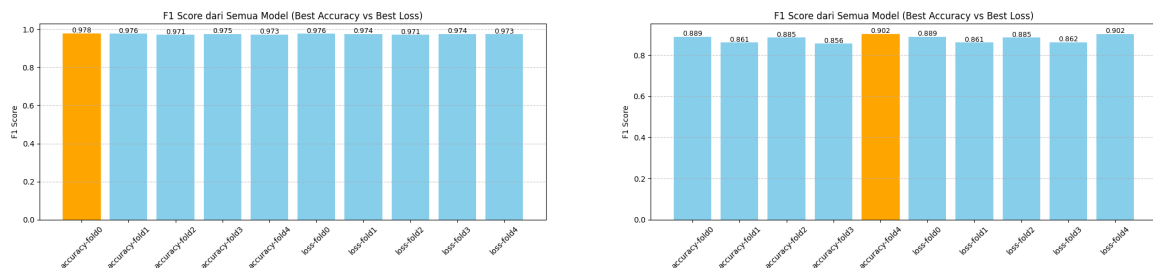
Tabel 2. Perbandingan hasil evaluasi nilai α sebagai nilai *ensemble weight*.

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Model Karakter	Epoch	21	19	13	19	15
	Akurasi	0,978	0,978	0,970	0,975	0,974
	Loss	0,088	0,120	0,101	0,094	0,099
Model Kata	Epoch	25	20	30	27	27
	Akurasi	0,978	0,978	0,970	0,975	0,974
	Loss	0,088	0,120	0,101	0,094	0,099

Menurut tabel 2 rata-rata epochs pelatihan model karakter berjalan selama 17,4 epochs. Sementara itu, rata-rata epochs pelatihan model kata berjalan selama 25,8 epochs. Dari data pelatihan diatas menunjukkan perbedaan kompleksitas dataset digunakan oleh kedua model. Dataset kata memiliki 220 kelas dengan jumlah rata-rata per-kelas memiliki 15 data. Hal ini menunjukkan level kompleksitas yang berbeda jika dibandingkan dataset karakter yang hanya memiliki 32 kelas dengan rata-rata per-kelas memiliki 50 data.

Pemilihan model terbaik yang akan digunakan sebagai model ensemble dilakukan melauai perbandingan nilai *F1-score* yang dihasilkan setiap fold. Dikarenakan adanya fungsi *early stopping* pada proses pelatihan model untuk metrik akurasi tertinggi dan loss terendah, maka setiap fold dari pelatihan menghasilkan 2 model yang bisa dievaluasi. Perbandingan nilai *F1-score* untuk pemilihan model karakter dan model kata ditampilkan oleh gambar 8.

Dari gambar 8 terlihat nilai tertinggi dari evaluasi dihasilkan oleh model akurasi tertinggi fold ke-1 sebesar 0,978. Sementara itu, ilai tertinggi dari evaluasi dihasilkan oleh model akurasi tertinggi fold ke-5 sebesar 0,902. Secara keseluruhan, rata-rata *F1-score* evaluasi model karakter adalah sebesar 0,9741. Sementara itu, secara keseluruhan, rata-rata *F1-score* evaluasi model kata adalah sebesar 0,8799. Dari perbandingan nilai *F1-score* diatas, model karakter yang dipilih untuk digunakan dalam model ensemble adalah model dengan akurasi tertinggi di fold ke-1. Sementara itu, model kata yang dipilih untuk digunakan dalam model ensemble adalah model dengan akurasi tertinggi di fold ke-5.



Gambar 8. Perbandingan F1 score dari model akurasi tertinggi dan loss terendah untuk model karakter (atas) dan model kata (bawah)


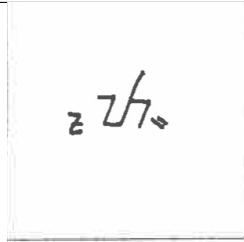

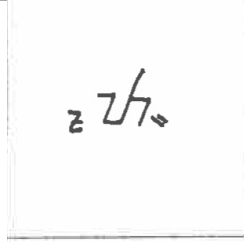
Kedua model terbaik dari proses pelatihan sebelumnya digabungkan menjadi satu model dengan metode confidence voting dengan menggunakan nilai α sebagai *ensemble weight*. Data perbandingan nilai α pada tabel 3 menunjukkan nilai 0,4 dan nilai 0,5 yang bisa menjadi pilihan terbaik. Nilai 0,5 memberikan evaluasi akurasi, presisi dan nilai *F1-score(weighted)* paling tinggi, sementara nilai 0,4 memberikan evaluasi nilai *F1-score (macro)* dan sensitifitas paling tinggi. Perbedaan hasil evaluasi ini disebabkan ada perbedaan jumlah kelas dengan data per-kelas untuk dataset karakter dan dataset kata.

Data perbandingan nilai α diatas menunjukkan nilai 0,4 dan nilai 0,5 yang bisa menjadi pilihan terbaik. Nilai 0,5 memberikan evaluasi akurasi, presisi dan nilai *F1-score (weighted)* paling tinggi, sementara nilai 0,4 memberikan evaluasi nilai *F1-score (macro)* dan recall paling tinggi. Perbedaan hasil evaluasi ini disebabkan ada perbedaan jumlah kelas dengan data per-kelas untuk dataset karakter dan dataset kata.

Tabel 3. Perbandingan hasil evaluasi nilai α sebagai nilai *ensemble weight*.

Nilai α	0,4	0,5	0,6
Akurasi	0,597	0,912	0,843
<i>F1-score(macro)</i>	0,795	0,755	0,380
<i>F1-score(weighted)</i>	0,676	0,910	0,819
Presisi	0,855	0,889	0,571
Sensitifitas	0,827	0,701	0,319

Tabel 4. Perbandingan hasil evaluasi nilai α sebagai nilai *ensemble weight*.

Nilai α	0,4		0,5	
Gambar (input)				
				
Model	Karakter	Kata	Karakter	Kata
Prediksi	ta	téh	ta	téh
<i>Confidence(model)</i>	0,826	0,999	0,826	0,999
<i>Confidence(weighted)</i>	0,495	0,399	0,413	0,499
Prediksi Akhir	ta		téh	

Tabel 4 memberikan ilustrasi kenapa nilai α bernilai 0,4 menghasilkan performa akurasi yang buruk. Kasus ini terjadi karena adanya penambahan satu atau dua rangkén menjadi satu kata yang menurunkan nilai *confidence* dari model karakter. Namun, penurunan tersebut tidak cukup signifikan untuk mengurangi nilai akhir *confidence (weighted)* model karakter agar lebih rendah daripada nilai *confidence (weighted)* dari model kata. Akibatnya, prediksi akhir tetap mengacu pada prediksi model karakter, padahal dalam kasus tersebut prediksi yang lebih diandalkan adalah prediksi dari model kata.

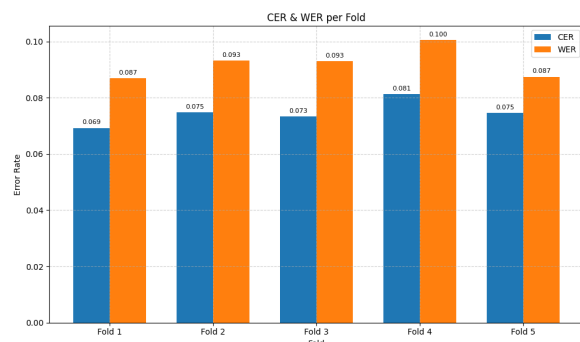
4.2 Model Connectionist Temporal Classification

Pada proses pelatihan model CTC, melalui proses *hypertuning parameter* dengan parameter *dropout* dan *learning rate*. Rentang nilai parameter *dropout* yang dipakai adalah 0,2; 0,3; dan 0,4. Sementara untuk parameter *learning rate* adalah 0,001; 0,0001; dan 0,0005. Dilihat dari table 5 parameter terbaik adalah *dropout* dengan nilai 0,2 dan *learning rate* dengan nilai 0,0001 dengan nilai *loss* terendah sebesar 2,094. Hasil tersebut mengindikasikan bahwa model CTC membutuhkan nilai *learning rate* yang cukup stabil dan membutuhkan *dropout* terkecil untuk menghindari *underfitting*. Model yang dihasilkan setiap fold lalu evaluasi melalui nilai CER dan WER agar menentukan model mana yang terbaik seperti terlihat pada gambar 9.

Dalam penelitian ini, evaluasi dilakukan pada tingkat kata, dan tidak mempertimbangkan kalimat. Gambar 9 menunjukkan nilai metrik WER lebih tinggi daripada metrik CER pada setiap fold. Hal ini disebabkan jika 1 huruf saja salah, maka sudah pasti satu kata dianggap salah dalam evaluasi WER. Setiap fold menghasilkan CER dan WER yang cukup konsisten dengan variasi yang tidak terlalu besar. Model CTC dengan performa terburuk dihasilkan oleh fold ke-4 dengan nilai CER sebesar 0,081 dan nilai WER sebesar 0,100. Model CTC terbaik dihasilkan oleh fold ke-3 dengan nilai CER sebesar 0,069 dan nilai WER sebesar 0,087. Rata-rata nilai CER adalah sebesar 0,074 dan rata-rata

Tabel 5. Hasil pengaturan hyperparameter untuk model CTC.

Percobaan	<i>Dropoout</i>	<i>Learning Rate</i>	<i>Loss</i>
1	0,2	0,001	4,820
2	0,3	0,001	4,682
3	0,4	0,001	4,616
4	0,2	0,0001	2,094
5	0,3	0,0001	2,211
6	0,4	0,0001	2,422
7	0,2	0,00005	3,661
8	0,3	0,00005	3,466

**Gambar 9.** Perbandingan hasil evaluasi CER dan WER setiap fold

nilai WER adalah sebesar 0,092.

4.3 Analisis Akhir

Model karakter dilatih dengan *hyperparameter* berupa *dropout* sebesar 0,4 dan *learning rate* sebesar 0,001. Nilai *F1-score* rata-rata hasil pelatihan model karakter adalah 0,974 dengan model terbaik menunjukkan nilai *F1-score* sebesar 0,978. Sementara itu, model kata dilatih dengan *hyperparameter dropout* sebesar 0,2 dan *learning rate* sebesar 0,001. Nilai *F1-score* rata-rata hasil pelatihan model kata adalah 0,880 dengan model terbaik menunjukkan nilai *F1-score* sebesar 0,902.

Perbedaan nilai *dropout* pada kedua model tersebut terjadi karena adanya perbedaan dataset yang digunakan. Dataset karakter memiliki distribusi data yang lebih merata dan relatif mencukupi pada 32 kelas yang ada. Sebaliknya, dataset kata terdiri dari 220 kelas dengan rata-rata sekitar 15 data per kelas. Oleh karena itu, model karakter memiliki *dropout* yang lebih tinggi untuk mencegah *overfitting*, sedangkan model kata memiliki *dropout* lebih rendah agar model dapat memanfaatkan seluruh informasi yang tersedia untuk mengimbangi keterbatasan jumlah data.

Kedua model tersebut digabung menjadi model ensemble dengan nilai *ensemble weight* sebesar 0,5. Evaluasi model menghasilkan akurasi sebesar 0,912, *F1-score* sebesar 0,755, presisi sebesar 0,889 dan sensitifitas sebesar 0,701. Selisih dari nilai *F1-score*, presisi dan sensitifitas ke nilai akurasi disebabkan oleh ketimpangan antara dataset karakter dengan dataset kata. Dengan penambahan data secara merata dan jumlah data yang mencukupi di setiap kelas dataset kata, diharapkan nilai *F1-score*, nilai presisi dan nilai sensitifitas akan meningkat menjadi lebih baik.

Pada skenario CTC loss, proses *hypertuning parameter* kombinasi parameter yang menghasilkan loss terendah adalah *dropout* sebesar 0,2 dan *learning rate* sebesar 0,0001. Evaluasi model CTC menunjukkan performa yang baik dengan rata-rata nilai *Character Error Rate* (CER) adalah sebesar 0,074 dan rata-rata nilai *Word Error Rate* (WER) adalah sebesar 0,092. Model CTC yang terbaik mencapai nilai CER sebesar 0,069 dan nilai WER sebesar 0,087. Sedangkan Model CTC dengan performa terburuk menghasilkan nilai CER sebesar 0,081 dan nilai WER sebesar 0,100. Nilai CER dan WER yang relatif rendah ini besar kemungkinan karena evaluasi dilakukan menggunakan dataset pada tingkat karakter dan kata, bukan pada tingkat kalimat. Untuk memperoleh hasil evaluasi yang lebih representatif, disarankan untuk menggunakan dataset tulisan tangan aksara sunda berbasis kalimat.

5. Kesimpulan

Penelitian ini bertujuan untuk mengimplementasikan *Optical Character Recognition* (OCR) tingkat kata tulisan tangan Aksara Sunda menggunakan model berbasis EfficientNet. Model OCR yang dikembangkan menggunakan EfficientNetB0 sebagai model dasar dengan dua dataset, yaitu dataset karakter dan dataset kata. Penelitian ini menghasilkan dua pendekatan model yaitu model ensemble dan model *Connectionist Temporal Classification* (CTC). Model ensemble dengan *ensemble weight* sebesar 0,5 berhasil mencapai akurasi sebesar 0,912, presisi sebesar 0,889, sensitifitas sebesar 0,701, dan *F1-score* sebesar 0,755. Model CTC terbaik berhasil mencapai nilai *Character Error Rate* (CER) sebesar 0,069 dan nilai *Word Error Rate* (WER) sebesar 0,087. Hasil ini menunjukkan penambahan dataset kata sebagai dataset baru membuka peluang pengembangan lebih lanjut dalam sistem OCR Aksara Sunda. Selain itu, performa model CTC yang cukup kompetitif memperlihatkan potensi besar dari penerapan metode ini dalam sistem OCR, tidak hanya Aksara Sunda tapi juga aksara nusantara lainnya. Dalam penelitian selanjutnya penulis menyarankan beberapa hal yang bisa dicoba. Pertama, menambahkan dataset kata lebih banyak dan dikembangkan lagi, baik dari sisi kuantitas kelas yang ada dengan kuantitas data per-kelas. Kedua, implementasi metode CTC Loss agar lebih mendetail. Disarankan agar mencoba metode CTC loss sebagai klasifikasi sekuensial dengan penelitian lebih mendetail. Ketiga, pengujian metode CTC loss dilakukan dengan dataset berbasis kalimat agar evaluasi model lebih representatif. Terakhir, penelitian bisa dilakukan menggunakan arsitektur CNN dan metode-metode lainnya.

Daftar Pustaka

- [1] Kementerian Pendidikan dan Kebudayaan Republik Indonesia Direktorat Jenderal Kebudayaan. *Statistik Kebudayaan 2023*, volume 8. 4 2023. URL https://budbas.data.kemdikbud.go.id/statistik/isi_55797b4c-197d-4108-a450-0e37cfffef80.pdf.
- [2] Pemerintah Daerah Jawa Barat. Perda no 14 tahun 2014.
- [3] Khairat Hayati, DrMahmud Dwi Sulistiyo, and DrAditya Fiman Ihsan. Klasifikasi aksara sunda dengan metode machine learning klasik. Technical report, 2024.
- [4] M.A. Prameswari, M. Dwi Sulistiyo, and A.F. Ihsan. Classification of handwritten sundanese script via transfer learning on cnn-based architectures. In *Proceedings - 2023 3rd International Conference on Electronic and Electrical Engineering and Intelligent System: Responsible Technology for Sustainable Humanity, ICE3IS 2023*, pages 401–406, 2023. ISBN 9798350327762. doi: 10.1109/ICE3IS59323.2023.10335382.

- [5] R.R. Akram, M. Dwi Sulistiyo, A. Firman Ihsan, P. Eko Yunanto, D. Richasdy, and M. Arzaki. Exploring relu activation functions in cnn for handwritten sundanese script recognition. In *2024 International Conference on Data Science and Its Applications, ICoDSA 2024*, pages 92–98, 2024. ISBN 9798350365351. doi: 10.1109/ICoDSA62899.2024.10652217.
- [6] Yana Aditia Gerhana, Muhammad Farid Padilah, and Aldy Rialdy Atmadja. Comparison of template matching algorithm and feature extraction algorithm in sundanese script transliteration application using optical character recognition. *Jurnal Online Informatika*, 5:73–80, 7 2020. ISSN 2527-9165. doi: 10.15575/JOIN.V5I1.580. URL <https://join.if.uinsgd.ac.id/index.php/join/article/view/580>.
- [7] Pulkit Batra, Nimish Phalnikar, Deepesh Kurmi, Jitendra Tembhurne, Parul Sahare, and Tausif Diwan. Ocr-mrd: performance analysis of different optical character recognition engines for medical report digitization. *International Journal of Information Technology (Singapore)*, 16:447–455, 1 2024. ISSN 25112112. doi: 10.1007/S41870-023-01610-2/TABLES/6. URL <https://link.springer.com/article/10.1007/s41870-023-01610-2>.
- [8] Idin Baidillah, Undang A Darsa, Oman Abdurahman, Tedi Permadi, Gugun Gunardi, Agus Suherman, Taufik Ampera, Harja Santana, Purba Dian, Tresna Nugraha, and Dadan Sutisna. Direktori aksara sunda untuk unicode. Technical report, Pemerintah Provinsi Jawa Barat Dinas Pendidik. Provinsi Jawa Barat, 2008.
- [9] Keras Team. Keras applications. URL <https://keras.io/api/applications/>.
- [10] Jaohar Riza, Shohag Barman, Sadia Ridita, Zaeed Mahmud, and Aditi Bhattacharya. *PhytoCare : A hybrid approach for identifying Rice, Potato and Corn diseases*. PhD thesis, 1 2024.
- [11] Yongshi Jie, Xianhua Ji, Anzhi Yue, Jingbo Chen, Yupeng Deng, Jing Chen, and Yi Zhang. Combined multi-layer feature fusion and edge detection method for distributed photovoltaic power station identification. *Energies*, 13:6742, 1 2020. doi: 10.3390/en13246742.
- [12] Ratnam Dodda, S. Balakrishna Reddy, Azmera Chandu Naik, and Venugopal Gaddam. A study on handwritten text recognition classification using diverse deep learning techniques and computation of ctc loss. *CVR Journal of Science & Technology*, 26:107–11, 6 2024. ISSN 22773916. doi: 10.32377/CVRJST2617.
- [13] Bounour Imane, · Ammour Alae, · Khaissidi Ghizlane, and Mostafa Mrabti. Enhancing arabic handwritten word recognition: a cnn-bilstm-ctc architecture with attention mechanism and adaptive augmentation. *Discover Applied Sciences*, 7:460, 123. doi: 10.1007/s42452-025-06952-z.
- [14] Rayyan Najam and Safiullah Faizullah. Analysis of recent deep learning techniques for arabic handwritten-text ocr and post-ocr correction, 7 2023. ISSN 20763417.
- [15] P Verma and GM Foomani. Improvement in ocr technologies in postal industry using cnn-rnn architecture: Literature review. *International Journal of Machine Learning and Computing*, 12, 9 2022. doi: 10.18178/ijmlc.2022.12.5.1095.
- [16] Xuan Hien Le, Duc Hai Nguyen, Sungho Jung, Minho Yeon, and Giha Lee. Comparison of deep learning techniques for river streamflow forecasting. *IEEE Access*, 9:71805–71820, 2021. ISSN 21693536. doi: 10.1109/ACCESS.2021.3077703.
- [17] Lamia Mosbah, Ikram Moalla, Tarek M. Hamdani, Bilel Neji, Taha Beyrouthy, and Adel M. Alimi. Adocnet: A deep learning ocr for arabic documents recognition. *IEEE Access*, 12:55620–55631, 2024. ISSN 21693536. doi: 10.1109/ACCESS.2024.3379530.
- [18] DR Vaibhav Khatavkar, DR Makarand Velankar, and Sneha Petkar. Segmentation-free connectionist temporal classification loss based ocr model for text captcha classification. 2 2024. URL <https://arxiv.org/pdf/2402.05417>.
- [19] Khin Me Me Chit and Laet Laet Lin. Exploring ctc based end-to-end techniques for myanmar speech recognition. 2021. doi: 10.1007/978-3-030-68154-8_87. URL https://doi.org/10.1007/978-3-030-68154-8_87.
- [20] Szilvia Szeghalmy and Attila Fazekas. A comparative study of the use of stratified cross-validation and distribution-balanced stratified cross-validation in imbalanced learning. *Sensors* 2023, Vol. 23, Page 2333, 23:2333, 2 2023. ISSN 1424-8220. doi: 10.3390/S23042333. URL <https://www.mdpi.com/1424-8220/23/4/2333/htmhttps://www.mdpi.com/1424-8220/23/4/2333>.
- [21] Dominic A. Neu, Johannes Lahann, and Peter Fettke. A systematic literature review on state-of-the-art deep learning methods for process prediction. *Artificial Intelligence Review*, 55:801–827, 2 2022. ISSN 15737462. doi: 10.1007/S10462-021-09960-8/FIGURES/6. URL <https://link.springer.com/article/10.1007/s10462-021-09960-8>.

- [22] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:10691–10700, 5 2019. URL <https://arxiv.org/abs/1905.11946v5>.
- [23] Gilang Sukma Permana, Sofia Sa'idah, and Rita Purnamasari. Deteksi aksara sunda menggunakan metode cnn arsitektur efficientnet. Technical report, Telkom University, 2023.
- [24] Mingxing Tan and Quoc V Le. Efficientnetv2: Smaller models and faster training. 2021. URL <https://github.com/google/>
- [25] Yuting Li, Dexiong Chen, Tinglong Tang, and Xi Shen. Htr-vt: Handwritten text recognition with vision transformer. *Pattern Recognition*, 158:110967, 2 2025. ISSN 0031-3203. doi: 10.1016/J.PATCOG.2024.110967. URL <https://www.sciencedirect.com/science/article/abs/pii/S0031320324007180>.
- [26] Ashish Shetty and Sanjeev Sharma. Ensemble deep learning model for optical character recognition. *Multimedia Tools and Applications*, 83:11411–11431, 1 2023. ISSN 15737721. doi: 10.1007/S11042-023-16018-0. URL <https://dl.acm.org/doi/10.1007/s11042-023-16018-0>.
- [27] Adebayo Rotimi Philip. Fine-tuning depth analysis: Identifying the sweet spot for maximum accuracy in cnns. *International Journal of Scientific Research in Computer Science and Engineering*, 13:48–63, 2025. ISSN 2320-7639. doi: 10.26438/ijsrcse.v13i3.703. URL www.isroset.org<https://doi.org/10.26438/ijsrcse.v13i3.703>.
- [28] Hassan Ali, Noora Shifa, Rachid Benlamri, Aitazaz A. Farooque, and Raziq Yaqub. A fine tuned efficientnet-b0 convolutional neural network for accurate and efficient classification of apple leaf diseases. *Scientific Reports*, 15:1–26, 12 2025. ISSN 20452322. doi: 10.1038/S41598-025-04479-2;SUBJMETA=114,1305,1564,631;KWRD=IMAGE+PROCESSING,MACHINE+LEARNING. URL <https://www.nature.com/articles/s41598-025-04479-2>.