Journal of Intelligent Systems Technology and Informatics
JISTICS | J. Intell. Syst. Technol. Inform. | Vol. 1, No. 2, July. 2025, pp. 53-61
DOI: https://doi.org/10.64878/jistics.v1i2.34 | ISSN (Online): 3109-757X
Journal homepage: https://journal.aptika.org/index.php/jistics

From Local Features to Global Context: Comparing CNN and Transformer for Sundanese Script Classification

Yoga Agustiansyah[1], Dhika Restu Fauzi[2]

[1,2] Informatics Engineering, Department of Computer Science, Institut Teknologi Garut, Garut, West Java, 44151, Indonesia

Corresponding Author: Yoga Agustiansyah (email: yogaagustiansyah01@gmail.com)
[Received 1 July 2025; Revised 17 July 2025; Accepted 19 July 2025; Available Online 14 September 2025]

ABSTRACT — The digital preservation of historical writing systems like Aksara Sunda is critical for cultural heritage, yet automated recognition is hindered by high character similarity and handwriting variability. This study systematically compares two dominant deep learning paradigms, Convolutional Neural Networks (CNNs) and Transformers, to evaluate the crucial trade-off between model accuracy and real-world robustness. Using a transfer learning approach, we trained five models (ResNet50, MobileNetV2, EfficientNetB0, ViT, and DeiT) on a balanced 30-class dataset of Sundanese script. Performance was assessed on a standard in-distribution test set and a challenging, independently collected Out-of-Distribution (OOD) dataset designed to simulate varied real-world conditions. The results reveal a significant performance inversion. While EfficientNetB0 achieved the highest accuracy of 96.9% on in-distribution data, its performance plummeted on the OOD set. Conversely, ResNet50, despite being lower in in-distribution accuracy, proved to be the most robust model, achieving the highest accuracy of 92.5% on the OOD data. This study concludes that for practical applications requiring reliable performance, the generalization capability demonstrated by ResNet50 is more valuable than the specialized accuracy of EfficientNetB0, offering a crucial insight for developing robust digital preservation tools for historical scripts.

KEYWORDS — Aksara Sunda, Convolutional Neural Network, Deep Learning, Transfer Learning, Vision Transformer.

1. INTRODUCTION

Aksara Sunda is a traditional writing system that represents a crucial part of the cultural and intellectual heritage of the Sundanese people in Indonesia. Its historical use, evidenced in inscriptions and manuscripts dating back to the 14th century, saw a significant decline due to colonial-era policies that promoted the use of Latin, Pegon Arabic, and Cacarakan scripts [1]. Recognizing this, the government has made efforts to preserve and revitalize Aksara Sunda, including its integration into the local school curriculum in West Java and Banten [2]. In the digital age, the survival and relevance of this script depend on its successful transition into digital platforms, a process that presents both challenges and opportunities [3].

The development of accurate automated character classification systems is a cornerstone of this digital preservation effort. However, applying this technology to traditional scripts presents unique challenges that are less prevalent with modern scripts like Latin. These challenges include high intra-class variability, where a single character can have numerous stylistic variations in handwriting; high inter-class similarity, where different characters are visually very similar; and the persistent issue of data scarcity, as datasets for traditional scripts are far more limited than their modern counterparts [4]. Furthermore, historical documents often suffer from image degradation, such as faded ink and background noise, which complicates the classification task [5].

To address these challenges, researchers have explored various deep learning architectures. Convolutional Neural Networks (CNNs) have become the standard for image classification tasks, demonstrating strong performance in recognizing multiple Indonesian scripts, including Sundanese [6], Javanese [7], and Balinese [8]. More recently, the Transformer architecture, originally from the field of Natural Language Processing, was adapted for vision tasks in the form of the Vision Transformer (ViT) [9]. ViT's global self-attention mechanism has shown promise for recognizing other complex scripts like Arabic and Devanagari [10]. However, while these studies validate the potential of individual models, a direct, systematic comparison to evaluate the crucial trade-off between their accuracy and real-world robustness for Aksara Sunda is still lacking. When faced with an Out-of-Distribution (OOD) domain shift, a critical test for practical applicability, the performance disparity between these architectures remains a particularly unaddressed research gap. This study aims to fill this void by providing a benchmark of modern CNN and Transformer models, offering crucial insights for developing robust digital preservation tools.

2. METHODOLOGY

This study employs an experimental methodology to systematically compare the performance of five different deep learning models on the Sundanese script image classification task. As illustrated in Figure 1, the research workflow consists of five main stages: data acquisition, data preprocessing, modeling, training, evaluation, and comparative analysis.

The process commences with acquiring a Sundanese script image dataset from various public sources. This raw dataset then undergoes an essential preprocessing stage, encompassing class balancing to address data bias and splitting into training, validation, and test sets. The models under investigation are constructed using a transfer learning approach, where a uniform, custom classification head is

ISSN 3109-757X
https://doi.org/10.64878/jistics.v1i2.38

appended to a pre-trained model backbone. The training process is conducted in two phases (feature extraction and fine-tuning) to optimize the model's adaptation to the specific Sundanese script data. The final stage involves a quantitative evaluation using a suite of standard metrics on both the test set and an Out-of-Distribution (OOD) set, the results of which form the basis for the comparative analysis.

All experiments, from data preprocessing to model training and evaluation, were conducted on the Kaggle Notebooks cloud computing platform. This computational environment was equipped with 29 GB of Random Access Memory (RAM) and accelerated by a single Tesla P100 Graphical Processing Unit (GPU) with 16 GB of VRAM to accommodate the intensive computational demands of deep learning model training.

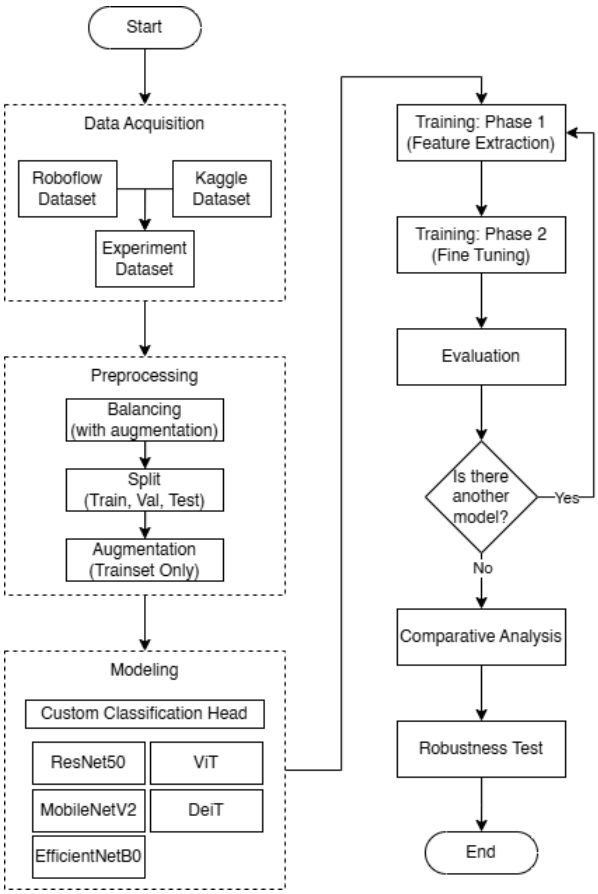

Figure 1. research workflow

2.1 Dataset

The dataset used in this study consists of images of individual Sundanese characters. After preprocessing and balancing, the dataset was partitioned into training and evaluation.

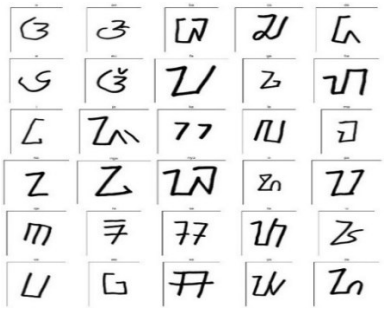

Figure 2. Dataset Sample

Figure 2 shows a sample of the images from the dataset, illustrating the variations in handwriting style and image quality that the models must handle.

2.2 Model Architectures

Five pre-trained models were selected to represent a broad spectrum of modern computer vision paradigms. These include ResNet50 as a classic deep CNN architecture, MobileNetV2 and EfficientNetB0 as representatives of highly efficient CNNs designed for mobile constraints, and Vision Transformer (ViT) and Data-efficient Image Transformer (DeiT) as the foundational and data-efficient examples of the attention-based paradigm.

- ResNet50: The Residual Network (ResNet) architecture was introduced to address the degradation problem, where accuracy saturates and rapidly declines as networks become excessively deep. Its core innovation is the "residual connection" or skip connection, which allows gradients to flow more easily during backpropagation [11]. These residual blocks modify the function being learned by a stack of layers. Instead of learning a direct mapping H(x), the block learns a residual function as formulated in Equation (1).

$$F(x) = H(x) - x \qquad (1)$$

  Intuitively, it is easier for the network to push the residual function F(x) towards zero than to learn an identity mapping H(x)=x through a stack of non-linear layers. This significantly facilitates the optimization of intense networks.
- MobileNetV2: MobileNetV2 is specifically designed for applications on resource-constrained devices, such as mobile phones, by introducing two key innovations[12]. The first is depth-wise separable convolutions, which factorize a standard convolution into two steps: a depth-wise convolution (applying a single filter per input channel) and a pointwise convolution (a 1×1 convolution to combine the outputs). This process dramatically reduces computational cost and the number of parameters. The second is the inverted residual block with a linear bottleneck. This contrasts with standard residual blocks that first narrow and then expand. Instead, an inverted residual block first expands the feature representation using a 1x1 convolution. It then applies a lightweight depth-wise convolution in this higher-dimensional space. Finally, it returns the features to a low-dimensional representation (the bottleneck) with a linear 1x1 convolution, which notably omits a non-linear activation. This linear bottleneck is crucial to prevent the loss of important information in low-dimensional spaces.
- EfficientNetB0: A key advantage of this CNN model lies in its systematic and efficient approach to architecture scaling, rather than in novel block design. It introduces compound scaling, a method that uniformly scales all three dimensions of a convolutional network—depth, width, and image resolution—using a single scaling parameter, ϕ [13]. This scaling relationship is formulated in Equation (2).

$$\mathrm{d} = \alpha^{\phi}, \quad w = \beta^{\phi}, \quad r = \gamma^{\phi} \qquad (2)$$

ISSN 3109-757X
https://doi.org/10.64878/jistics.v1i2.38

This approach is based on the premise that increasing input resolution requires a deeper network to expand the receptive field and a wider network to capture more complex feature patterns. By scaling these three dimensions concurrently, EfficientNet achieves significant improvements in accuracy and efficiency compared to arbitrary, single-dimension scaling approaches.

- Vision Transformer (ViT): ViT adapts the successful Transformer architecture from NLP for vision tasks [14]. ViT represents a paradigm shift from convolution to attention. The process begins by splitting an input image into a sequence of non-overlapping, fixed-size patches. These patches are then flattened and linearly projected into embedding vectors. To retain spatial information lost during the flattening process, learnable positional embeddings are added to each patch embedding [15]. The core of ViT is the Multi-Head Self-Attention mechanism, which allows the model to weigh and integrate information from all other patches in the image simultaneously [16]This provides a global receptive field from the very first layer, enabling the capture of long-range dependencies between image parts, a feat challenging for CNNs in their initial layers [15].
- Data-efficient Image Transformer (DeiT): DeiT is designed to address a key weakness of ViT: its reliance on massive-scale pre-training datasets. DeiT achieves data efficiency through knowledge distillation, a strategy where a "student" model (DeiT) learns from a more powerful "teacher" model (often a pre-trained CNN) [17]. DeiT's primary innovation is the introduction of a distillation token. This additional, learnable vector is passed to the input sequence alongside the standard patch tokens and class token. During training, the class token learns to predict the ground truth label. In contrast, the distillation token is specifically trained to replicate the predictions (either "soft" or "hard" labels) of the teacher model. These tokens interact through the self-attention mechanism, allowing the DeiT model to effectively absorb beneficial inductive biases from the teacher, thereby significantly boosting its performance on smaller datasets.

To conduct a comprehensive comparison, five pre-trained models were strategically selected to represent a broad spectrum of modern computer vision paradigms. The selection was justified: ResNet50 was chosen as a benchmark for classic, deep CNN architectures known for their strong feature extraction capabilities. MobileNetV2 and EfficientNetB0 were selected as representatives of highly efficient CNNs, specifically designed to balance performance with computational constraints, making them ideal for mobile and edge devices. On the other hand, Vision Transformer (ViT) represents the foundational shift towards attention-based mechanisms in vision. At the same time, Data-efficient Image Transformer (DeiT) was included to evaluate a state-of-the-art solution to overcome ViT's heavy reliance on large-scale pre-training datasets.

All models were constructed consistently: a pre-trained backbone (from Keras Applications or Hugging Face Transformers) with weights derived from ImageNet pre-training, followed by a uniform, custom classification head. This head consists of a Global Average Pooling 2D layer (for CNNs) or a token extraction layer (for Transformers), followed by a Dense layer with 128 neurons (ReLU activation), a Dropout layer with a rate of 0.5 for regularization, and a final Dense output layer with 30 neurons (corresponding to the number of classes) and a Softmax activation function [18]. All models were trained using the Adam optimizer with a batch size 32. A two-phase training protocol was applied to maximize the benefits of transfer learning:

- Phase 1 (Feature Extraction): The pre-trained backbone was frozen, and only the custom classification head was trained for 20 epochs with a learning rate of $1e^{-3}$. This initial phase is critical as it allows the new, randomly initialized head to adapt to the feature distribution of the Sundanese script dataset without the risk of large, erroneous gradients corrupting the powerful, pre-trained weights.
- Phase 2 (Fine-Tuning): Once the head was stabilized, the entire model (except Batch Normalization layers) was unfrozen and trained for 10 epochs using a much lower learning rate $1e^{-5}$. This fine-tuning phase allows the model to gently adjust the pre-trained features better to suit the specific nuances of the new data, leading to a more specialized and higher-performing model without undoing the foundational learning.

2.3 Evaluation Protocol

To comprehensively assess model performance, this study employed a multi-faceted evaluation protocol to measure both predictive accuracy under ideal conditions and generalization capability in real-world scenarios.

The primary evaluation was conducted on a held-out, in-distribution test set. Model performance was quantified using a standard classification metric derived from the confusion matrix, which compares predicted labels against the ground truth. These metrics include:

- Loss, specifically Categorical Cross-Entropy, measures the dissimilarity between the predicted probability distribution and the actual label. A lower loss value indicates a more confident and accurate model.
- Accuracy, the most straightforward metric, calculates the ratio of correctly classified instances to the total number of instances [19].
- Precision, which measures the model's ability to avoid false positives. It answers the question: "Of all instances the model predicted as a certain class, how many were correct?"
- Recall (or Sensitivity) measures the model's ability to identify all relevant class instances (i.e., avoid false negatives). It answers: "Of all actual instances of a class, how many did the model correctly identify?"
- F1-Score provides a single, balanced measure of a model's performance by calculating the harmonic mean of Precision and Recall. It is beneficial when there is an uneven class distribution, but it also serves as a robust overall performance indicator.

Beyond standard metrics, a crucial secondary evaluation was performed to assess model robustness—the ability to maintain performance under a domain shift. Standard cross-validation on a single dataset can often lead to an overly optimistic assessment of a model's real-world utility. We

conducted tests on a new, independently collected Out-of-Distribution (OOD) dataset to address this. This dataset, comprising 360 images captured under various lighting conditions, backgrounds, and handwriting styles, was intentionally designed not to follow the same distribution as the training data. This OOD evaluation serves as a stress test, providing a more honest and realistic measure of how each model will likely perform when deployed in uncontrolled, practical environments.

3. RESULT AND DISCUSSION

3.1 Dataset Acquisition and Preprocessing

The initial dataset, aggregated from public sources on Kaggle [20] and Roboflow [21]Comprised 30 classes of Aksara Sunda. An initial analysis revealed a class imbalance, as shown in Figure 3(a), with sample counts ranging from 141 to 168 images per class. A balancing process was performed using augmentative oversampling to prevent the models from becoming biased towards the majority classes. Each class with fewer than 168 samples (the maximum sample count) was augmented with new images until all 30 classes uniformly contained 168 samples. Augmentation was performed by applying a series of random transformations. Specifically, brightness levels were randomly adjusted by a factor between 0.6 and 1.4 (with a 70% probability of application), contrast was similarly adjusted within the same factor range (also with a 70% probability), and Gaussian noise, with a standard deviation (sigma) randomly selected between 5 and 20, was added to images (with a 50% probability). This resulted in a balanced dataset as depicted in Figure 3(b).

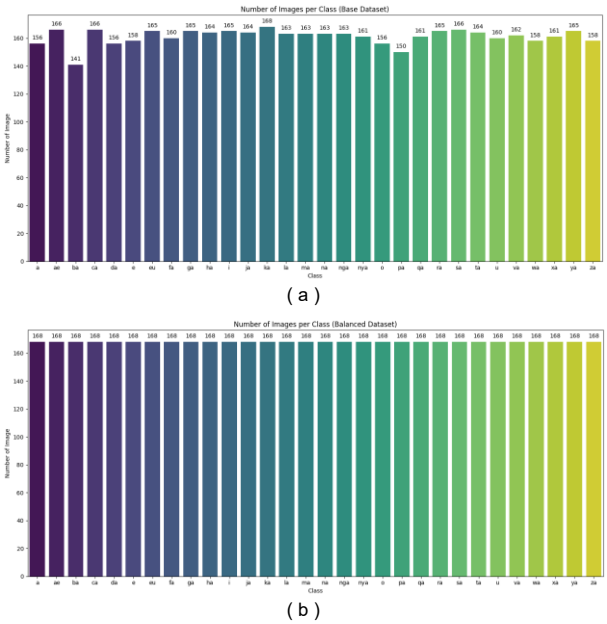

Figure 3. Class distribution of the Sundanese script dataset. (a) The initial imbalanced dataset had image counts per class ranging from 141 to 168. (b) After augmentative oversampling, the balanced dataset ensures a uniform distribution of 168 images per class.

After balancing, the total of 5,040 images was partitioned into three sets with a 70:15:15 ratio for training, validation, and testing, respectively. This resulted in 3,510 training images, 750 validation images, and 780 test images. During training, additional online augmentation techniques, such as rotation (±10°), shifting, shearing, and zooming, were dynamically applied to the training data to increase variance and improve the models' robustness to minor geometric transformations.

3.2 Model Architecture and Training Dynamics

All five model backbones were connected to an identical custom classification head to ensure a fair comparison. This head consisted of a Global Average Pooling layer (for CNNs) or a CLS Token extraction layer (for Transformers), followed by a Dense layer (128 units, 'relu' activation), a Dropout layer (0.5 rate), and a final Dense output layer (30 units, 'softmax' activation).

The training process revealed distinct dynamics for each model:

- ResNet50: As shown in Figure 4(a), ResNet50 struggled significantly during the initial training phase. Its validation accuracy remained very low in Phase 1 (feature extraction), failing to surpass 30%. Performance improved slowly only during Phase 2 (fine-tuning), where all layers were trained to adapt more deeply, eventually reaching a validation accuracy of approximately 92%
- MobileNetV2 & EfficientNetB0: Both models demonstrated rapid convergence in Phase 1, as shown in Figures 4(b) and 4(c). They achieved relatively high validation accuracies of 83-84% even before fine-tuning, confirming the efficiency of their architecture. In the fine-tuning phase, both models experienced a significant jump in validation accuracy to around 93-95% in the first epoch, with subsequent epochs yielding 0.1-0.3% marginal gains.
- ViT: Although not as fast to converge as MobileNetV2 and EfficientNetB0, the Vision Transformer showed better initial improvement than ResNet50 in Phase 1 (Figure 4(d)). Its accuracy started low but increased significantly after several epochs, a common behavior for Transformers that require more time to learn spatial relationships from scratch. It concluded Phase 1 with a validation accuracy of approximately 79%. Like the other models, it experienced a significant performance leap during fine-tuning, though improvements were slower (less than 0.1% per epoch), reaching a final validation accuracy of 91%.
- DeiT: The performance trajectory of DeiT (Figure 4(e)) was very similar to that of MobileNetV2 and EfficientNetB0. At the end of the feature extraction phase, DeiT achieved a validation accuracy of 86%. It jumped to 95% at the start of fine-tuning, but like ViT, its subsequent improvements were very slow, making its performance appear stagnant on the graph.

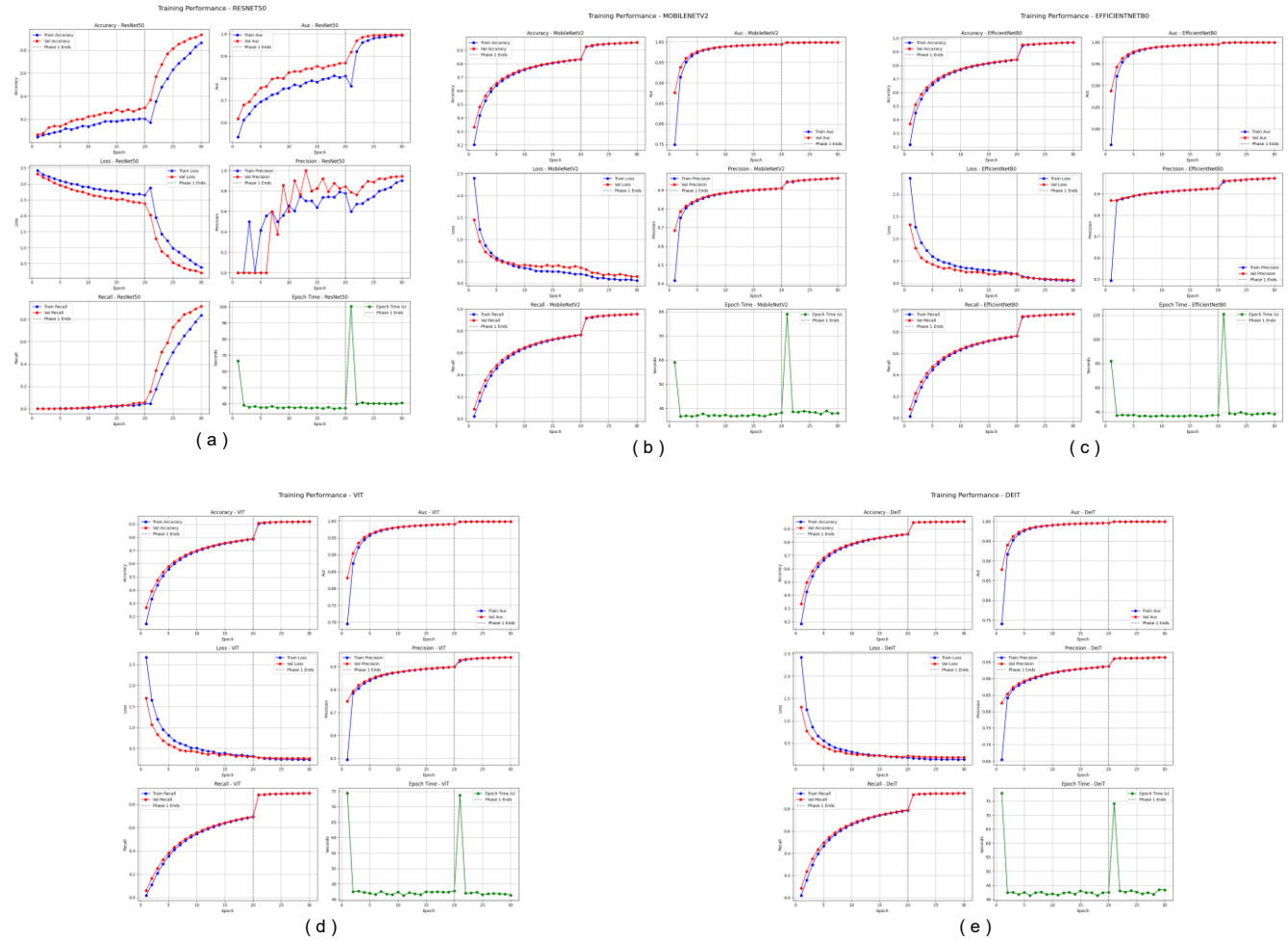

Figure 4. Training dynamics of the five evaluated models. Each image group (a-e) shows the performance progress on the training and validation data over time. The plots are displayed in the following order: Accuracy (top-left), AUC (top-right), Loss (middle-left), Precision (middle-right), Recall (bottom-left), and Epoch Time (bottom-right). Each group corresponds to the respective model: (a) ResNet50, (b) MobileNetV2, (c) EfficientNetB0, (d) ViT, and (e) DeiT.

3.3 Performance On In-Distribution Test Data

The final evaluation stage was conducted on a test set of 780 images to assess the models' generalization capabilities on data unseen during training. The quantitative results in Table 1 show a clear performance hierarchy among the tested models.

EfficientNetB0 convincingly emerged as the top-performing model. With the lowest loss of 0.0776, it demonstrated the highest prediction confidence. This superiority extended across all evaluation metrics: an accuracy of 96.99%, a precision of 97.40% (the ability to avoid mislabelling negative classes as positive), and a recall of 96.56% (the ability to find all positive samples). This exceptional balance between precision and recall is reflected in its peak F1-score of 96.97%.

The second tier consists of MobileNetV2 and DeiT, which achieved F1-scores above 95%. As a CNN representative, MobileNetV2 (F1-Score 95.88%) slightly outperformed the Transformer-based DeiT (F1-Score 95.34%), primarily due to its lower loss. The following position was occupied by ResNet50, which showed robust and consistent performance in the 93-94% range for most metrics. Finally, the baseline ViT model was the key keeper in this in-distribution test. Although its performance was the lowest, ViT achieved a respectable accuracy of 91.90% and an F1-score of 91.78%.

Overall, on test data identically distributed to the training data, CNN architectures, particularly those designed for efficiency like EfficientNetB0 and MobileNetV2, demonstrated clear superiority over the Transformer architectures.

Table 1. Performance Comparison on In-Distribution Test Set

Architecture	Model	Loss	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CNN	EfficientNetB0	0.0776	96.99	97.40	96.56	96.97
CNN	MobileNetV2	0.1087	95.79	96.41	95.36	95.88
Transformer	DeiT	0.1651	95.46	96.39	94.32	95.34
CNN	ResNet50	0.1625	93.72	94.16	93.08	93.61
Transformer	ViT	0.2325	91.90	94.03	89.64	91.78

We analyzed their confusion matrices for a more profound, qualitative understanding of the models' behavior (see Figure 5). This analysis allows us to move beyond aggregate metrics and identify each architecture's unique error patterns, strengths, and weaknesses. Several key examples of these challenging pairs are visualized in Figure 6.

EfficientNetB0, the top-performing model (Figure 5c), exhibited a nearly flawless matrix. Its errors were minimal and confined mainly to pairs with extreme visual similarity, such as misclassifying 'fa' as 'pa' (2 instances), a challenge highlighted in Figure 6a.

The next tier of performance was occupied by MobileNetV2 and DeiT (Figures 5b and 5e, respectively), whose matrices revealed remarkably similar characteristics. Both models were highly accurate but shared a specific, significant weakness: an intense one-way confusion where the character 'ae' was frequently misclassified as 'a' (5 instances for MobileNetV2, 4 for DeiT). This difficulty in detecting the small diacritic that distinguishes the two characters is illustrated in Figure 6c.

In contrast, ResNet50 (Figure 5a) presented a unique and telling profile. While its overall performance was solid, it suffered from a critical, standout flaw: a severe and reciprocal confusion between the characters 'wa' and 'ta' (see Figure 6b). It misclassified 'wa' as 'ta' 6 times and 'ta' as 'wa' 3 times, the highest error concentration on a single pair across all models.

Finally, the baseline Vision Transformer (ViT) (Figure 5d) produced the most distinct error patterns. Beyond major confusions with visually similar pairs, such as the misclassification of 'ba' as 'na' (6 instances), ViT was prone to less intuitive misclassifications. For instance, it frequently confuses 'nya' with 'a (3 instances), as Figure 6d shows two characters with limited visual resemblance to a human observer.

This detailed analysis of the confusion matrices confirms that a high-level accuracy score does not tell the whole story. The specific error patterns reveal the architectural "personality" of each model, from the understandable mistakes of efficient CNNs to the particular flaws of ResNet50 and the less intuitive errors of a pure Transformer. This qualitative insight is crucial for selecting the right model for a specific application, balancing overall performance with potential, predictable failure modes.

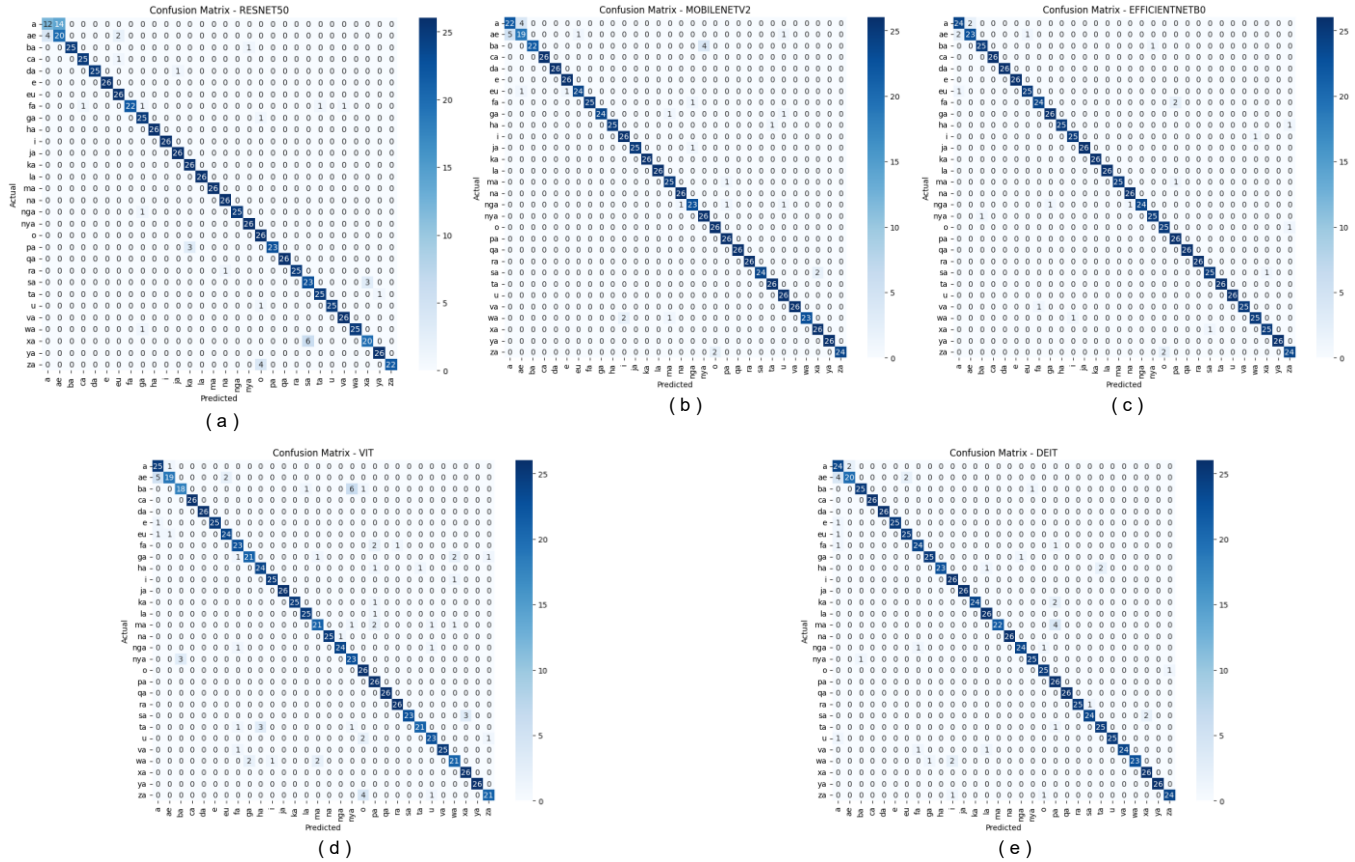

Figure 5. Confusion matrices for each model on the in-distribution test set. The main diagonal represents correct predictions, while off-diagonal cells indicate misclassifications, with darker shades signifying higher confusion rates. The matrices correspond to: (a) ResNet50, (b) MobileNetV2, (c) EfficientNetB0, (d) ViT, and (e) DeiT.

To visually contextualize the key challenges identified from the confusion matrices, Figure 6 displays four of the most representative misclassification pairs. These images provide direct evidence for the different types of errors encountered. For instance, the pairs 'fa' and 'pa' (Figure 6a) illustrate a universal challenge stemming from high similarity in the core character shape, which affected nearly all models. In contrast, the confusion between 'wa' and 'ta' (Figure 6b) highlights a severe architectural flaw within ResNet50. The challenge of detecting small but critical diacritics is shown in the 'a' versus 'ae' pair (Figure 6c), a common failure mode.

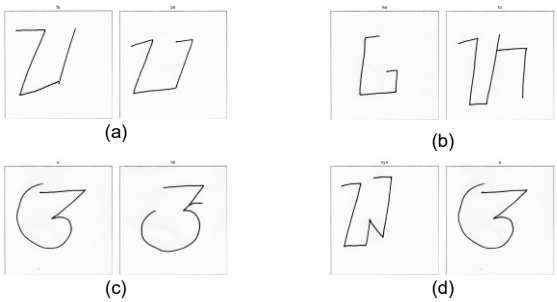

Figure 6. Visual examples of representative misclassification pairs, highlighting different error types.
(a) Universal challenge from similar core shapes ('fa' vs. 'pa').

(b) Model-specific confusion ('wa' vs. 'ta' for ResNet50).
(c) Common failure to detect diacritics ('a' vs. 'ae').
(d) Less intuitive error unique to the baseline ViT ('nya' vs. 'a').

Finally, the less intuitive misclassification of 'nya' as 'a' (Figure 6d) exemplifies the unique error pattern of the baseline Vision Transformer, likely due to its lack of strong spatial biases.

3.4 Robustness Test on Out-of-Distribution Data

The generalization ability of the models was tested on an OOD dataset comprising 360 images. This dataset was also balanced, containing exactly 12 images for each of the 30 classes, to ensure an unbiased robustness evaluation. This dataset was collected independently to simulate real-world conditions. The creation process involved manually writing each character on paper, which was then digitized using a flatbed scanner at a high resolution of 600 DPI. This method produced technically clean data but with natural variations in handwriting style distinct from the training distribution. A sample of this robustness test data can be seen in Figure 7.

The creation process for the OOD dataset was designed to simulate a typical, practical use case: the digitization of physical documents or real-time recognition of handwritten text. By manually writing each character and scanning it at a high resolution (600 DPI), we aimed to produce technically clean images (low noise, uniform background) but contain natural, authentic variations in human handwriting that are inherently different from the more curated public datasets. This setup effectively tests the models' ability to handle natural stylistic deviations, a crucial aspect of robustness for any real-world OCR application.

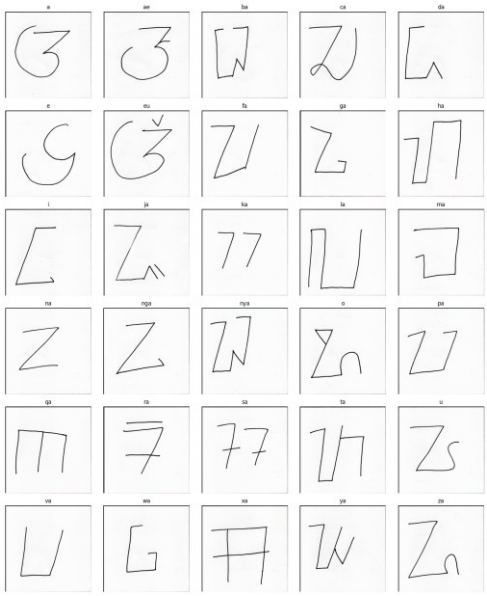

Figure 7. Data Sample for OOD Test

The results of this OOD test revealed a fascinating performance shift. The data in Table 2 uncovers the most significant finding of this study: a performance inversion.

Although EfficientNetB0 was the champion on the in-distribution test set with 97.0% accuracy, its performance degraded significantly to 84.2% on the OOD set. This indicates that despite its high accuracy, the model tends to be brittle and less capable of generalizing to unexpected variations.

Conversely, ResNet50, a mid-tier performer on the in-distribution test set (93.7% accuracy), emerged as the most robust model on the OOD set. It exhibited a much smaller performance drop and achieved the highest accuracy of 92.5%.

Table 2. Model Robustness and Efficiency On Out-of-Distribution Data

Model	Accuracy	F1-Score	Inference Time (ms)
ResNet50	92.50	91.58	80.93
EfficientNetB0	84.16	82.97	75.53
ViT	81.66	81.16	80.16
MobileNetV2	79.16	76.21	74.43
DeiT	75.00	71.57	79.83

This phenomenon highlights a critical trade-off. With its compound scaling, highly optimized architectures like EfficientNet may have over-specialized to the specific training data distribution. Its ability to capture subtle patterns in the training data became a disadvantage when faced with new, slightly different data. On the other hand, the simpler and more constrained architecture of ResNet50, built upon the strong inductive bias of residual connections, appears to learn more fundamental and less dataset-specific features. This allowed it to generalize better to previously unseen variations. The Transformer models (ViT and DeiT), which lack strong spatial inductive biases, also showed greater difficulty handling the domain shift than ResNet50.

Inference time analysis from the OOD test adds another dimension to this trade-off. The lightweight CNN models (MobileNetV2 and EfficientNetB0) proved to be the fastest, with average inference times of around 74-75 ms per sample. ResNet50 and the Transformer models were slightly slower, at around 80 ms. This implies that the most robust model (ResNet50) is not the most computationally efficient, presenting a complex design choice for application developers.

4. DISCUSSION

The most significant finding of this study is the performance inversion observed between the in-distribution and out-of-distribution tests. For real-world applications, such as a mobile app for tourists or a learning tool for students using their camera to recognize Aksara Sunda under various lighting and background conditions, model robustness is far more valuable than peak laboratory accuracy. A model that can maintain reasonable performance and "fail gracefully" when faced with unexpected new data is more reliable than a highly accurate but brittle model prone to catastrophic failure when encountering a slight domain shift (a weakness exhibited by EfficientNetB0 in this context).

This result can be interpreted through the lens of architectural foundations. The strong, inherent spatial inductive biases of CNN architectures, particularly in classic designs like ResNet50, likely contribute to their superior generalization on the OOD set. The hierarchical structure and limited receptive fields compel the model to learn fundamental local features (such as strokes, curves, and corners) that tend to be invariant across different conditions.

Conversely, while powerful, the flexibility and global attention of Transformer architectures can lead to overfitting on relatively small and less diverse datasets like the one used in this study. Without strong spatial biases, Transformer models may be more inclined to "memorize" specific global patch configurations from the training data rather than learning generalizable component features. This is evident in ViT's lower performance. The better performance of DeiT suggests

that knowledge distillation can effectively "inject" some of the beneficial inductive biases from a teacher model (often a CNN), thus bridging the gap between the two architectural paradigms.

The accuracy achieved by the top-performing models in this study (e.g., 97.0% for EfficientNetB0, 95.8% for MobileNetV2, and 95.5% for DeiT on in-distribution test data) is highly competitive. It aligns with other studies on recognizing Sundanese and related scripts like Javanese, reporting accuracy above 95%[22]. This validates that the experimental setup, including data preprocessing and training strategies, is solid and capable of producing state-of-the-art results. However, the key differentiator of this research is the OOD testing. By explicitly measuring and reporting the performance degradation due to domain shift, this study provides a more realistic benchmark and urges researchers in the field to adopt more rigorous evaluation protocols beyond simple cross-validation on a single dataset.

These technical findings can be translated into practical recommendations for developers. For those aiming to create a robust, all-purpose Sundanese script recognition application, ResNet50 presents the most reliable starting point, despite not having the highest "on-paper" accuracy. Conversely, for applications where the input data can be highly controlled, such as in archival digitization projects where manuscripts are scanned under uniform lighting and resolution, the peak performance of EfficientNetB0 might be preferable. The choice of architecture, therefore, should be driven by the intended use case, with a clear understanding of the trade-off between accuracy and robustness.

While the models achieve high overall accuracy, the qualitative analysis reveals persistent challenges with high inter-class similarity, exemplified by the 'fa'–'pa' and 'a'–'ae' pairs. Addressing these specific weaknesses is a critical next step for improving performance. Future work could explore several advanced techniques to mitigate these issues.

One promising direction is the adoption of Fine-Grained Visual Classification (FGVC) methods, which are specifically designed to distinguish between highly similar categories by forcing the model to focus on subtle, discriminative regions [23]. Furthermore, specialized data augmentation and loss functions could be employed. Instead of generic transformations, one could generate samples emphasizing the differences between confused pairs. On the training side, moving beyond standard cross-entropy to metric learning-based losses, such as Triplet Loss or Contrastive Loss, could be highly effective [24]. These functions are designed to minimize intra-class distance while maximizing inter-class distance in the feature space, potentially creating more separable features for challenging character pairs.

To maintain academic integrity, it is important to acknowledge several limitations that, in turn, open promising avenues for future research.

First, the scope of the Out-of-Distribution (OOD) dataset, while a critical component for testing real-world robustness, has its specific boundaries. It was intentionally designed to evaluate performance against natural handwriting variations, a key challenge for modern applications. We acknowledge, however, that this does not encompass the distinct challenge of digitizing historical manuscripts, which often involves significant image degradation. Therefore, while our findings are essential for contemporary use cases, a definitive evaluation for archival purposes would require a larger, more diverse OOD dataset including historically degraded samples.

Second, this research focuses exclusively on single-character classification, which we frame as a foundational and necessary first step for any comprehensive recognition system. More complex challenges, such as character segmentation from words and the modeling of linguistic context in sequential text, were beyond the scope of this study but represent the logical next phase of this research path toward a full Optical Character Recognition (OCR) system.

Lastly, all models relied on transfer learning from weights pre-trained on ImageNet. While this standard practice leverages powerful, universal low-level features (e.g., edges, textures), we recognize the potential for domain bias, as these features are not optimized for stroke-based scripts. Future work could mitigate this by exploring alternative pre-training strategies, such as using large-scale datasets of other world scripts or employing self-supervised learning on an unlabeled corpus of Sundanese text to generate more domain-relevant feature representations.

5. CONCLUSION

This research has successfully evaluated and compared five modern deep learning architectures for the Sundanese script image classification task. The results reveal an apparent dichotomy between peak performance and real-world robustness. EfficientNetB0, with its compound scaling method, achieved the highest accuracy of 97.0% on the curated test set, making it the most accurate model under ideal conditions. However, a crucial out-of-distribution (OOD) robustness test revealed that ResNet50 is the most resilient and reliable model, with an accuracy of 92.5%, demonstrating the least performance degradation when faced with a domain shift. This finding underscores the importance of evaluating models on peak accuracy and their generalization ability—a vital consideration for practical cultural preservation applications.

Building upon these findings and identified limitations, future work can proceed in several promising and specific technical directions. A foundational step involves pursuing model-specific optimizations; for instance, tailoring the classification head for each backbone could unlock further performance, as this study's standardized head may have constrained each architecture's full potential. Future research should also explore hybrid architectures that combine the local feature extraction of CNNs with the global context modeling of Transformers to address the challenges of high inter-class similarity. Performance on visually similar characters could be further enhanced by implementing specialized techniques such as Fine-Grained Visual Classification (FGVC) and metric learning-based losses like Triplet Loss. The research scope must also be expanded from single-character classification to end-to-end sequential recognition of words and sentences. This necessitates a deeper exploration of full Optical Character Recognition (OCR) systems. Finally, investigating optimization techniques like quantization and pruning on the most robust models is essential for practical deployment to create lightweight versions that balance accuracy, robustness, and inference speed for mobile and edge devices.

CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

AUTHORS CONTRIBUTIONS

Conceptualization, Yoga Agustiansyah; methodology, Yoga Agustiansyah, Dhika Restu Fauzi; software, Yoga

Agustiansyah; validation, Yoga Agustiansyah, Dhika Restu Fauzi; formal analysis, Yoga Agustiansyah; investigation, Yoga Agustiansyah; resources, Yoga Agustiansyah; data curation, Yoga Agustiansyah; writing—original draft preparation, Yoga Agustiansyah, Dhika Restu Fauzi; writing—reviewing and editing, Yoga Agustiansyah, Dhika Restu Fauzi; visualization, Yoga Agustiansyah.

REFERENCES

[1] M. I. Ilham and Y. Asriningtias, "Aplikasi Mobile Augmented Reality untuk Mendukung Pengenalan Aksara Sunda," *Edumatic: Jurnal Pendidikan Informatika*, vol. 7, no. 2, pp. 426–434, Dec. 2023, doi: 10.29408/edumatic.v7i2.23602.

[2] E. Yuliyanti, "Teknologi, Budaya, dan Digitalisasi: Menghidupkan Aksara Nusantara dalam Ruang Digital - Dinas Komunikasi, Informatika dan Statistik Kota Cirebon," dkis.cirebonkota.go.id. Accessed: Jun. 28, 2025. [Online]. Available: https://dkis.cirebonkota.go.id/teknologi-budaya-dan-digitalisasi-menghidupkan-aksara-nusantara-dalam-ruang-digital/

[3] Chrismonica, "Belajar Aksara Sunda: Sejarah, Jenis dan Contohnya! | Orami," /www.orami.co.id. Accessed: Jun. 28, 2025. [Online]. Available: https://www.orami.co.id/magazine/aksara-sunda

[4] J. Li *et al.*, "A comprehensive survey of oracle character recognition: challenges, benchmarks, and beyond," Nov. 2024, [Online]. Available: http://arxiv.org/abs/2411.11354

[5] E. M. Puckett, "Optical character recognition helps unlock history | Virginia Tech News | Virginia Tech," news.vt.edu. Accessed: Jun. 28, 2025. [Online]. Available: https://news.vt.edu/articles/2024/03/univlib-ocr.html

[6] S. N. Rahmawati, E. W. Hidayat, and H. Mubarok, "Implementasi Deep Learning pada Pengenalan Aksara Sunda Menggunakan Metode Convolutional Neural Network," *INSERT : Information System and Emerging Technology Journal*, vol. 2, no. 1, pp. 46–58, Jun. 2021, doi: 10.23887/insert.v2i1.37405.

[7] M. F. Naufal, J. Siswantoro, and J. T. Soebroto, "Transliterating Javanese Script Images to Roman Script using Convolutional Neural Network with Transfer Learning," *JOIV : International Journal on Informatics Visualization*, vol. 8, no. 3, p. 1460, Sep. 2024, doi: 10.62527/joiv.8.3.2566.

[8] A. A. Pratama, M. D. Sulistiyo, and A. F. Ihsan, "Balinese Script Handwriting Recognition Using Faster R-CNN," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 6, pp. 1268–1275, Nov. 2023, doi: 10.29207/resti.v7i6.5176.

[9] Y. Agustiansyah and D. Kurniadi, "Indonesian Sign Language Alphabet Image Classification using Vision Transformer," *Journal of Intelligent Systems Technology and Informatics*, vol. 1, no. 1, pp. 1–9, Jun. 2025, Accessed: Jun. 29, 2025. [Online]. Available: https://journal.aptika.org/index.php/jistics/article/view/5

[10] C. Boufenar, M. A. Rabiai, B. N. Zahaf, and K. R. Ouaras, "Bridging the Gap: Fusing CNNs and Transformers to Decode the Elegance of Handwritten Arabic Script," Mar. 2025.

[11] H. Alaeddine and M. Jihene, "Deep Residual Network in Network," *Comput Intell Neurosci*, vol. 2021, no. 1, Jan. 2021, doi: 10.1155/2021/6659083.

[12] A. R. Hermanto, A. Aziz, and S. Sudianto, "Perbandingan Arsitektur MobileNetV2 dan RestNet50 untuk Klasifikasi Jenis Buah Kurma," *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, vol. 12, no. 4, pp. 630–637, Nov. 2024, doi: 10.26418/JUSTIN.V12I4.80358.

[13] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," Sep. 2020, Accessed: Jul. 17, 2025. [Online]. Available: https://arxiv.org/abs/1905.11946

[14] A. Dosovitskiy *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *ICLR 2021 - 9th International Conference on Learning Representations*, Oct. 2020, Accessed: May 21, 2025. [Online]. Available: https://arxiv.org/pdf/2010.11929

[15] R. Grainger, T. Paniagua, X. Song, N. Cuntoor, M. W. Lee, and T. Wu, "PaCa-ViT: Learning Patch-to-Cluster Attention in Vision Transformers," Apr. 2023.

[16] K. Islam, "Recent Advances in Vision Transformer: A Survey and Outlook of Recent Work," Oct. 2023.

[17] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," Jan. 2021, Accessed: Jul. 17, 2025. [Online]. Available: https://arxiv.org/abs/2012.12877

[18] D. R. Fauzi and G. A. H. D, "Comparison of CNN Models Using EfficientNetB0, MobileNetV2, and ResNet50 for Traffic Density with Transfer Learning," *Journal of Intelligent Systems Technology and Informatics*, vol. 1, no. 1, pp. 22–30, Jun. 2025, Accessed: Jun. 29, 2025. [Online]. Available: https://journal.aptika.org/index.php/jistics/article/view/6

[19] I. D. Id, *MACHINE LEARNING: Teori, Studi Kasus dan Implementasi Menggunakan Python*, 1st ed. UR PRESS, 2021. doi: 10.5281/zenodo.5113507.

[20] A. D. Ramdani, "Aksara Sunda," www.kaggle.com. Accessed: Jun. 29, 2025. [Online]. Available: https://www.kaggle.com/datasets/abdidwiramdani/aksara-sunda

[21] Aksara Sunda Dataset, "Aksara Sunda Computer Vision Project," universe.roboflow.com. Accessed: Jun. 29, 2025. [Online]. Available: https://universe.roboflow.com/aksarasunda/aksara-sunda-eayhq

[22] L. Abdiansah, S. Sumarno, A. Eviyanti, and N. L. Azizah, "Penerapan Algoritma Convolutional Neural Networks untuk Pengenalan Tulisan Tangan Aksara Jawa," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 5, no. 2, pp. 496–504, Mar. 2025, doi: 10.57152/malcom.v5i2.1814.

[23] F. Bougourzi, F. Dornaika, and C. Zhang, "Extremely Fine-Grained Visual Classification over Resembling Glyphs in the Wild," Aug. 2024, Accessed: Jul. 17, 2025. [Online]. Available: https://arxiv.org/abs/2408.13774

[24] D. Pant, D. Talukder, D. Kumar, R. Pandey, A. Seth, and C. Arora, "Use of Metric Learning for the Recognition of Handwritten Digits, and its Application to Increase the Outreach of Voice-based Communication Platforms," Apr. 2025, Accessed: Jul. 17, 2025. [Online]. Available: https://arxiv.org/abs/2504.18948