# **BAB III METODE PENELITIAN**

## **Alur Penelitian**

Penelitian akan dilakukan dengan bahasa pemrograman Python dan berbagai *library* yang tertera. Platform yang digunakan untuk mengimplementasikan penelitian adalah Kaggle Notebooks. Untuk mencapai tujuan yang telah dipaparkan pada bab 1. Berikut flowchart penelitian yang dirancang untuk menjelaskan langkah - langkah yang akan diambil di penelitian ini, yang akan didetailkan pada sub bab 3.2 dan seterusnya.

**A white square with black text

Description automatically generated with low confidence**

A picture containing screenshot, diagram, rectangle, line

Description automatically generated

A picture containing text, screenshot, diagram, line

Description automatically generated

## **DATA**

Data yang peneliti gunakan diambil dari kaggle. Kaggle adalah sebuah platform komunitas *data science* yang berisi berbagai sumber dataset, notebook, kompetisi, dan lain sebagainya. Terdapat dua sumber dataset yang peneliti ambil yaitu,

* [Aksara Jawa](https://www.kaggle.com/datasets/phiard/aksara-jawa) | Kaggle (2634) | Sumber pertama
* [Aksara Jawa / Hanacaraka](https://www.kaggle.com/datasets/vzrenggamani/hanacaraka?select=ya) | Kaggle (1579) | Sumber kedua

Data ini akan digabung menjadi satu sebagai dataset yang akan diuji oleh peneliti. Total data yang akan digunakan di dalam penelitian ini adalah 4213 data.

A picture containing text, diagram, sketch, design

Description automatically generated

Gambar 1 Data sumber pertama

A picture containing text, font, black and white, typography

Description automatically generated

Gambar 2 Data sumber kedua

A picture containing text, diagram, font, design

Description automatically generated

Gambar 3 Data gabungan

## **ALAT DAN BAHAN**

1. **SPESIFIKASI PERANGKAT KERAS**

Perangkat yang digunakan pada penelitian ini diambil dari platform notebook kaggle. Kaggle menyediakan jupyter notebook yang bisa digunakan untuk menjalankan kode dengan menggunakan GPU, CPU, dan RAM yang cukup untuk mengimplementasikan *deep learning*. Berikut detail spesifikasinya:

* + - *20 GB of auto-saved disk space*
    - *1 Nvidia Telsa P100 GPU*
    - *2 CPU cores*
    - *13 Gigabytes of RAM*

Lebih lengkapnya dapat dilihat di dokumentasi kaggle: https://www.kaggle.com/docs/notebooks

1. **LIBRARIES**

*Library* yang akan digunakan pada penelitian ini mencakup visualisasi, *preprocessing*, *modeling*, dan evaluasi. Tensorflow, Albumentations, Numpy, Pandas, Matplotlib, Plotly, Sklearn akan digunakan di dalam penelitian ini.

1. **PREPROCESSING**

A picture containing screenshot, diagram, rectangle, line

Description automatically generated

Dari data tersebut akan dilakukan *preprocessing* untuk mengoptimalkan pelatihan model yang digunakan nantinya. Pada penelitian ini, dilakukan berbagai skenario penelitian untuk melihat perbandingan hasil dari berbagai proses yang dilakukan. Pada bagian ini, augmentasi data akan dilakukan sesuai dengan skenario pengujian peneliti.

1. **RESCALE/RESIZE**

Dalam tahap ini citra akan melalui tahap *rescale* ke dalam ukuran 300 x 300. Hal ini dilakukan agar data yang akan digunakan untuk proses *training* seragam. Citra yang masuk ke dalam proses *training* juga diharuskan seragam, dalam data sebelumnya, beberapa citra mempunyai dimensi yang tidak sama dengan citra lainnya. Berikut adalah contoh rescale dari citra dimensi 364 x 300 menjadi 300 x 300.

A picture containing diagram, sketch, line, drawing

Description automatically generated

Gambar 4 Contoh rescale

1. **NORMALIZATION**

Di tahap ini, peneliti akan menggunakan teknik standardisasi pada citra. Hal ini dilakukan untuk mengoptimalkan dan menjaga kualitas data yang akan digunakan pada proses *training* nantinya. Berikut adalah contoh standardisasi data yang dilakukan dengan *library* tensorflow.

A picture containing text, sketch, handwriting, design

Description automatically generated

Gambar 5 Contoh standardisasi

1. **AUGMENTATION**

Dalam tahap ini, dilakukan augmentasi data dengan *config* seperti pada penelitian sebelumnya terkait data citra teks (Rizky et al., 2023). Berikut confignya:

* *Rotation: 15 °*
* *Image Scale: 0.9*
* *Blur Effect*

A picture containing line, font, sketch, white

Description automatically generated

Gambar 6 Contoh augmentasi

1. **DATA SPLITTING**

Peneliti akan melakukan 3 split menjadi *train, validation*, dan *test set* dari total data yang ada. Total data awal adalah 4213, hal ini akan menghasilkan *data spli*t sebesar 3791, 211, 211 secara terurut dari *train, validation,* dan *test set*.

1. **MODELING**

A picture containing text, screenshot, diagram, line

Description automatically generated

Tahap *modeling* akan dilakukan sesuai skenario pengujian yang tertera. Berikut detail dari tiap prosesnya.

1. **DOWNLOAD PRE-TRAINED MODEL**

Semua pre-trained model ini akan diperoleh menggunakan *library* keras tensorflow. *Keras Applications* menyediakan berbagai *pre-trained* model yang dapat digunakan untuk penggunaan lebih lanjut seperti *prediction, feature extraction,* dan *fine-tuning*. Dokumentasi lebih lanjut dapat dilihat di <https://keras.io/api/applications/>. Model yang akan diunduh dari *keras application library* adalah VGG, Inception, dan Xception.

1. **APPLY TRANSFER LEARNING**

Untuk melakukan transfer learning diperlukan kustomisasi pada layer yang diunduh dari *pre-trained* model keras *library*. Dalam beberapa penelitian sebelumnya, *transfer learning* dengan *pre-trained* model VGG dilakukan dengan menambahkan *fully connected layer* dan *output layer* dengan *activation function* softmax (Tammina, 2019).

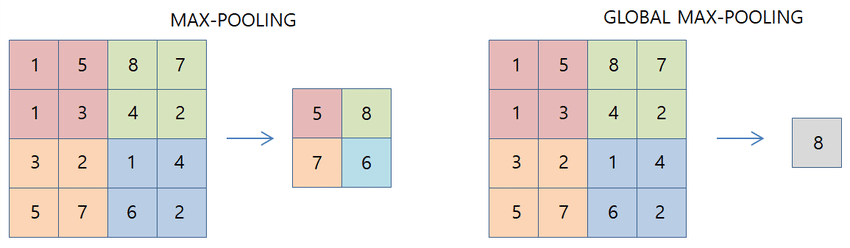
...

A picture containing text, screenshot, font, line

Description automatically generated

Gambar 7 Contoh transfer learning  
(Tammina, 2019)

Untuk menjaga kualitas model dan mencegah *overfitting,* digunakan juga *pooling layer* dan *dropout layer* pada lapisan *fully connected layer*. Ilustrasi dari kedua *layer* tersebut adalah sebagai berikut.



Gambar 8 Contoh pooling  
sumber: <https://www.researchgate.net/figure/The-difference-of-max-pooling-and-global-max-pooling_fig4_338079465>

A picture containing diagram, line, circle, origami

Description automatically generated

Gambar 9 Contoh dropout  
(Zaccone & Karim, 2018)

1. **FINE TUNING**

Definisi dari *fine-tune* menurut *Oxford Languages* adalah membuat penyesuaian kecil (sesuatu) untuk mencapai yang terbaik atau kinerja yang diinginkan. Dalam konteks *deep transfer learning*, hal ini dapat dicapai dengan melakukan *freezing layer, train fully connected layer*, maupun *train all the network* (Iman et al., 2022). Pelaksanaan *fine-tuning* lebih detailnya dapat dilihat pada tabel skenario pengujian.

1. **TRAIN**

Dalam proses ini, dilakukan pelatihan model *transfer learning* dengan data yang sudah dilakukan *preprocessing*. Di dalam *training*, digunakan data *train* dan *validation* untuk mengukur nilai *overfitting* yang terjadi.

1. **EVALUASI**

Evaluasi akan dilakukan dengan metriks akurasi, sebagai tambahan digunakan juga *confusion matrix* untuk melihat kesalahan prediksi yang dilakukan oleh model yang sudah di-*train*. Berikut adalah contoh *confusion matrix* dengan 6 kelas prediksi. Dalam kasus penelitian ini, total kelas adalah 20, sehingga total baris dan kolom masing – masing adalah 20.

A picture containing screenshot, square, rectangle

Description automatically generated

Gambar 10 Contoh confusion matrix  
(Zaccone & Karim, 2018)

1. **SKENARIO PENGUJIAN**

Dalam tahap penelitian akan dilakukan evaluasi terhadap skenario pengujian tanpa melakukan augmentasi data. Jika akurasi dari hasil proses *transfer learning* dengan *fine-tuning freezing layer* dibawah 70% atau terdapat *overfitting*, peneliti akan melakukan augmentasi untuk menyelesaikan masalah tersebut. Skenario secara garis besar adalah seperti tabel dibawah.

|  |  |  |  |
| --- | --- | --- | --- |
| **MODEL** | **AUGMENTASI (*optional*)** | **FREEZE** | **HASIL** |
| VGG | *yes or no* | *full* | ? |
| INCEPTION | *yes or no* | *1/2* | ? |
| XCEPTION | *yes or no* | *none* | ? |

Tabel 1. Skenario Pengujian

**DAFTAR PUSTAKA**

Iman, M., Rasheed, K., & Arabnia, H. R. (2022). A Review of Deep Transfer Learning and Recent Advancements. *Technologies*, *11*(2), 40. https://doi.org/10.3390/technologies11020040

Rizky, A. F., Yudistira, N., & Santoso, E. (2023). *Text recognition on images using pre-trained CNN*. https://arxiv.org/abs/2302.05105v1

Tammina, S. (2019). Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images. *International Journal of Scientific and Research Publications (IJSRP)*, *9*(10), p9420. https://doi.org/10.29322/IJSRP.9.10.2019.P9420

# **DAFTAR PUSTAKA BUKU**

Zaccone, G., & Karim, M. R. (2018). *Deep Learning with TensorFlow: Explore neural networks and build intelligent systems with Python (2nd ed.).* Birmingham, United Kingdom: Packt Publishing.