Evaluation of Synthetic Data Effectiveness Using Generative Adversarial Networks (GAN) in Improving Javanese Script Recognition on Ancient Manuscripts

Muhammad 'Arif Faizin [1], Nanik Suciati [2,*], and Chastine Fatichah [3]

[1, 2] Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

[3] Study Program in Informatics Engineering. Universitas Dian Nuswantoro, Semarang, Indonesia

E-mail: 6025231092@student.its.ac.id[1], nanik@its.ac.id[2], and chastine@if.its.ac.id[3]

ABSTRACT

The imbalance of Javanese script data in ancient manuscript recognition poses a challenge due to the limited availability of data. A potential approach to addressing this issue is the use of Generative Adversarial Networks (GAN). This study evaluates the effectiveness of synthetic data generated using Enhanced Balancing GAN (EBGAN) in mitigating data imbalance. Various evaluation scenarios are conducted, including: (i) assessing the impact of synthetic data as augmentation, (ii) evaluating the sufficiency of synthetic data for recognition models, (iii) analyzing minority class oversampling with different selection strategies, and (iv) evaluating model generalization through cross-validation. Quantitative analysis of the generated synthetic data, based on Fréchet Inception Distance (FID) and Structural Similarity Index (SSIM), as well as visual assessment, indicates that the quality of synthetic data closely resembles real data. Additionally, experimental results demonstrate that combining real and synthetic data improves accuracy, precision, recall, and F1-score. The oversampling strategy for synthetic data has proven effective in meeting the data sufficiency requirements for training recognition models. Meanwhile, selecting minority classes and determining threshold values based on percentage, distribution, and model performance in oversampling can serve as guidelines for enhancing script recognition performance. Compared to previous methods, the use of EBGAN has been shown to produce more diverse synthetic data with better visual quality. However, further research is still needed to optimize GAN performance in supporting script recognition.

Keywords: Character recognition, data imbalance, GAN, javanese script, synthetic data

1. Introduction

Javanese script has an important historical value as one of the cultural heritages of the Nusantara that reflects the identity and rich traditions of the Indonesian nation [1]. It was widely used in ancient manuscripts to record a variety of information, from religious texts to royal history [2]. However, since the 19th century Javanese script has experienced a decline in use due to the use of more practical Latin script, including the influence of the cost of using it in more expensive production media [2].

In addition, the introduction of Javanese script through a technological approach faces great challenges [3]. Javanese script is in danger of experiencing data imbalance due to limited data access like other local scripts in Indonesia (such as Sundanese and Balinese scripts) [4]. Javanese script has structural uniqueness and complexity that often makes data collection difficult, especially in ancient manuscripts that have a variety of writing forms [5]. This problem becomes more complicated by the presence of underrepresented characters, so the model tends to be biased towards the majority class.

To overcome this problem, one of the approaches is the use of synthetic data, which is considered to be more robust than algorithmic approaches [6]. There are several traditional methods that are often used to handle

* Corresponding author.

Received: January 30th, 2025. Revised: February 4th, 2025. Accepted: February 22nd, 2025.

Available online: February 25th, 2025.

DOI: 10.12962/j24068535.v23i1.a1256

data imbalance, such as image transformation (rotation, translation, tilt, etc.) or image interpolation approach with Synthetic Minority Oversampling Technique (SMOTE) [7]. A more recent approach using Generative Adversarial Networks (GAN) has proven to be effective in generating synthetic data that resembles the original data both in terms of data distribution and recognition accuracy [8]. However, there is a need for in-depth research and evaluation of strategies for using synthetic data that improving script recognition on imbalanced data effectively.

This study aims to evaluate the effectiveness of using synthetic data using GAN in improving Javanese script recognition on ancient manuscripts. The GAN approach used in this research is using Enhanced Balancing GAN (EBGAN) [9], which is designed to make synthetic data using imbalanced data, allowing it to overcome the problem of data imbalance effectively. In addition, some contributions from this research are as follows:

- Evaluating the effectiveness of synthetic data in Javanese script recognition, either as a substitute or complement to the original data.
- Proposing a synthetic data generation approach using GAN which adaptively handling imbalanced data.
- Identify empirical guidelines in the use of synthetic data for Javanese script recognition.

By conducting the approaches and methods mentioned, this research aims to contribute to the development of Javanese script recognition methods in ancient manuscripts.

2. Literature Review

Handling data imbalance in the early stages using traditional methods is done by using sampling methods or data augmentation (such as rotation, translation, scale, etc.). Sampling methods can be divided into undersampling, oversampling, and hybrid methods that combine both. In the oversampling method, other than using random oversampling, there are more systematic methods using interpolation between data such as Synthetic Minority Oversampling Technique (SMOTE) [10], [11], Borderline-SMOTE (B-SMOTE) [12], [13], and Adaptive Synthetic Sampling (ADASYN) [14], [15]. The weakness of traditional methods is that they are not capable of generating data outside the existing distribution, and the possibility of creating the same synthetic data repeatedly is high [16].

Augmentation methods using Generative Adversarial Networks (GAN) [17], [18], are considered to overcome the shortcomings of traditional methods. GANs consist of two main components, namely generators and discriminators, which are trained simultaneously in an adversarial framework. Various types of GANs have been generated in previous studies to generate class-based synthetic data, such as Conditional GAN [19], [20], Auxiliary Classifier GAN (ACGAN) [21], Conditional Wasserstein GAN (CWGAN) [22], or Balancing GAN (BAGAN) [23]. The BAGAN approach focuses on the generation of synthetic data using unbalanced data, making it suitable for use on real data. The development of BAGAN, namely Enhanced Balancing GAN (EBGAN) [9] optimizes the previous method with a better approach and has been proven to improve the quality of the synthetic data generated.

At the time of this research, no handling of data imbalance on Javanese script data was found, but some research on other similar scripts has been conducted. The study [4] using traditional methods of translation and rotation on Sundanese script, was proved to increase the accuracy of character detection from 72.29% to 93.92%, but the study did not explain the distribution of the resulting data. The study [24] proposed handling data imbalance in Japanese script data using a two-stage generation system Cascade Variational Auto Encoder (VAE). The results showed that the use of synthetic data was able to increase accuracy from 94.02% to 95.56%.

Handling data imbalance using GAN has been done in previous studies. The study [25] proposed the use of Multiple Fake Class GAN (MFC-GAN) to handle data imbalance by creating different minority class samples for each class. MFC-GAN was shown to accelerate model convergence, and improve classification accuracy on MNIST, E-MNIST, SVHN, and CIFAR-10 data. The study [26] used a two-stage combination with ScrabbleGAN and Bidirectional LSTM to generate synthetic data on handwriting in Arabic script in one word assisted with Connectionist Temporal Classification (CTC). The method proved to overcome the problem of data imbalance in the INF/ENIT and AHDB datasets. Meanwhile, research [27] used Transfer Historical GAN (TH-GAN) by optimizing the use of U-Net in the generator and WGAN in the discriminator. TH-GAN is able to perform style transfer from ancient Chinese manuscript characters into printed characters from Hei Ti, Song Ti, and Kai Ti fonts to produce synthetic data that can improve recognition accuracy in Chinese characters. Research [28] proposed the Semi-MixFontGAN method using Multi-Task Font Encoder to extract features from data and Font Transfer Network and combine it with labelled and unlabeled data through MixFont to generate synthetic data. Experimental results show that Semi-MixFontGAN can improve the classification accuracy from 93.52% to 96.69% on Kuzushiji dataset.

According to those studies, the use of synthetic data generated from GAN can help improve the accuracy of character recognition. Thus, this study aims to evaluate the handling of data imbalance in Javanese characters of ancient manuscripts using GAN.

3. Methodology

In this research, there are several stages in the evaluation. First, splitting the original Javanese script data into training data and test data. The training data is then further divided into training data and validation data which are used to train the GAN model. Then, the Enhanced Balancing GAN (EBGAN) model was trained using the training data to produce synthetic data as augmentation data. Furthermore, the synthetic data generated by the GAN generator is used in various scenarios, either as a single training data, in combination with the original data, or to balance the class data distribution. This balanced dataset is used to train the classification model, which is evaluated using test data based on accuracy, precision, recall, and F1-score metrics to assess the effectiveness of using synthetic data. Fig. 1 shows the overall process of this research.

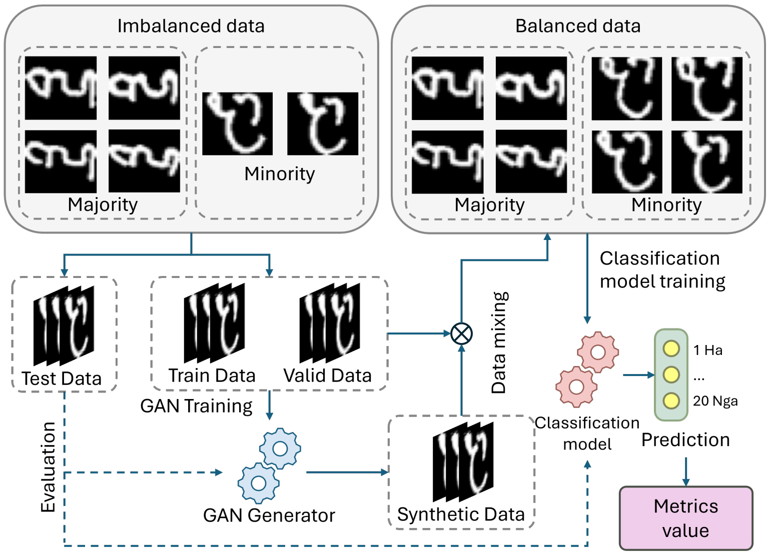

Fig. 1: Research Methodology

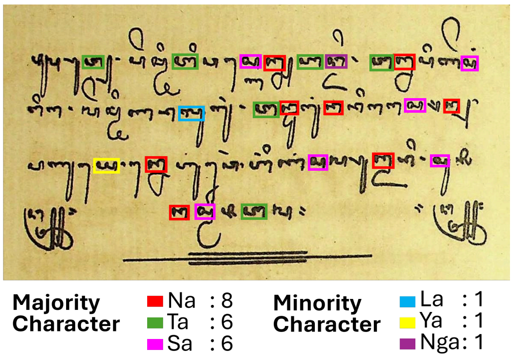

Fig. 2: Illustration of images in the HJCS_DETC dataset and distribution of majority and minority classes

3.1. Dataset Description

This research uses the HJCS_DETC dataset [29], a dataset originally used for object detection. The dataset consists of 60 images that have been collected and annotated from manuscripts, consisting of 74 character classes of the Javanese script. This research focuses on 20 character classes taken from wyanjana script to represent the basic characters in Javanese script. Examples of images in the HJCS_DETC dataset and illustrations of the distribution of majority and minority character positions can be seen in Fig. 2.

Each character is extracted and processed as a single image to produce uniform data. Several stages of processing were performed: conversion to grayscale, application of gaussian blur to reduce noise, application of Otsu thresholding to increase character contrast, and conversion of the image to a size of 32x32 pixels while adding padding to position the character in the center of the image. This processing ensures that the data is standardized and uniform, making it easier for the model to create synthetic data and perform classification. The total characters contained in this dataset are 10,975 characters and are imbalanced, as can be seen in Fig. 3. Fig. 4 shows the feature representation of HJCS_DETC using pre-trained ResNet50 and embedding as a feature extractor, it can be seen that

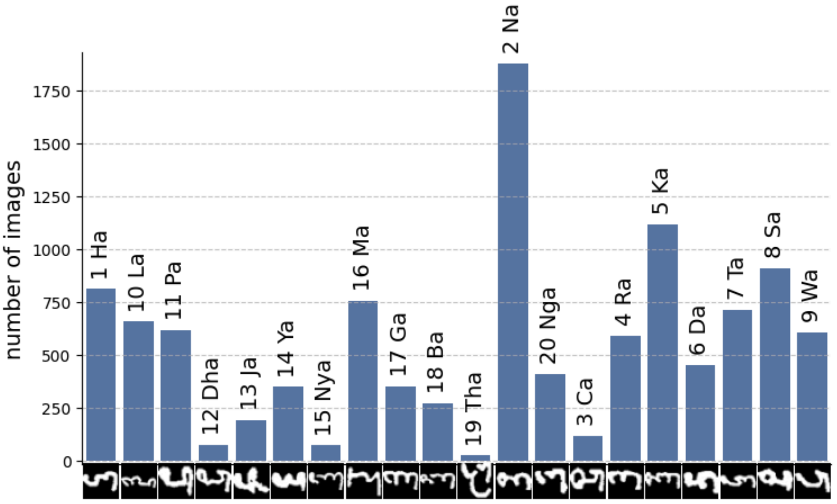

Fig. 3: Character class distribution on the HJCS_DETC dataset

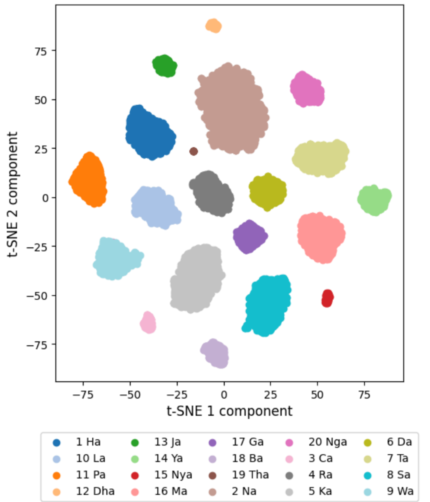

Fig. 4: Visualization of HJCS_DETC dataset features using pre-trained ResNet50 and Embedding

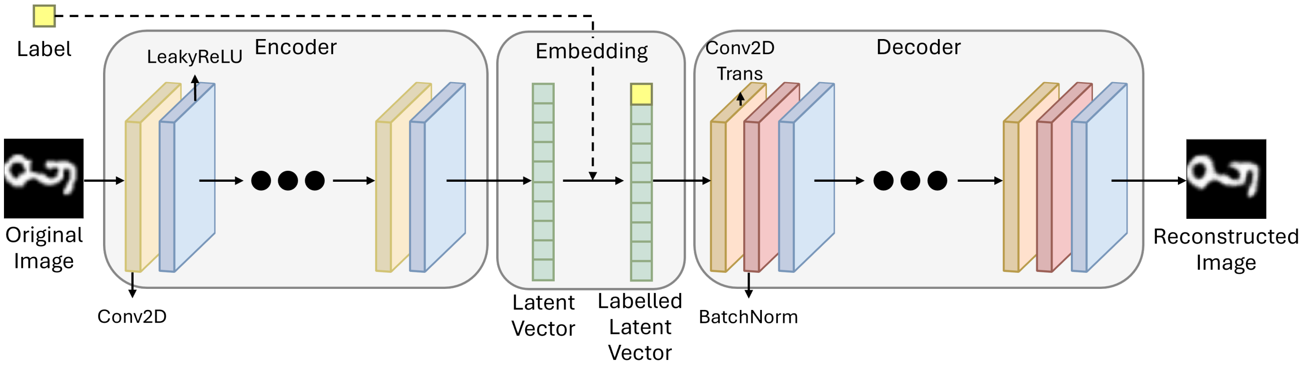

Fig. 5: Autoencoder initialization process on EBGAN

the feature representation between classes can be clearly distinguished and there is a difference in the amount of each feature, according to the number of images in the dataset.

3.2. Data Synthetic using Enhanced Balancing GAN (EBGAN)

In this study, EBGAN [9] is used to generate synthetic data with a focus on increasing the number of samples for minority characters. EBGAN was chosen for this task primarily because it addresses the critical challenge of data imbalance, which is especially relevant in Javanese script recognition, where certain characters appear less frequently. EBGAN consists of two main stages, namely autoencoder initialization and GAN training. The basic idea is derived from Balancing GAN (BAGAN) [23] by learning the distribution of the original data using encoders and decoders, resulting in reconstructed image data. EBGAN then adds class labels through an embedding process at initialization to clarify the differences between the resulting character classes. In this process, EBGAN minimizes the L1 loss value between the original data and the generated data to train the autoencoder. Unlike BAGAN, EBGAN can learn information from the classes, thus helping to generate synthetic data in a controlled manner. This method not only preserves the structural integrity of the characters, as autoencoders do, but also introduces the necessary variations to enhance generalization, making it more robust for training models on imbalanced datasets. An illustration of the encoder initialization process can be seen in Fig. 5.

The pre-trained decoder is then used in the GAN training process as a generator. In this process, the generator is trained using normal noise and labels from the original data, and produces a reconstructed image as fake data. Unlike BAGAN, EBGAN uses the discriminator architecture of cWGAN-GP [22] to eliminate the dependency

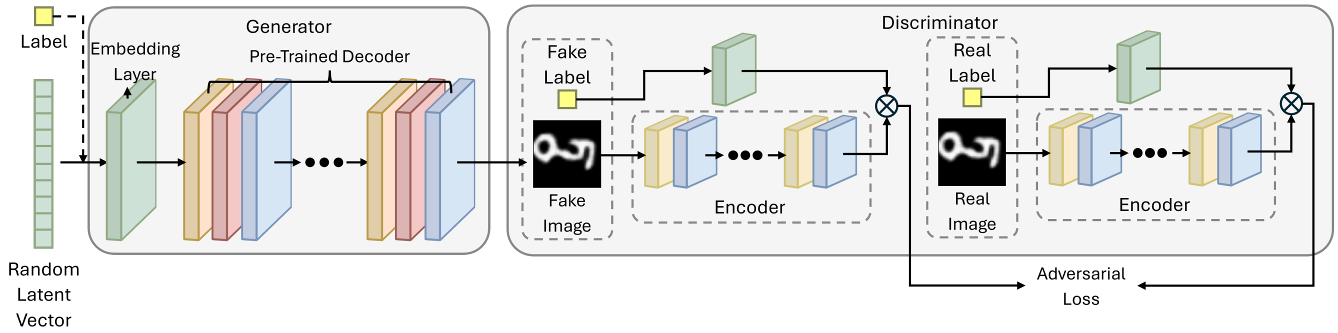

Fig. 6: GAN training process on EBGAN

of using the encoder from the initialization process. In addition, EBGAN also uses gradient-penalty as in [30] to limit the gradient during the training process. Moreover, EBGAN introduces a training ratio of more than one to improve convergence between the generator and discriminator. An illustration of the GAN training process can be seen in Fig. 6.

The GAN training process applies an adversarial process, i.e., the process will be considered optimal if it is able to maximize the loss of the discriminator and minimize the loss of the generator. If the generator , discriminator , real image sample from the real image probability distribution , fake image sample from the fake image probability distribution , where and is a random noise vector sample obtained from the normal distribution , then the GAN objective function can be defined in (1). The discriminator maximizes the loss function using (2) and the generator minimizes the loss function using (3).

$$\underset{G}{min}\,\underset{V}{max}\,V(G,D)=E_{x_r\sim X_{real}}[log(D(x_r))]+E_{x_g\sim X_{generated}}\left[log\left(1-D\left(x_g\right)\right)\right] \tag{1}$$

$$\underset{D}{max}\,L_D\left(X_{real},X_{generated}\right)=-E_{x_r\sim X_{real}}[log(D(x_r))]-E_{x_g\sim X_{generated}}\left[log\left(1-D\left(x_g\right)\right)\right] \tag{2}$$

$$\underset{G}{min}\,L_G\left(X_{generated}\right)=-E_{x_g\sim X_{generated}}\left[log\left(D\left(x_g\right)\right)\right] \tag{3}$$

Then to accelerate convergence and improve stability in the training process, EBGAN adapted Wasserstein GAN-gradient penalty (WGAN-GP) [30], [31] by applying Wasserstein distance and gradient-penalty as the main concepts. Wasserstein distance is defined as the minimum amount to move to form . However, it is not computationally effective, so using Kanotrovich-Rubinstein duality, for as a 1-Lipschitz function, the Wasserstein distance function can be defined as (4).

$$W\left(X_{real},X_{generated}\right)=\underset{L_D\sim Lip_1}{sup}\left(E_{x_r\sim X_{real}}[D(x_r)]-E_{x_g\sim X_{generated}}\left[D\left(x_g\right)\right]\right) \tag{4}$$

Then the 1-Lipschitz function will force the discriminator function , to avoid extreme predictions. In WGAN, the discriminator is referred as “critic” C which will maximize the difference of expected values of and . Unlike a typical GAN that will produce a probability value, Wasserstein GAN will produce a more meaningful value to train the GAN. However, it becomes a problem if the value is infinite, hence the use of gradient-penalty will limit the value by applying regularization to the gradient in training using interpolation between the two images and . If is the interpolation, and , then the gradient-penalty function can be defined in (5).

$$GP=E_{\hat{x}\sim\hat{X}}\left[\left(\left\|\nabla_{\hat{}}(x)C(\hat{x})\right\|_2-1\right)^2\right] \tag{5}$$

Thus, if is the weight of the gradient-penalty function then the objective function of WGAN-GP is defined in (6).

$$\begin{aligned} &\underset{C}{max}\, L_C\left(X_{real}, X_{generated}\right) \\ &\quad = E_{x_r \sim X_{real}}[C(x_r)] - E_{x_g \sim X_{generated}} | \left[C\left(x_g\right)\right] + \lambda E_{\hat{x} \sim \hat{X}}\left[\left(\|\nabla_{\hat{}}(x) C(\hat{x})\|_2 - 1\right)^2\right] \end{aligned} \tag{6}$$

Then EBGAN adapted draGAN [32] by using log-sigmoid loss function and adapted cWGAN-GP [22] by adding the label of the real image to the generator and discriminator to control the class in the generator. EBGAN adopts the model training system in BAGAN to overcome data imbalance and applies class-based training convergence optimization by using a random label for fake images and a random label for real images. So, the final loss value of the discriminator is defined in (7) and the loss value of the generator is defined in (8).

$$\begin{aligned} \underset{C}{max}\, L_C\left(X_{real}, Z, Y_{real}, Y_{fake}, Y_{wrong}\right) = &-E_{x_r, y_r} \sim (X_{real}, Y_{real})[log(C(x_r, y_r))] \\ &-E_{z, y_f} \sim \left(Z, Y_{fake}\right)\left[log\left(1 - C\left(G\left(z, y_f\right), y_f\right)\right)\right] \\ &-E_{x_r, y_w} \sim \left(X_{real}, Y_{wrong}\right)\left[log\left(1 - C\left(x_r, y_{wrong}\right)\right)\right] \\ &+\lambda E_{(\hat{x}, y_r) \sim \left(\hat{X}, Y_{real}\right)}\left[\left(\left\|\nabla_{(\hat{x}, y_r)} C(\hat{x}, y_r)\right\|_2 - 1\right)^2\right] \end{aligned} \tag{7}$$

$$\underset{G}{min}\, L_G\left(Z, Y_{fake}\right) = -E_{(z, y_f) \sim (Z, Y_{fake})}\left[log\left(C\left(G\left(z, y_f\right)\right)\right)\right] \tag{8}$$

The decision to use EBGAN is also supported by previous studies demonstrating its effectiveness in generating high-quality synthetic data for imbalanced datasets. For instance, in the original EBGAN paper, the method showed superior performance in domains requiring balanced data augmentation, outperforming traditional GAN models in both visual quality and distribution accuracy. EBGAN has been successfully tested on benchmark datasets such as MNIST, CIFAR-10, and cell imaging datasets, where it consistently achieved lower FID scores, and better class distribution balance compared to models like WGAN and BAGAN. In the MNIST dataset, EBGAN was able to generate clearer and more distinct digit representations, while in CIFAR-10, it managed to maintain inter-class diversity without sacrificing image quality. For biomedical applications, such as cell image datasets, EBGAN demonstrated its ability to generate high-fidelity synthetic images, aiding in data augmentation for rare cell types. By leveraging these proven strengths, this paper applies EBGAN to the specific challenges of Javanese manuscript recognition, aiming to replicate similar improvements in performance.

3.3. Recognition Model Architecture

The model used for Javanese script recognition is a Simple Convolutional Neural Network (CNN) with a customized architecture for script image recognition. The architecture consists of three convolutional layers with the number of filters 32, 64, and 128 respectively, each using a kernel size of 3×3 and a ReLU activation function. In addition, a dropout at 0.5 was used to reduce overfitting. The model was computed using Adam's optimizer with learning rate 1e-4 and sparse categorical cross entropy loss function. Training was performed for 10 epochs with a batch size of 32, and validation using test data to evaluate the performance of the model. An illustration of the Simple CNN architecture used can be seen in Fig. 7.

3.4. Evaluation Scenario

This study aims to evaluate the effectiveness of using synthetic data generated by GAN in Javanese script recognition through three main scenarios designed to cover different aspects of evaluation. Each scenario is designed to answer specific questions regarding the impact of synthetic data on model performance, both globally and on minority classes.

A. Effect of Synthetic Data on Original Data

This scenario aims to evaluate the impact of synthetic data on classification model performance when used as a supplement to the original data. Synthetic data was generated using the GAN and added to the original dataset with the aim of balancing the number of samples between classes. In this evaluation, each class in the dataset

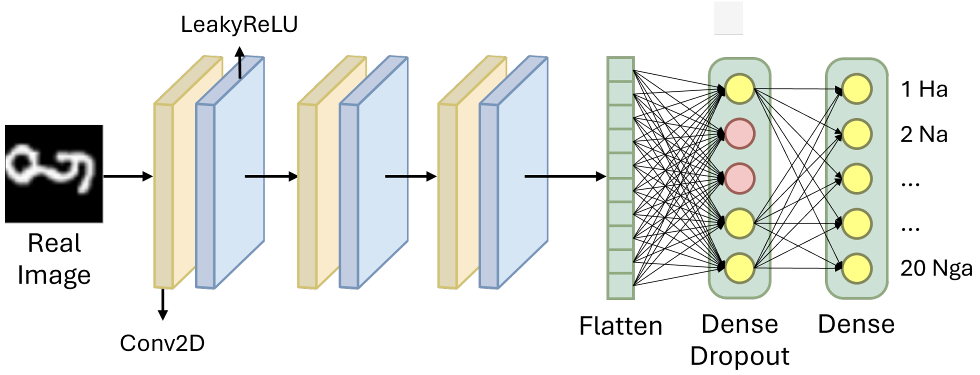

Fig. 7: Simple CNN Architecture for character recognitions

was expanded to the maximum number of classes with the largest number of samples. The evaluation is done by comparing the performance of the models trained using the original data, synthetic data, and also their combinations.

B. Synthetic Data Sufficiency for the Recognition Model

This scenario is designed to determine the optimal amount of synthetic data that can be used without overfitting the model. The synthetic data was added in varying amounts, varying from the same amount as the original data up to several times more. The analysis was conducted to identify the optimal limit at which the synthetic data still provided performance improvement before it started to have a diminishing returns effect. This is important to ensure that augmentation using synthetic data is sufficient and still effective in improving the generalizability of the model.

C. Effect of Minority Class Selection

In this scenario, the study evaluates an oversampling strategy that targets only minority classes compared to oversampling all classes. The minority class is determined based on several approaches, such as the number of samples in the dataset, statistical distribution (median, quartile), and initial model performance (low recall or $F1\text{-}score$). These strategies are compared to assess whether a more selective augmentation can provide more optimal results than a whole-of-class oversampling approach.

D. Generalizability of the Recognition Model

To evaluate the stability and generalizability of the model, cross validation was performed using the K-Fold Cross Validation method with =5 and =10. There is no fixed number of k values, but 5 and 10 were chosen because they represent a division of data that is suitable for both sufficient and limited data [33]. This approach aims to assess how well the model is able to adapt over variations in the training data. The model is tested on various subsets of data ensuring the use of synthetic data does not cause unwanted bias and consistently improves the model's performance. The evaluation also considers the standard deviation of the test results to measure the variability and stability of the model's predictions in various scenarios.

The evaluation in each scenario was conducted using two main metrics: data generation quality metrics and classification performance metrics. Generation quality metrics are obtained by the Fréchet Inception Distance (FID) and Structural Similarity Index Measure ($SSIM$) values. Meanwhile, classification metrics are obtained by the $accuracy$, $precision$, $recall$, and $F1\text{-}score$ values both globally and specifically for each class.

FID value is a metric to measure the distance between two data distributions, performed by comparing the mean and covariance of the feature vectors generated from the Inception V3 pre-trained network between the original data and the fake data. If and are the mean and covariance of the original and fake data features, then the FID value can be defined in (9). The lower FID value indicates that the fake data is closer to the distribution of the test data.

$$FID(r,g) = \left\|\mu_r - \mu_g\right\|_2^2 + Tr\left(\Sigma_r + \Sigma_g - 2\left(\Sigma_r\Sigma_g\right)^{\frac{1}{2}}\right) \tag{9}$$

$SSIM$ value is a metric to measure the structural similarity between two data using the luminance, contrast, and structure of the data. If is the average intensity, is the standard deviation, and is the covariance between the original and fake data for the original image and fake image, then the $SSIM$ value can be defined in (10). The larger $SSIM$ value and closer to a score of 1, indicates a better structural similarity between the original data and the fake data.

$$SSIM(r,g) = \left(2\mu_r\mu_g + C_1\right)\frac{2\sigma_{rg} + C_2}{\left(\mu_r^2 + \mu_g^2 + C_1\right)\left(\sigma_r^2 + \sigma_g^2 + C_2\right)} \tag{10}$$

In evaluating the classification performance, several metrics can be used to indicate the performance of the classification model. The $accuracy$, $precision$, $recall$, and $F1\text{-}score$ can be calculated using the confusion matrix. If is the number of correct positive predictions, is the number of correct negative predictions, is the number of incorrect positive predictions, and is the number of incorrect negative predictions, then the $accuracy$, $precision$, $recall$, and $F1\text{-}score$ values can be calculated with Equations (12)-(15).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

$$Precision = \frac{TP}{TP + FP} \tag{12}$$

$$Recall = \frac{TP}{TP + FN} \tag{13}$$

$$F1\text{-}score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{14}$$

4. Experiments and Results

4.1. Characteristic Evaluation of Synthetic Data

To evaluate the effectiveness of the synthetic data generated by EBGAN, an analysis was conducted on the characteristics of the synthetic data. Using EBGAN's generator trained on the original data, a total of 2000 synthetic data samples were generated. See Fig. 8 for sample of results. EBGAN is successful in replicating original data visually, particularly in the majority class with consistent patterns and detail. However, in the minority class, some weaknesses are observed, such as broken, too thick, or wrongly connected characters. Although, the t-SNE visualization in Fig. 9 shows that synthetic data still has clear inter-class clusters, reflecting a good representation of the original data distribution at the global level. This shows that despite the shortcomings in visual quality of some classes, EBGAN is still effective in representing inter-class information for data augmentation.

Table 1 shows the FID score between the original data, generated image using EBGAN, and reconstructed image of Autoencoder, compared with the test data. The FID value of the original data is used as the reference minimum value and the Autoencoder reconstructed image is used as the reference maximum value. The results show that the EBGAN and Autoencoder methods generate synthetic data that does not fully resemble the original distribution. The autoencoder tends to produce higher FID values than EBGAN in most classes, indicating that although the autoencoder can generate more data variations, the distribution is further away from the original data.

However, EBGAN's performance resulting in lower FID scores compared with autoencoder in minority classes such as Ca (3), Wa (9), Dha (12), Ja (13) and Tha (19). This indicates that EBGAN faces difficulties in representing characters that rarely appear in the original dataset, resulting in less realistic synthetic data for those classes. In contrast, the autoencoder still shows more consistent results on minority classes because of its goals on simplifying reconstruction of the original data without generating variations. This shows that the imbalance in the original dataset affects EBGAN's performance, especially in generating synthetic data for classes with a small amount data.

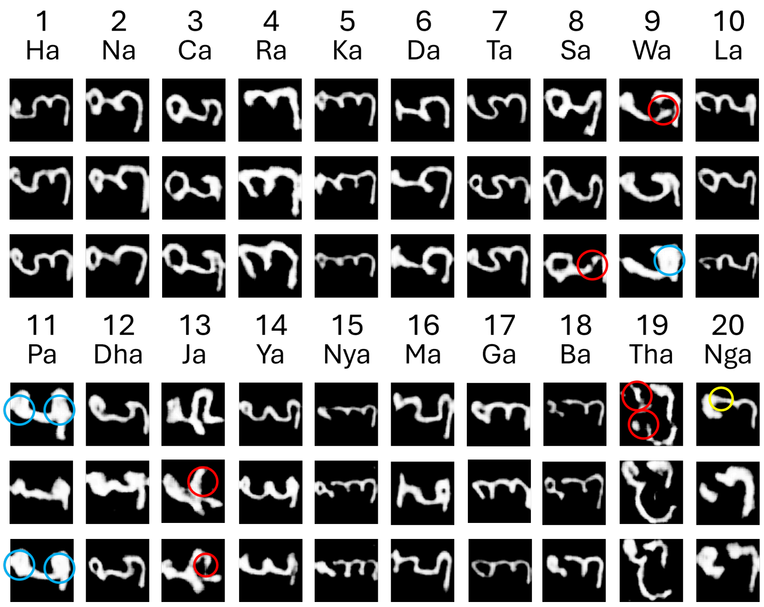

Fig. 8: Samples of synthetic data results from the GAN process in each class, red circles indicate characters that failed to be formed, blue circles indicate characters that are over-thick, yellow circles indicate characters that are connected

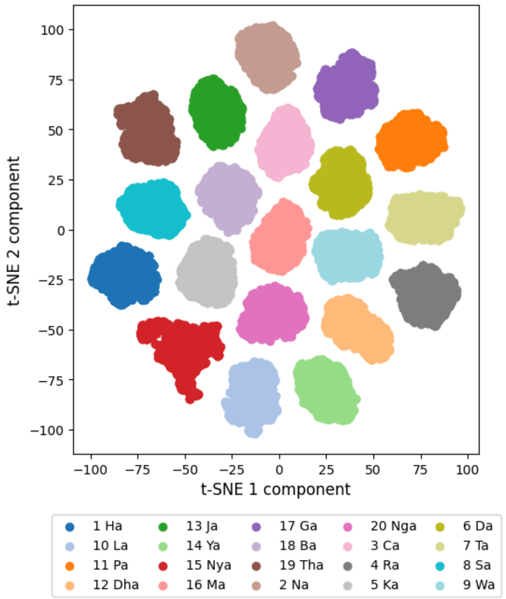

Fig. 9: Visualization of synthetic data features using pre-trained ResNet50 and Embedding

Table 1: Comparison of FID values over 2000 samples of synthetic data using GAN

Class	1 Ha	2 Na	3 Ca	4 Ra	5 Ka	6 Da	7 Ta	8 Sa	9 Wa	10 La
Original*	88.72	113.67	134.50	123.86	88.74	121.17	103.34	119.28	135.10	87.62
EBGAN	**133.79**	**160.47**	195.46	**188.26**	**153.51**	**179.10**	**158.96**	**177.55**	218.65	**152.50**
Autoencoder	157.03	199.58	**188.24**	191.24	173.42	202.92	182.73	178.66	**211.09**	173.28
Class	11 Pa	12 Dha	13 Ja	14 Ya	15 Nya	16 Ma	17 Ga	18 Ba	19 Tha	20 Nga
Original*	130.16	139.82	163.81	85.44	101.31	138.17	94.95	128.34	183.01	144.12
EBGAN	**204.08**	240.05	263.54	**145.70**	**193.93**	**192.84**	**152.29**	**184.71**	298.04	**196.58**
Autoencoder	218.95	**205.32**	**222.47**	174.74	221.81	228.89	179.71	212.84	**233.22**	217.77

Table 2: Comparison of SSIM values over 2000 samples of synthetic data using GAN

Class	1 Ha	2 Na	3 Ca	4 Ra	5 Ka	6 Da	7 Ta	8 Sa	9 Wa	10 La
Original*	0.605	0.536	0.548	0.518	0.642	0.484	0.564	0.492	0.510	0.579
EBGAN	0.575	0.522	0.514	0.504	0.602	0.470	0.535	0.473	0.490	0.560
Autoencoder	**0.613**	**0.542**	**0.562**	**0.521**	**0.649**	**0.489**	**0.571**	**0.498**	**0.513**	**0.586**
Class	11 Pa	12 Dha	13 Ja	14 Ya	15 Nya	16 Ma	17 Ga	18 Ba	19 Tha	20 Nga
Original*	0.474	0.573	0.418	0.613	0.688	0.531	0.626	0.595	0.410	0.513
EBGAN	0.466	0.484	0.385	0.578	0.581	0.499	0.586	0.578	0.320	0.478
Autoencoder	**0.476**	**0.579**	**0.419**	**0.619**	**0.687**	**0.535**	**0.630**	**0.604**	**0.411**	**0.518**

Evaluation using the Structural Similarity Index (SSIM) in Table 2 shows that the autoencoder produces higher scores than EBGAN and even the original data, reflecting its ability to reconstruct images with visual structures that are very similar to the original data. This is understandable as the autoencoder is designed to preserve the original details without adding variations. Meanwhile, EBGAN resulted in lower SSIM scores due to its process aimed at creating synthetic data with additional variations. This drop in scores is more pronounced for minority classes, suggesting that EBGAN faces challenges in maintaining visual structure for underrepresented classes, while still providing the variation essential for data augmentation.

Table 3: Classification model performance in Scenario 1

Code	Scenario	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
S1	Original data (100%)	95.96	96.15	95.96	95.93
S2	Synthetic data (2000 per class)	93.89	94.53	93.89	93.95
S3	Original data (100%) + Synthetic data (100%)	**96.79**	**96.94**	**96.79**	**96.77**

While EBGAN effectively generates synthetic data that resembles the original distribution, it has both strengths and weaknesses in data augmentation. Its main strength is maintaining inter-class distribution while adding variations, which improves model robustness. EBGAN achieves lower FID scores than autoencoders, especially in majority classes, indicating it captures fine details better. However, it struggles with minority classes, producing distorted characters like broken strokes or misconnected structures. This happens because the model doesn't have enough data to learn these rare features well. EBGAN also shows lower SSIM scores in minority classes, meaning it sacrifices structural accuracy for variation. While EBGAN is a strong tool for data augmentation, additional methods, like class-aware training, may be needed to improve the quality of samples for rare classes.

Based on the evaluation of score characteristics and visual analysis, EBGAN is generally able to produce synthetic data that resembles the original data, both in terms of structure and distribution between classes. Although there are weaknesses in minority classes, such as broken or misconnected characters, the t-SNE visualization shows that the synthetic data still represents the original data distribution well. This indicates that EBGAN is effective as a data augmentation method, especially in handling data imbalance.

In summary, the strengths of using EBGAN lie in its ability to balance class distributions effectively and generate synthetic data that maintains global inter-class relationships, as evidenced by the t-SNE visualization and lower FID scores in majority classes. However, its weaknesses become apparent in minority classes, where the model struggles to generate visually accurate characters, leading to issues like broken or misconnected components. This limitation is influenced by the imbalance in the original dataset and EBGAN's focus on introducing variation, which can compromise the visual quality in underrepresented classes. Despite these challenges, EBGAN remains a valuable tool for data augmentation, particularly in applications where balancing class distributions is critical.

4.2. Performance Evaluation of Recognition Model

A. Scenario 1: Effect of Synthetic Data on Original Data

This evaluation aims to assess the effectiveness of the synthetic data in augmenting and balancing the original data. In Scenario 1, synthetic data is used to perform oversampling until the number of each class reaches the maximum number of classes in the original data, which is 1881 characters. From Table 3, it is observed that using synthetic data as augmentation improves the performance of the recognition model compared with using the original data or synthetic data separately. The combination of original and synthetic data yields the highest accuracy, precision, recall, and F1-score, indicating that synthetic data effectively complements the original data by adding variety and helping to overcome data imbalance.

B. Scenario 2: Synthetic Data Sufficiency for the Recognition Model

In Scenario 2, this evaluation aims to determine the amount of data needed to be used in the classification model, and to see the impact of adding more synthetic data beyond the number of original data on the model's performance. Based on Table 4, the additional synthetic data more than the original data (S4-S7) can slowly improve the model's performance, indicating its capability to represent the original data in conducting classification. The use of synthetic data as augmentation in S7 shows that the recognition model does not experience overfitting synthetic data, even though the amount of original data is much smaller than synthetic data.

However, this improvement becomes smaller after the addition of synthetic data beyond a certain amount, which indicates the phenomenon of diminishing returns [34]. The metrics value decreased in S5 but increased again in S6 and S7. This indicates that adding more synthetic data than a certain amount starts to provide more limited

Table 4: Classification model performance in Scenario 2

Code	Scenario	Num. of data	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
S3	Oversample synthetic data 1881 (100% of original data)	37,620	96.79	96.94	96.79	96.77
S4	Oversample synthetic data 2000	40,000	97.13	97.24	97.13	97.10
S5	Oversample synthetic data 3000	60,000	96.86	97.06	96.86	96.86
S6	Oversample synthetic data 4000	80,000	97.27	97.39	97.27	97.25
S7	Oversample synthetic data 5000	100,000	97.51	97.58	97.51	97.49

benefits. Therefore, although increasing the amount of synthetic data can improve model performance, there is a point where the improvement starts to decrease, which needs to be considered in the selection of the optimal amount of synthetic data.

C. Scenario 3: Effect of Minority Class Selection

In Scenario 3, the goal of this evaluation is to evaluate the effectiveness of oversampling only minority classes compared to oversampling all classes. In oversampling minority classes, the selection of minority classes can be done with several approaches: (i) percentage-based approach (25%, 50%, 75% thresholds), (ii) distribution-based approach (mean and quartile statistics thresholds), (iii) base model performance-based approach (recall and F1-score thresholds).

The results in Table 5 show the experimental results of various oversampling strategies on classification performance. The results of using oversampling on all classes close to the maximum number of original data classes in S3 show the highest overall performance. However, more targeted oversampling strategies (S8-S16) show that targeted augmentation can still improve accuracy, precision, recall and F1-score. Oversampling based on a fixed percentage (S8-S10) shows that increasing the augmentation percentage provides a gradual increase in performance, but experiences diminishing returns as the amount of synthetic data approaches the maximum limit. Meanwhile, oversampling based on statistical distribution (S11-S14) and model performance metrics (S15-S16) show that strategically selecting minority classes, rather than uniformly augmenting all classes, can result in equivalent or even better improvements in some cases. Similarly, statistical methods such as median and quartile thresholds (S12-S14) provided better generalization than random selection, with S14 (third quartile-based oversampling) achieving one of the best performances.

The effectiveness of targeted oversampling depends on how minority classes are selected. The recall-based and F1-score-based strategies (S15-S16) show competitive results, indicating that prioritizing hard-to-recognize classes can provide greater benefits. These results suggest that rather than simply augmenting all classes equally, performing selective augmentation based on performance gaps or data distribution can optimize model improvement while minimizing unnecessary data augmentation.

D. Scenario 4: Generalizability of the Recognition Model

In Scenario 4, this evaluation aims to determine the ability of synthetic data to help generalize the classification model. The dataset is evaluated using 5-Fold and 10-Fold Cross Validation. In this method, the dataset is divided into equal parts. The model is trained using k-1 parts and tested on the remaining parts, repeated k times until each part is used as test data. The evaluation result is calculated based on the mean and standard deviation of all folds. This approach ensures that the model is tested on various subsets of data.

The evaluation results in Table 6 using 5-Fold and 10-Fold Cross Validation show that the addition of synthetic data with GAN-based oversampling consistently improves model performance compared to using only original data. In both scenarios, both accuracy, precision, recall, and F1-score improved after oversampling. In addition, the smaller standard deviation in the 5-Fold scenario with synthetic data indicates that the model is more stable and has lower variability in its predictions. However, using 10-Fold increases the standard deviation, although the

Table 5: Classification Model performance for Scenario 3

Basic	Code	Scenario	Minority Class	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
-	S3	Oversample all (1881)	All class except 2	96.79	96.94	96.79	96.77
Percentage	S8	Oversample by value of top 25% (470)	[3, 6, 12, 13, 14, 15, 17, 18, 19, 20]	95.99	96.15	95.99	95.97
	S9	Oversample by value of top 50% (940)	All class except 2 dan 5	96.23	96.48	96.44	96.25
	S10	Oversample by value of top 75% (1410)	All class except 2	96.44	96.60	96.44	96.43
Distribution	S11	Oversample by value of mean (549)	[3, 6, 12, 13, 14, 15, 17, 18, 19, 20]	95.92	96.11	95.92	95.89
	S12	Oversample by value of Q1 (252)	[3, 12, 13, 15, 19]	96.37	96.50	96.37	96.34
	S13	Oversample by value of median (522)	[3, 6, 12, 13, 14, 15, 17, 18, 19, 20]	95.82	96.03	95.82	95.80
	S14	Oversample by value of Q3 (724)	All class except 1, 2, 5, 8, 16	96.55	96.70	96.55	96.53
Base Performance Model	S15	Oversample by value of recall < 90%	[3, 7, 12, 15, 17]	95.89	96.08	95.89	95.89
	S16	Oversample by value of F1-score < 90%	[3, 12, 15]	96.30	96.41	96.30	96.29

Table 6: Classification model performance using 5-Fold and 10-Fold cross validation

Scenario	Num. of Fold	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Original Data	5	95.52 ± 0.43	95.68 ± 0.46	95.52 ± 0.43	95.40 ± 0.54
Original Data + Oversampling with GAN	5	**96.83 ± 0.18**	**96.97 ± 0.17**	**96.83 ± 0.18**	**96.82 ± 0.18**
Original Data	10	95.85 ± 0.54	96.04 ± 0.52	95.85 ± 0.54	95.79 ± 0.56
Original Data + Oversampling with GAN	10	96.73 ± 0.50	96.89 ± 0.49	96.73 ± 0.50	96.71 ± 0.50

performance is better compared to the original data. This shows that the data generated is evenly distributed, and the model is able to generalize quite well.

However, while these results show improved generalization within the current dataset, this may not hold for larger or more complex datasets. The synthetic data generated by EBGAN is based on patterns from the original data, which may limit its ability to capture variations in more diverse datasets, such as different handwriting styles, degraded manuscripts, or varying script sizes. As the dataset grows, the model may face challenges in maintaining consistent performance, especially if the synthetic data cannot represent rare or complex features accurately. The increased standard deviation in the 10-Fold Cross Validation suggests that the model's stability decreases with more varied data splits, indicating potential limitations in EBGAN's ability to generalize to more complex scenarios.

4.3. Comparation with Previous Methods

In this scenario, this study compares the effectiveness of GAN (EBGAN) with several previously used data augmentation methods, namely traditional methods (translation and rotation), SMOTE, B-SMOTE, and ADASYN.

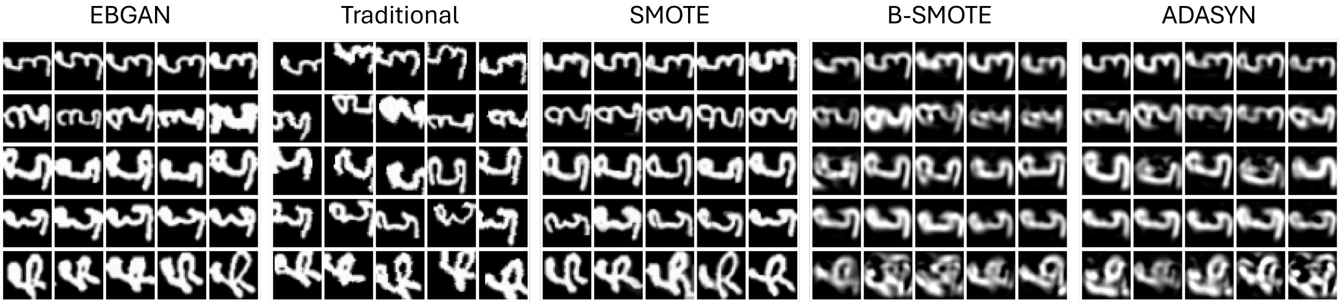

Fig. 10: Visual comparison of synthetic data using EBGAN, Traditional, SMOTE, B-SMOTE, and ADASYN

Table 7: Classification model performance compared with previous methods

Scenario	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
EBGAN	96.56 ± 0.20	96.78 ± 0.16	96.56 ± 0.20	96.56 ± 0.19
Traditional	96.81 ± 0.31	96.98 ± 0.27	96.81 ± 0.31	96.80 ± 0.31
SMOTE	**97.18 ± 0.14**	**97.32 ± 0.15**	**97.18 ± 0.14**	**97.18 ± 0.14**
B-SMOTE	96.12 ± 0.31	96.36 ± 0.31	96.12 ± 0.31	96.11 ± 0.31
ADASYN	96.12 ± 0.43	96.33 ± 0.39	96.12 ± 0.43	96.11 ± 0.42

The main purpose of this comparison is to evaluate the ability of synthetic data generated by GAN to improve the performance of the classification model, compared to the commonly used conventional augmentation methods.

Fig. 10 shows the comparative visualization of synthetic data from each method, from the visualization it can be seen that the EBGAN method is visually able to replicate the original data quite well. While the traditional method produces a more varied image due to differences in character size and position, but in some cases makes the resulting character cropped. In the SMOTE method, the resulting character tends to produce an interpolation between two images, seen from the characters that seem to have shadows. This shows the limitations of SMOTE in producing synthetic data limited to existing data, with less variation. Meanwhile, B-SMOTE and ADASYN show visually shapeless and tend to be difficult to recognize. This shows the superiority of the EBGAN method which is more visually consistent and indicates more variety than the previous methods.

The comparison of various augmentation methods in Table 7 shows that the B-SMOTE and ADASYN methods produce relatively low performance, as indicated by the visualization of characters that are not representative of the original data. Meanwhile, the traditional method is able to produce better performance, but the resulting truncated characters as in the visualization allow overfitting when used in larger numbers. In addition, the SMOTE method produces the best performance compared to other methods, even compared to EBGAN. This can happen because SMOTE is used on limited data, allowing SMOTE to fit data based on existing data, but limited to the variety of data available and allowing it to produce the same data. In the proposed GAN method using EBGAN, the resulting performance is quite good although lower than the traditional method and SMOTE. This shows the potential for better development of GANs to generate characters which support the improvement of classification model performance. Overall, GAN-based augmentation is a good alternative, especially in scenarios with imbalanced data.

5. Conclusion

This study evaluates the effectiveness of synthetic data generated using Enhanced Balancing GAN (EBGAN) in overcoming data imbalance in script recognition. Based on the results of quantitative analysis using Fréchet Inception Distance (FID) and Structural Similarity Index (SSIM), as well as visual evaluation, the synthetic data generated has a quality close to the original data.

Experiments show that the combination of original and synthetic data can improve accuracy, precision, recall, and F1-score compared to using them separately with F1-score of 95.93% increasing to 96.77%. In addition, the oversampling strategy of synthetic data proved to be effective in meeting the data requirements for training the recognition model. The selection of minority classes and the selection of thresholds based on percentage, distribution, and model performance in oversampling can be used as a guide to improve script recognition performance. Validation using 5-Fold Cross Validation and 10-Fold Cross Validation shows that the generated model is able to generalize evenly for each subset of generated data.

Compared to previous methods, EBGAN is able to produce synthetic data that is more varied and has better visual quality, thus it can be used as an alternative in improving the performance of script recognition models. However, further research is still needed to optimize the performance of GAN in supporting script recognition more effectively and efficiently.

CRediT Authorship Contribution Statement

Muhammad A. Faizin: Conceptualization, Methodology, Software, Formal analysis, Investigation, Resources, Data Curation, Writing – Original Draft, Writing – Review & Editing, Visualization, Project Administration. **Nanik Suciati:** Conceptualization, Methodology, Validation, Formal analysis, Writing – Review & Editing, Supervision, Project Administration, Funding Acquisition. **Chastine Fatichah:** Conceptualization, Methodology, Validation, Formal analysis, Writing – Review & Editing, Supervision, Project Administration, Funding Acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

The data supporting this study are available in Zenodo at DOI 10.5281/zenodo.14897894 upon reasonable request.

Declaration of Generative AI and AI-assisted Technologies in The Writing Process

The authors used generative AI to improve the writing clarity of this paper. They reviewed and edited the AI-assisted content and take full responsibility for the final publication.

References

[1] D. Iskandar, S. Hidayat, U. Jamaludin, and S. Mukti Leksono, “Javanese script digitalization and its utilization as learning media: an etnopedagogical approach,” *International Journal of Mathematics and Sciences Education*, vol. 1, no. 1, pp. 21–30, Jun. 2023, doi: 10.59965/ijmsed.v1i1.24.

[2] S. O. Robson, “Javanese script as cultural artifact: Historical background,” *RIMA: Review of Indonesian and Malaysian Affairs*, vol. 45, no. 1 and 2, pp. 9 – 36, 2011, doi: 10.3316/ielapa.422100940117015.

[3] Y. Sugianela and N. Suciati, “Ekstraksi Fitur pada Pengenalan Karakter Aksara Jawa Berbasis Histogram of Oriented Gradient,” *JUTI: Jurnal Ilmiah Teknologi Informasi*, vol. 17, no. 1, pp. 64–72, 2019.

[4] E. Paulus, M. Suryani, S. Hadi, and F. Natsir, “An initial study to solve imbalance sundanese handwritten dataset in character recognition,” *Proceedings of the 3rd International Conference on Informatics and Computing, ICIC 2018*, Oct. 2018, doi: 10.1109/IAC.2018.8780496.

[5] A. R. Widiarti, R. Pulungan, A. Harjoko, Marsono, and S. Hartati, “A Proposed Model for Javanese Manuscript Images Transliteration,” in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Oct. 2018. doi: 10.1088/1742-6596/1098/1/012014.

[6] D. Tursina, S. R. Anggraeni, C. Fatichah, M. Munir, and I. Subakti, “Metode Hibrida Oversampling untuk Menangani Imbalanced Multi-Label,” *JUTI: Jurnal Ilmiah Teknologi Informasi*, vol. 22, no. 1, p. 32, 2024, doi: 10.12962/j24068535.v22i1.a1208.

[7] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *J Big Data*, vol. 6, no. 1, p. 60, 2019, doi: 10.1186/s40537-019-0197-0.

[8] T. Dam, M. M. Ferdaus, S. G. Anavatti, S. Jayavelu, and H. A. Abbass, “Does Adversarial Oversampling Help us?,” Aug. 2021.

[9] G. Huang and A. H. Jafari, “Enhanced balancing GAN: minority-class image generation,” *Neural Comput Appl*, vol. 35, no. 7, pp. 5145–5154, Mar. 2023, doi: 10.1007/s00521-021-06163-8.

[10] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[11] D. B. Soomro, W. ChengLiang, M. Ashraf, D. A. AlHammadi, S. Alsubai, C. Medaglia, N. Innab, and M. Umer, “Automated dual CNN-based feature extraction with SMOTE for imbalanced diabetic retinopathy classification,” *Image and Vision Computing*, vol. 159, p. 105537, 2025, doi: 10.1016/j.imavis.2025.105537.

[12] H. Han, W.-Y. Wang, and B.-H. Mao, “Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning,” 2005, pp. 878–887. doi: 10.1007/11538059_91.

[13] Y. Sun *et al.*, “Borderline SMOTE Algorithm and Feature Selection-Based Network Anomalies Detection Strategy,” *Energies (Basel)*, vol. 15, no. 13, p. 4751, Jun. 2022, doi: 10.3390/en15134751.

[14] Haibo He, Yang Bai, E. A. Garcia, and Shutao Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, IEEE, Jun. 2008, pp. 1322–1328. doi: 10.1109/IJCNN.2008.4633969.

[15] R. Mitra, A. Bajpai, and K. Biswas, "ADASYN-assisted machine learning for phase prediction of high entropy carbides," *Computational Materials Science*, vol. 223, p. 112142, 2023, doi: 10.1016/j.commatsci.2023.112142.

[16] W. Zhang and G. Ding, "A Study on the GAN-Stacking Model Framework for Fraud Dataset," in *2023 4th International Conference on Machine Learning and Computer Application*, New York, NY, USA: ACM, Oct. 2023, pp. 310–315. doi: 10.1145/3650215.3650270.

[17] Y. Li, F. Bai, C. Lyu, X. Qu, and Y. Liu, "A systematic review of generative adversarial networks for traffic state prediction: Overview, taxonomy, and future prospects," *Information Fusion*, vol. 117, p. 102915, 2025, doi: 10.1016/j.inffus.2024.102915.

[18] C. Lv, W. Wang, L. Ding, W. Zhou, Z. Ma, and J. Jia, "Local data augmentation of biomass gasification performance prediction based on improved generative adversarial network," *Energy*, vol. 330, p. 136808, 2025, doi: 10.1016/j.energy.2025.136808.

[19] Z. Luo, "Automatic text generation system for endangered languages based on conditional generative adversarial networks," *Systems and Soft Computing*, vol. 7, p. 200306, 2025, doi: 10.1016/j.sasc.2025.200306.

[20] G. Zhao, P. Liu, K. Sun, Y. Yang, T. Lan, and H. Yang, "Research on data imbalance in intrusion detection using CGAN," *PLoS One*, vol. 18, no. 10, p. e0291750, Oct. 2023, doi: 10.1371/journal.pone.0291750.

[21] A. Odena, C. Olah, and J. Shlens, "Conditional Image Synthesis with Auxiliary Classifier GANs," in *Proc. ICML '17: 34th International Conference on Machine Learning*, May 2017, pp. 2642–2651. doi: 10.5555/3305890.3305954.

[22] D. Cui, R. Zhou, H. Li, R. Hua, Z. Chen, H. Liu, L. Dong, Z. Cheng, and X. Wang, "Rotor fault diagnosis of centrifugal pumps in nuclear power plants based on CWGAN-GP-CNN for imbalanced dataset," *Progress in Nuclear Energy*, vol. 178, p. 105500, 2025, doi: 10.1016/j.pnucene.2024.105500.

[23] G. Mariani, F. Scheidegger, R. Istrate, C. Bekas, and C. Malossi, "BAGAN: Data Augmentation with Balancing GAN," *arXiv preprint*, Mar. 2018.

[24] G. Cao and S.-I. Kamata, "Data Augmentation for Historical Documents via Cascade Variational Auto-Encoder," in *2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, IEEE, Sep. 2019, pp. 340–345. doi: 10.1109/ICSIPA45851.2019.8977737.

[25] A. Ali-Gombe and E. Elyan, "MFC-GAN: Class-imbalanced dataset classification using Multiple Fake Class Generative Adversarial Network," *Neurocomputing*, vol. 361, pp. 212–221, Oct. 2019, doi: 10.1016/j.neucom.2019.06.043.

[26] M. Eltay, A. Zidouri, I. Ahmad, and Y. Elarian, "Generative adversarial network based adaptive data augmentation for handwritten Arabic text recognition," *PeerJ Comput Sci*, vol. 8, p. e861, Jan. 2022, doi: 10.7717/peerj-cs.861.

[27] J. Cai, L. Peng, Y. Tang, C. Liu, and P. Li, "TH-GAN: Generative Adversarial Network Based Transfer Learning for Historical Chinese Character Recognition," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, Sep. 2019, pp. 178–183. doi: 10.1109/ICDAR.2019.00037.

[28] Z. Yuan and S. Kamata, "Data Augmentation for Ancient Characters via Semi-MixFontGan," in *2020 Joint 9th International Conference on Informatics, Electronics &amp; Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision &amp; Pattern Recognition (icIVPR)*, IEEE, Aug. 2020, pp. 1–6. doi: 10.1109/ICIEVicIVPR48672.2020.9306588.

[29] M. A. Faizin, "Deteksi Aksara Jawa Menggunakan YOLO untuk Transliterasi Berbasis LSTM pada Manuskrip Jawa Kuno," Institut Teknologi Sepuluh Nopember, 2023. Accessed: May 19, 2024. [Online].

[30] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved Training of Wasserstein GANs," *arXiv preprint*, Mar. 2017.

[31] Z. Qu, G. Fan, Z. Zhao, L. Jia, J. Shi, and J. Ai, "Synthetic aperture radar ground target image generation based on improved Wasserstein generative adversarial networks with gradient penalty," *J Appl Remote Sens*, vol. 17, no. 03, Jul. 2023, doi: 10.1117/1.JRS.17.036501.

[32] L. O. Guertler, A. Ashfahani, and A. T. Luu, "How to train your draGAN: A task oriented solution to imbalanced classification," *arXiv preprint*, Nov. 2022, doi: 10.48550/arXiv.2211.10065.

[33] S. J. Haddadi, A. Farshidvard, F. dos Santos Silva, J. C. dos Reis, and M. da Silva Reis, "Customer churn prediction in imbalanced datasets with resampling methods: A comparative study," *Expert Systems with Applications*, vol. 246, p. 123086, 2024, doi: 10.1016/j.eswa.2023.123086.

[34] L. J. Ba and R. Caruana, "Do Deep Nets Really Need to be Deep?," in *Advances in Neural Information Processing Systems*, 2014, vol. 27, p. , [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2014/file/b0c355a9dedccb50e5537e8f2e3f0810-Paper.pdf.