Research Article

A novel GAN-transformer framework for early Brahmi script generation and recognition

Pabasara Surasinghe[1) and Kokul Thanikasalam*[2)

[1)Department of Computing, Rajarata University, Mihintale, Sri Lanka
[2)Department of Computer Science, University of Jaffna, Thirunelvely, Sri Lanka

Abstract

Recognizing ancient scripts is crucial for understanding the historical, cultural, and linguistic context of past civilizations. However, the recognition of Early Brahmi letters from Sri Lankan inscriptions faces significant challenges due to scarce digitized data, degradation of inscriptions, and visual similarity among characters. This study introduces BrahmiGAN, a novel Generative Adversarial Network (GAN) designed to generate realistic synthetic Early Brahmi letters, addressing data limitations that hinder recognition model training. Using a benchmark dataset of 73 inscriptions, 21,195 high-quality synthetic images were generated from 888 real samples. These synthetic images demonstrated high fidelity, validated through feature-based, raster-based, and vector-based evaluations, and achieved a 92.15% approval rate from human experts. Furthermore, a vision Transformer-based ensemble model integrating Pyramid Vision Transformer and Swin Transformer is proposed for Early Brahmi letter recognition. A classification accuracy of 96.06% was attained by the ensemble model when trained on combined synthetic and real images, outperforming existing methods and surpassing the same model trained exclusively on real images. The generated dataset is publicly available to support future research (https://zenodo.org/records/14961074).

Keywords: Brahmi letter synthesis, Character recognition, Data scarcity, GANs, Transformers

1. Introduction

Recognizing ancient scripts is essential for understanding the history, culture, language, and social structures of past civilizations. These scripts serve as repositories of cultural and historical knowledge, yet their recognition and analysis face significant challenges, particularly due to the scarcity of digitized image data [1]. While well-annotated, large-scale datasets are critical for training models, compiling such data is labor-intensive and requires specialized archaeological expertise [2, 3]. Additionally, obtaining undamaged images of ancient scripts is inherently difficult: most were inscribed on perishable materials, which have deteriorated over centuries due to environmental factors and decay. Compounding this issue, ancient societies relied on manual inscription methods, and the high cost of materials, combined with the scarcity of skilled scribes, limited written records to a small elite. In this context, developing computational models for generating ancient scripts emerges as a promising solution to address data scarcity and advance the preservation of historical texts in computational archaeology.

The Brahmi script, one of the oldest writing systems in the Indian subcontinent, serves as the foundation for many modern South and Southeast Asian scripts, including Tamil, Sinhala, Kannada, and Devanagari [4]. It was introduced to Sri Lanka during the reign of King Ashoka (3rd century BCE), when Buddhist missionaries disseminated the script alongside religious teachings across the region. In Sri Lanka, the script evolved into a distinct regional variant known as Early Brahmi, which was used from the 3rd century BCE to the 1st century CE and remains the island's earliest attested writing system. Derived from its mainland Indian counterpart, Early Brahmi consists of 40 characters: 6 vowels and 34 consonants, as shown in Figure 1.

The majority of Early Brahmi inscriptions in Sri Lanka were carved into cave walls and rock surfaces, exhibiting angular, straight-lined features that distinguish them from later scripts [5]. Despite Sri Lanka's extensive corpus of over 4,000 inscriptions, Early Brahmi texts remain relatively scarce. These inscriptions are typically brief and contain fewer characters than those from later periods in Sri Lanka's epigraphic evolution. In addition, the angularity and linearity of Early Brahmi characters make them prone to weathering, and poor environmental conditions often distort their shapes. Moreover, the morphological similarity between many Early Brahmi letters complicates automated recognition efforts. Nevertheless, deciphering these inscriptions is vital for reconstructing Sri Lanka's early social, cultural, and religious contexts.

In recent years, several computational approaches have been proposed for recognizing Brahmi characters extracted from Sri Lankan inscriptions [6-8]. While these methods demonstrate promising results, their accuracy remains suboptimal compared to human expert performance. Furthermore, many were trained and evaluated on proprietary datasets, limiting reproducibility and hindering comparative analysis. Compounding this issue, the reliance on limited sample sizes raises concerns about overfitting and poor generalizability to diverse inscription contexts. Some studies also do not disclose the exact number of samples or annotation protocols

used, further reducing methodological transparency. These limitations collectively highlight the urgent need for scalable, reproducible frameworks to improve both recognition accuracy and robustness in Early Brahmi script analysis.

a	ā	i	u	e	o	ka	kha	ga	gha
⅄	⅄	∴	∟	△	Ζ	✝	ʔ	∧	⊔
ṅa	ca	cha	ja	jha	ña	ṭa	tha	ḍa	ḍha
·	↺	Ⴔ	Ε	Ⱶ	ⱨ	C	Ο	⊐	⊙
ṇa	ta	tha	da	dha	na	pa	pha	ba	bha
Ⱦ	⟨	⊙	⥾	D	⊥	⊔	⍵	☐	⊓
ma	ya	ra	la	va	śa	ṣa	sa	ha	ḷa
⊌	↓	Ι	⏝	♂	⸙	�└	⌇	⊔	φ

Figure 1 Early Brahmi alphabet. The first 6 letters are vowels and the remaining 34 are consonants.

This study introduces BrahmiGAN, a novel Generative Adversarial Network (GAN) designed to synthesize realistic Early Brahmi characters and mitigate the scarcity of training data for Sri Lankan inscriptions. A recently developed benchmark dataset, consisting of real images of letters extracted from 73 Early Brahmi inscriptions, is utilized for generating synthetic images of Early Brahmi letters, as it has been identified as the only publicly accessible dataset in this domain [9]. From this dataset, 23 letter classes are selected for the study, while the remaining letters are excluded due to unavailability or insufficient samples resulting from their rare occurrence in inscriptions.

The proposed BrahmiGAN is engineered to synthesize realistic Early Brahmi characters by replicating their defining paleographic attributes. A SqueezeNet-based [10] discriminator architecture is integrated into BrahmiGAN, enhanced with channel and spatial attention mechanisms, residual blocks, skip connections, and multi-scale feature extraction for optimized adversarial training. Evaluations across vector and raster formats revealed that proposed BrahmiGAN produces more epigraphically faithful character reconstructions than existing approaches. The perceptual quality of generated outputs was validated through expert assessments, while the synthetic data's utility was empirically confirmed in downstream recognition tasks.

In addition to the proposed GAN in this study, an ensemble model is introduced for the recognition of Early Brahmi letters. Unlike previous approaches reliant on convolutional neural networks, transformer-based architectures are prioritized, owing to their superior ability to capture global morphological features inherent to the script. The ensemble framework integrates the Pyramid Vision Transformer (PVT) [11] and Swin Transformer [12], which are employed to process both standard and skeletonized letter representations. To optimize recognition accuracy, confidence scores from both models are averaged to derive the final classification output. Training of the ensemble classifier is conducted using a hybrid dataset comprising real and synthetically generated images. Experimental validation on a real-image test set confirms that state-of-the-art performance is achieved by the proposed classifier in Early Brahmi letter recognition.

In summary, the contributions of this study are threefold:
1. A novel generative adversarial network, BrahmiGAN, is introduced to synthesize Early Brahmi characters. The model is demonstrated to surpass baseline generative methods in realism, as evidenced by feature-based, raster-based, and vector-based evaluations, alongside validation by epigraphic experts.
2. A Transformer-based ensemble classifier is developed for precise recognition of Early Brahmi letters, leveraging complementary architectures to address morphological complexity.
3. The proposed GAN-Transformer framework demonstrates exceptional performance on the test set of real images. Furthermore, it surpasses similar approaches based on reported recognition results.

2. Related work

In this section, GAN-based studies on the synthesis of ancient character images, the recognition of Sri Lankan Early Brahmi characters, and vision Transformer, and CNN based ancient character recognition approaches are reviewed, as they are closely related to this research.

2.1 GANs in ancient script generation

Over the past decade, several image synthesis approaches using GANs have been proposed. However, only a few have focused on generating images of ancient scripts to address data scarcity, as these scripts have limited available resources. Yuan and Kamata [13] introduced Semi-MixFontGAN, a semi-supervised GAN model designed to generate font-consistent characters from ancient documents, thereby enhancing recognition accuracy. The model addresses challenges related to unbalanced datasets and intra-class font variability by integrating labeled, unlabeled, and generated data. Vögtlin et al. [14] employed a CycleGAN based approach to generate synthetic historical documents in English using a two-step process. First, they created template documents with user-specified

content and structure. Second, they transferred the style of unlabelled historical images to these templates while preserving the original text and layout. Their model significantly reducing the character error rate (CER) compared to other approaches.

Similarly, Perrin et al. [15] introduced a two-stage method for reconstructing damaged ancient Greek manuscripts using GANs. In the first stage, they reconstructed the text by pasting character patches into their probable locations. The second stage focused on background reconstruction through inpainting techniques. Vidal-Gorène et al. [16] employed GANs and style transfer techniques to generate synthetic handwritten lines for under-resourced scripts. Their approach, based on ScrabbleGAN [17], demonstrated feasibility across various scripts and was benchmarked against CycleGAN [18].

Although a few GAN-based approaches have been proposed to address the training data deficiency in ancient character recognition, they have not undergone a comprehensive evaluation of the realism of the generated characters. As a result, their effectiveness in addressing data scarcity while mitigating issues such as mode collapse and overfitting remains questionable. Furthermore, the architectures of these GAN-based models differ from one another, as they were designed for different languages and, therefore, incorporate unique characteristics. Moreover, although several approaches have been proposed to address the issue of training data deficiency in other ancient scripts, no notable work has been done to generate synthetic images of Brahmi letters, despite its limited resources.

2.2 Brahmi character recognition approaches

Over the past decade, a few studies have been conducted on Brahmi character recognition using letter images extracted from inscriptions; however, the majority of these studies have focused on Indian Brahmi letters, which are not directly applicable to recognizing Brahmi letters used in Sri Lanka due to the distinct diacritics and symbols unique to Sri Lankan Brahmi script [19-21].

Recently, several approaches have been proposed for recognizing Sri Lankan Early Brahmi characters. Gunasekara et al. [6] developed a deep learning-powered mobile application for recognizing and translating Early Brahmi characters. They utilized a dataset of 250-300 inscription images and applied semantic segmentation using a multi-class U-Net model, which resulted in improved recognition performance. However, given that their application considered only 10 to 15 letters, the generalizability of their model is questionable. Moreover, they did not report quantitative outcomes, which are essential for assessing the model's overall effectiveness and enabling comparative analysis with other methods.

Wickramarathna and Ranathunga [7] proposed a data-driven method for correcting OCR errors in Brahmi character arrays and generating Sinhala meanings. They employed bigram and trigram language models and achieved high accuracy in word detection, OCR error detection, and OCR error correction. Their method also improved translation accuracy, particularly for noisy or incomplete inscriptions, demonstrating precision and recall rates of up to 94% and 90%, respectively. Wijerathna et al. [8] utilized deep learning and NLP techniques to automate the recognition and translation of Brahmi characters. Although their VGG16-based model achieved a classification accuracy of 93.33%, its generalizability is limited because the study considered only 10 classes and employed a small number of images for training and testing.

Although previous studies on Sri Lankan Early Brahmi script recognition have demonstrated promising performance, their accuracy remains significantly constrained by data deficiency issues [6-8]. Consequently, these approaches predominantly rely on hand-crafted features. Furthermore, they were trained and evaluated on proprietary datasets with limited training and testing samples, which are not publicly accessible. In some cases, the exact number of samples used for training and testing was not specified, casting doubt on the reliability of their reported performance metrics. These limitations severely restrict the generalizability of such approaches, making them inadequate for real-world applications.

Beyond Sri Lanka, a few approaches have been proposed for the detection of Indian and other Brahmi characters. Vincen and Samsuryadi [22] proposed a VGG16-based model for Indonesian Brahmi script classification. Agrawal et al. [23] developed an OCR system for Ashokan Brahmi inscriptions, and achieved 95.94% validation accuracy using MobileNet. Gautam et al. [24] introduced a deep CNN with dropout and Gabor filters for Brahmi word recognition, achieving 92.47% accuracy. All these studies employed conventional data augmentation techniques which, aside from increasing dataset size, do not address the fundamental challenge: geometric transformations inherently distort the precise angular relationships and stroke structures characteristic of Early Brahmi characters, potentially producing training samples that contravene paleographic principles.

2.3 Transformers in ancient character recognition

Ancient character recognition has traditionally been approached as a multi-class classification problem, with CNNs demonstrating superior performance until recently. However, in the past two to three years, a limited number of studies have explored the use of Transformers as classification models, reporting improved performance over CNN-based approaches. Transformers excel in capturing long-range dependencies and global contextual relationships within an image of an ancient character, enabling robust recognition even when portions of the characters are damaged.

Recently, Bhuvaneswari and Kathiravan [25] employed vision Transformers for the recognition of ancient Tamil temple inscription images. Their study demonstrated that vision Transformers are particularly effective in handling the intricate curves and connected components characteristic of ancient Tamil scripts. Similar to their work, Murugan and Visalakshi [26] proposed a hybrid approach that integrates vision Transformers with ResNet for the recognition of ancient Tamil scripts, achieving a recognition rate of 98.8%. Lalitha et al. [27] further demonstrated the versatility of Transformer architectures in handling multiple Indic scripts. Their encoder-decoder Transformer processes image patches at the encoder level and predicts character sequences at the decoder level. The advantages of Transformer architectures in Indic script recognition were also highlighted in the comprehensive review by Mostafa et al. [28]. Cheema et al. [29] demonstrated the effectiveness of Transformer-based approaches for low-resource Indic languages. Their work tackles the challenges of developing OCR systems for languages with limited linguistic resources.

Although vision Transformers have demonstrated remarkable performance in recognizing ancient scripts, their reliance on extensive training data presents a significant challenge. This limitation is particularly pronounced in the context of ancient scripts, where the availability of annotated data is inherently scarce compared to modern datasets. Unlike CNNs, which can achieve reasonable performance with smaller datasets due to their inherent inductive biases, vision Transformers require large-scale data to effectively learn spatial relationships and generalize well. Furthermore, despite the advancements in vision Transformer-based approaches for various ancient scripts, there remains a notable research gap in their application to Sri Lankan Brahmi script recognition, which features unique diacritics and symbols that demand specialized attention.

3. Methodology

In this section, BrahmiGAN, a novel Generative Adversarial Network (GAN) framework, is introduced for synthesizing realistic Early Brahmi letter images. Additionally, a vision Transformer-based ensemble model is presented, for the recognition of Early Brahmi letters. By integrating these two architectures, the challenges of generating high-quality synthetic data and accurately recognizing ancient Brahmi script are addressed. The section begins with a description of the real-image dataset, followed by details of the proposed BrahmiGAN architecture, its training process, and loss functions. Finally, an in-depth explanation of the proposed ensemble classifier architecture for recognition is provided.

3.1 Collection of real images of early brahmi letters

In this study, real images of Sri Lankan Early Brahmi letters were obtained from a publicly available benchmark dataset [9]. The dataset comprises 1,269 pre-processed images of Early Brahmi letters, extracted from 73 Sri Lankan inscription estampages belonging to the third century BCE to the first century CE, which were utilized for this research. In [9], a series of pre-processing steps were applied during dataset creation to obtain clearer images from raw estampage data. These steps included converting the images to grayscale, applying Non-Local Means denoising to reduce noise, using Gaussian blur to smooth letter contours, applying a threshold function to produce binary images, inverting white backgrounds to black through the image negative operation, isolating the most prominent white blob while removing remaining blobs and background using morphological image processing techniques, and finally resizing all individual letter images to 64×64 pixels while preserving their original aspect ratio. Although 40 Early Brahmi letters were utilized in Sri Lanka, this study focuses on 23 letters, as the remaining 17 letters are either exceedingly rare or absent in the only publicly available dataset, as well as in Sri Lankan inscriptions. Out of the total 1,269 real images in the dataset, 888 images (70%) were allocated for training the proposed BrahmiGAN model, and the remaining 381 images were used for testing. No separate validation set was utilized, as hyperparameter optimization was performed using Optuna's built-in cross-validation mechanisms. To train the proposed Vision Transformer-based recognition model, 21,195 synthetically generated images were used in addition to the 888 real training images. The 381 real test images were used exclusively to evaluate the recognition performance and to assess the model's ability to identify authentic Early Brahmi letters. Figure 2 provides samples of Early Brahmi letter images. A summary of the distribution of training and testing images for each class is presented in Table 1.
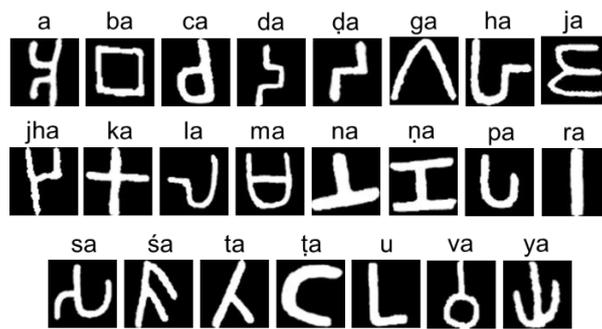

Figure 2 Sample real images of Early Brahmi letters.

Table 1 Number of training and testing images of Early Brahmi letters used in this study.

Letter	Training images		Testing images
	Real	Synthetic	Real
a	36	970	15
ba	28	961	12
ca	24	824	10
da	48	965	21
ḍa	8	975	4
ga	76	950	33
ha	38	952	17
ja	17	683	7
jha	13	839	5
ka	61	978	26
la	32	870	13
ma	55	990	23
na	57	999	25
ṇa	50	767	21
pa	58	993	25
ra	28	998	12
sa	36	823	15
śa	78	937	34
ta	68	975	29
ṭa	18	959	7
u	10	988	5
va	18	937	8
ya	31	862	14

Prior to feeding the real images into the proposed BrahmiGAN model, data augmentation was applied in a balanced manner to improve the model's ability to learn character variations and mitigate overfitting. Various augmentation techniques, including random rotations, zoom adjustments, and translations, were employed on the real images while preserving the structural integrity and identity of the Brahmi letters. These augmentations serve to artificially expand the diversity of the training dataset, thereby enhancing the model's robustness and adaptability to variations in the input data.

3.2 Architecture of the generator of the proposed BrahmiGAN

In this study, the WGAN-GP [30] is used as the baseline model for designing the proposed BrahmiGAN. WGAN-GP is chosen because it effectively addresses the mode collapse issue, which is a major challenge in generating datasets for low-resource ancient scripts.

The generator in the proposed BrahmiGAN model was meticulously designed to address the challenges associated with generating high-quality images of Sri Lankan Early Brahmi letters. These letters are characterized by intricate curves, variable stroke widths, and asymmetric shapes. To achieve this, the generator architecture integrates advanced components, including channel attention, spatial attention, residual blocks, skip connections, and dropout layers as illustrated in Figure 3.
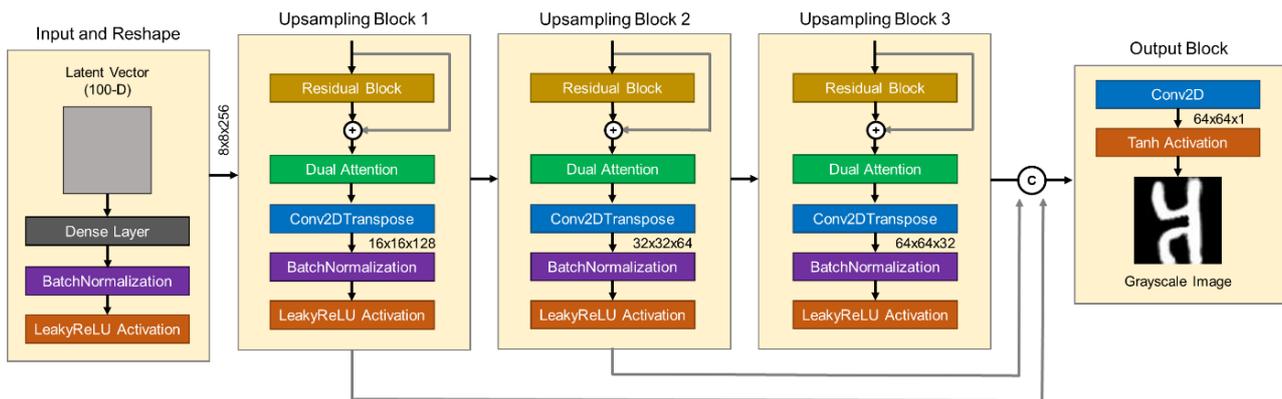

Figure 3 An outline of the proposed Generator architecture.

The generator of the proposed BrahmiGAN transforms a 100-dimensional random noise vector into a 64×64-pixel image of an Early Brahmi letter through a series of upsampling convolutional blocks. Initially, the input noise vector is passed through a fully connected layer, followed by batch normalization and LeakyReLU activation to stabilize the training process and introduce non-linearity. Subsequently, the vector is reshaped into a tensor of dimensions 8×8×256.

The proposed generator consists of three upsampling blocks, each containing a residual block with a skip connection, a dual attention module, a transposed convolutional layer, batch normalization, LeakyReLU activation, and a dropout layer. The residual block is strategically designed to mitigate the vanishing gradient problem during training. Each residual block comprises two convolutional layers with a kernel size of 3×3, a stride of 1, and padding configured to preserve spatial dimensions. These convolutional layers are followed by batch normalization and LeakyReLU activation to stabilize training and introduce non-linearity. A skip connection is integrated into each block, directly passing the input to the output. This design enables the network to learn identity mappings when necessary, ensuring that critical structural features of the Early Brahmi letters are preserved and refined throughout the network.

To further enhance feature learning, the dual attention module incorporates both channel attention and spatial attention mechanisms. First, channel attention is applied to the final features of the residual block, enabling the generator to prioritize the most relevant feature channels by weighting them based on their importance. Subsequently, spatial attention is applied to the resulting features, allowing the generator to focus on specific spatial regions and improve the localization of fine details. This dual attention mechanism enables the generator to produce sharper and more realistic images by emphasizing critical features while suppressing irrelevant ones.

To progressively increase the spatial resolution, the generator employs transposed convolutional layers after each residual and attention module, which double the spatial dimensions while reducing the number of channels by half. These layers are configured with a kernel size of 5×5, a stride of 2 to perform upsampling of the feature maps. In the final layers of the upsampling block, batch normalization stabilizes training, LeakyReLU introduces non-linearity, and dropout with a low rate regularizes the network to prevent overfitting. Together, these components enhance the generator's ability to learn complex features, and generate high-quality, realistic images of Brahmi letters.

In the proposed generator, multi-level and multi-scale feature concatenation is employed to enhance information flow across intermediate layers. This mechanism ensures that low-level features, such as edges and contours, are preserved while higher-level features are progressively refined. Prior to the final block, the output features from all upsampling blocks are upsampled to a consistent spatial resolution and concatenated, enabling the generator to effectively integrate multi-scale representations.

The final block of the generator consists of a transposed convolutional layer that outputs a 64×64×1 tensor, representing a single-channel grayscale image. This layer uses a tanh activation function, which is crucial for stabilizing GAN training by ensuring that the generated images align with the discriminator's input expectations and preventing unbounded pixel values that could destabilize the process.

3.3 Architecture of the discriminator of the proposed BrahmiGAN

The discriminator in the proposed BrahmiGAN model is designed to distinguish between real and generated images of Sri Lankan Early Brahmi letters. The architecture of the discriminator is inspired by SqueezeNet [10], a lightweight and efficient model known for its effectiveness in image classification tasks. The discriminator employs a series of modified Fire Modules, each consisting of a squeeze convolutional layer followed by expand layers, which reduce the number of parameters while maintaining the network's capacity to learn complex features. The proposed discriminator takes a 64×64 grayscale image as input and produces classification scores for real or fake images.

Initially, features of the input image are extracted by a convolutional layer with a 7×7 kernel size, followed by LeakyReLU activation and batch normalization. The larger kernel size is chosen to capture broader patterns and structures in the input image. The extracted features are then passed through four blocks, as shown in Figure 4. The first three blocks each contain two Residual Fire Modules with Dual Attention (RFMD), followed by max pooling and another dual attention mechanism to refine the features. The final block contains two RFMD modules without max pooling or dual attention. This hierarchical structure allows the discriminator to capture both low-level and high-level features, which are essential for accurately distinguishing real and synthetic images of Early Brahmi letters.
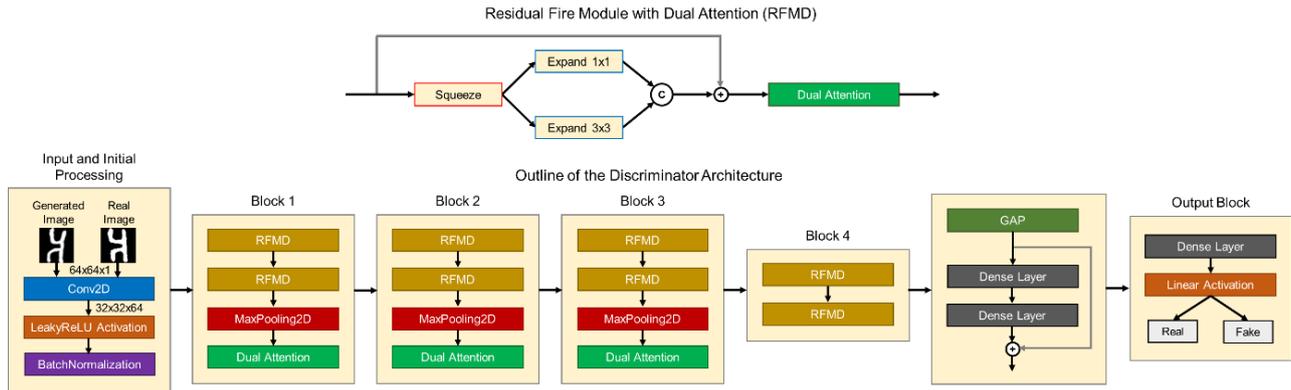

Figure 4 An outline of the proposed Discriminator architecture.

The proposed RFMD is designed by modifying the Fire Module of SqueezeNet [10] to better extract the unique features of Early Brahmi letters. In this module, after the squeeze and expanding convolutional layers, a residual skip connection is introduced to ensure robust gradient flow and feature reuse. Additionally, the RFMD utilizes a dual attention mechanism, consisting of channel and spatial attention, to highlight key feature channels and spatial regions.

After the four blocks, global average pooling is applied to summarize the spatial information into a compact feature vector. Two dense layers with a residual connection are then used to process these features, enabling the network to learn complex decision boundaries. The residual connection improves gradient flow, facilitates feature reuse, and enhances the network's learning capacity. Finally, a dense layer with a linear activation function produces the discriminator's output, providing classification scores for real or fake images.

3.4 Training and loss function of BrahmiGAN

The loss functions for both the generator and discriminator were designed to optimize the Wasserstein distance, which measures the difference between the distributions of real and generated images. The generator's loss can be expressed as follows, where G(z) represents the generated images and D(G(z)) denotes the discriminator's output for these images.

$$L_G = -E\big[D\big(G(z)\big)\big] \tag{1}$$

The discriminator's loss function was designed to maximize the difference between its outputs for real and generated images, while incorporating a gradient penalty term to enforce the Lipschitz constraint. The discriminator's loss can be expressed as follows, where x represents the real images, is a linear interpolation between real and generated images, and λ is the gradient penalty coefficient.

$$L_D = E_{x \sim pR(x)}\big[\big(D(x)\big)\big] - E_{z \sim p_z(Z)}\big[D\big(G(z)\big)\big] + \lambda E_{\hat{x} \sim p_{\hat{x}}}\left[\left(\big|\big|\nabla_{\hat{x}} D(\hat{x})\big|\big|_2\right)^2\right] \tag{2}$$

The Adam optimizer is used in training with a learning rate of 0.0001, β1 of 0.5, and β2 of 0.9. These hyperparameters were selected based on established best practices for WGAN-GP training and validated through preliminary experiments. The gradient penalty coefficient λ was set to 10, following the original WGAN-GP training protocol. The training process was performed for 80 epochs. A batch size of 8 was chosen to balance computational efficiency and training stability. No early stopping was employed during GAN training, as mode collapse and quality improvements often occur in later epochs. Instead, model checkpoints were saved after each epoch, and the final model was selected based on visual quality assessment and quantitative metrics.

3.5 Generation of synthetic images

After training the proposed BrahmiGAN, a total of 1,000 synthetic images were generated for each of the 23 letter classes. These images underwent rigorous quality evaluation by archaeology research officers, who assessed each sample for historical accuracy and

stylistic consistency. Out of the 23,000 generated images, 21,195 were approved by experts and subsequently incorporated, together with the real training set, to train the proposed letter recognition model in the next phase of the methodology.

3.6 Letter recognition using an ensemble transformer model

The proposed ensemble recognition model treats Early Brahmi letter recognition as a multi-class classification task. Although both synthetic and real Early Brahmi letter images were used to train the proposed recognition model, the evaluation was conducted exclusively on unseen real images. The Vision Transformers were utilized in the proposed recognition model as they are effective in capturing long range dependencies. They are also inherently more flexible in handling varying scales and orientations of characters, which is a common challenge in Early Brahmi recognition due to the degraded and irregular nature of historical inscriptions.

Recognition of Early Brahmi letters is challenging since they exhibit significant intra-class variability due to weathering, inscription surface irregularities, and scribal variations. To handle this challenge, two complementary pre-trained Transformer models, Pyramid Vision Transformer (PVT) [11] and Swin Transformer [12], are fine-tuned, and their performances are ensembled for robust recognition. While PVT excels at capturing multi-scale global features through its pyramid structure, the Swin Transformer excels at capturing neighborhood-level dependencies through its shifted window mechanism. By combining these architectures, the proposed ensemble model leverages their individual strengths while mitigating their respective weaknesses. Furthermore, ensemble methods have been shown to reduce overfitting, which is a critical concern given the limited availability of Early Brahmi training data.

In the proposed approach, both the raw image of an Early Brahmi letter and its corresponding skeletonized image are used for recognition. The raw image contains texture, shading, and fine details, while the skeletonized image captures structural and topological features of the characters, such as strokes, curves, and junctions. Since these two types of image features complement each other, they are utilized together to recognize Early Brahmi letters by training two individual Transformer models.

The skeletonization process begins by converting the input raw image of a letter to grayscale and applying denoising to reduce noise. The resulting image is then blurred using a Gaussian filter to smooth out details, followed by binary thresholding to create a binary image. This binary image is skeletonized to extract the thin, structural representation of the characters. Finally, the skeleton is resized to a fixed resolution of 64×64.

In the proposed recognition model, the raw image of an Early Brahmi letter is processed by a PVT model, while the skeleton image is processed by a Swin Transformer. The Swin Transformer is selected for skeleton images owing to its ability to capture hierarchical features, and long-range dependencies, which are essential for analyzing structural and topological details. On the other hand, PVT is chosen for raw images due to its pyramid structure, which efficiently captures multi-scale textural and contextual features. By leveraging the complementary strengths of these models, the proposed system achieves robust and accurate recognition of Early Brahmi letters. An overview of the recognition process is illustrated in Figure 5.
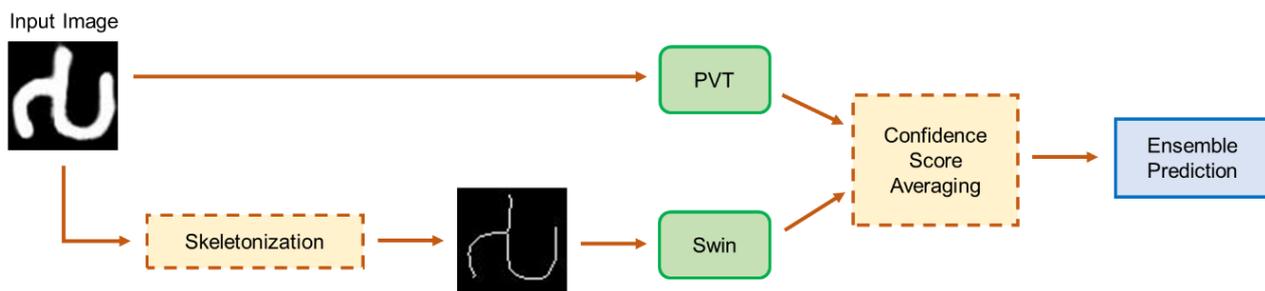

Figure 5 An outline of the ensemble prediction model.

During training, the pre-trained PVT and Swin Transformer classifiers were fine-tuned individually using the corresponding raw and skeleton images, respectively. To enhance the diversity of the training data, augmentation techniques such as random rotations, width and height shifts, and shearing were applied. Owing to the limited amount of training data, only the classification heads of both Transformer models were trained, while all encoder layers were kept frozen to prevent overfitting.

During the training process of each Transformer models, hyperparameter optimization is automated through the Optuna framework [31]. The AdamW optimizer with a weight decay of 0.01 was utilized together with the cross-entropy loss function. Also, Early stopping is implemented through Optuna's pruning mechanism, which terminates poorly performing configurations based on validation accuracy.

After training both Transformer models individually, their predictions are ensembled at test time. For each test image, the raw logits from both models are averaged with equal weights, and the final prediction is obtained by applying the argmax function to the averaged logits. This equal-weight averaging approach was selected based on two main considerations. First, both models exhibited comparable performance on the validation dataset, indicating that neither model should disproportionately influence the ensemble. Second, empirical evaluations demonstrated that equal weighting yielded better results than alternative strategies, including confidence-based and learned weighting schemes.

4. Results and discussion

4.1 Implementation details

The proposed BrahmiGAN model was implemented using TensorFlow and for the letter recognition, Transformer models were obtained from Hugging Face Transformers library. The development environments included both Google Colab and Kaggle. To support future research, the generated synthetic letter dataset is publicly available at (https://zenodo.org/records/14961074).

4.2 Metrics for evaluating letter generation performance

Evaluating the performance of the proposed BrahmiGAN is essential to ensure the quality and realism of the generated Early Brahmi letters. To thoroughly assess the model's effectiveness, a diverse set of evaluation metrics is employed. Feature-based metrics, including Fréchet Inception Distance (FID) [32] and Kernel Inception Distance (KID) [33], measure the statistical similarity between the feature distributions of real and generated images. Raster-based metrics such as Mean Squared Error (MSE) and Structural Similarity Index (SSIM) are used to evaluate pixel-level similarity and error. Additionally, vector-based metrics, including Hausdorff Distance, and Procrustes Distance are used to assess the geometric and structural accuracy of the generated characters.

The FID [32] and KID [33] scores are computed using the pre-trained InceptionV3 model. Features of real and generated images are extracted from the last layer before the classification head. Both metrics compare the extracted feature distributions of real and generated images. Lower FID and KID scores indicate that the generated samples are more similar to the real data in terms of visual realism and statistical distribution.

MSE and SSIM are well-known raster-based metrics used to measure the similarity between real and generated images. A low MSE indicates that the generated images are closer to the real images in terms of pixel-level accuracy, while a high SSIM suggests better structural similarity, including luminance, contrast, and texture. For each pair of real and generated images, MSE and SSIM are computed individually, and the average values of these metrics are reported to provide an overall assessment of image quality.

For the computation of vector-based metrics, stroke vectors are obtained for each real and generated letter. A properly shaped real Early Brahmi letter image is used as the ground truth for each class of letters. The stroke vector is extracted from the binary image of a letter using contour detection, which identifies the boundaries of the letter strokes. These contours are then stacked to form a vector path representing the stroke of the letter. The extracted stroke vectors are normalized and resampled to ensure consistency in the comparison. In this study, stroke vectors are generated and compared between real and generated images, as well as between their corresponding skeletons. An illustration comparing stroke vectors is shown in Figure 6.
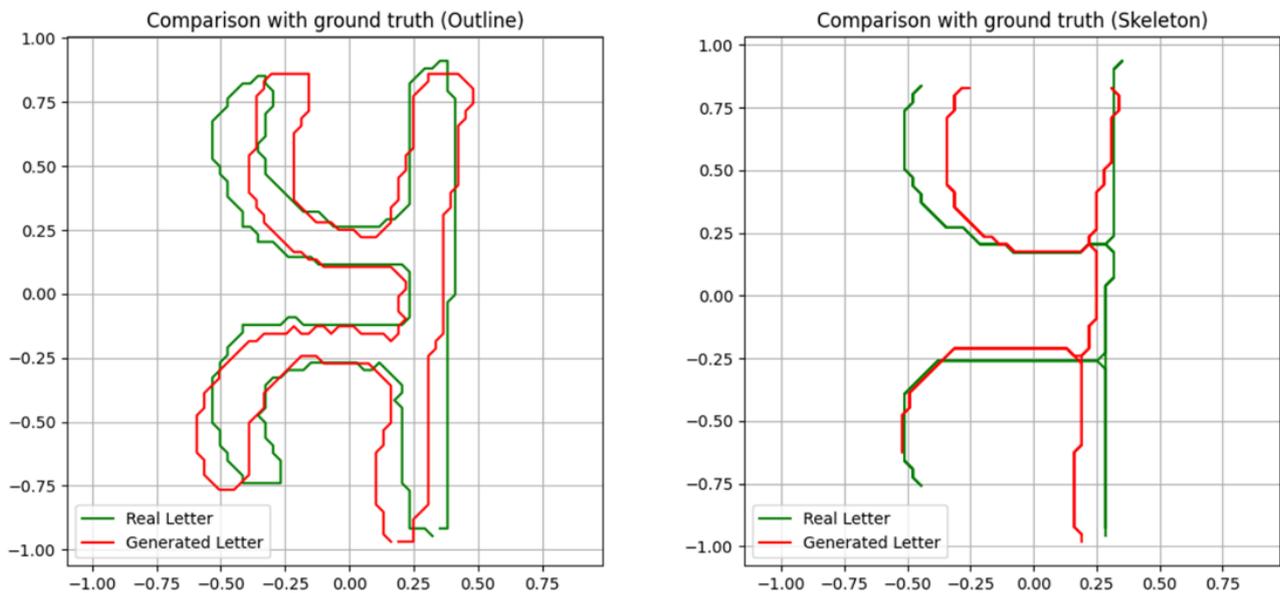

Figure 6 Comparison of stroke vectors between generated and real early Brahmi letters. Left: Comparison of the full letterforms. Right: Comparison of their skeletonized representations.

The Hausdorff Distance [34] and Procrustes Distance [35] are metrics used to measure the similarity between the stroke vectors of real and generated Early Brahmi letters. A lower Hausdorff Distance indicates a closer resemblance between the generated and real letters. The Procrustes Distance, on the other hand, measures intrinsic shape similarity while disregarding differences in position, orientation, and scale, with a lower value indicating better alignment.

In addition to the quantitative measures, a qualitative evaluation is conducted in this study by involving archaeology research officers to ensure the accuracy and cultural authenticity of the generated Early Brahmi letter images. The archaeology research officers carefully inspect all the generated images and exclude those deemed unacceptable based on predefined criteria such as realism, legibility, and adherence to historical standards. To quantify this human evaluation process and assess the BrahmiGAN's performance, an additional metric is introduced as follows:

$$\text{Letter Acceptability} = \frac{\text{Number of acceptable letters}}{\text{Total number of generated letters}} \times 100 \tag{3}$$

4.3 Metrics for evaluating letter recognition performance

To measure the performance of ensemble classification model, accuracy, precision, recall, and F1 score are utilized, and they are measured as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively.

4.4 Experimental results

The letter generation performance of the proposed BrahmiGAN is compared with a baseline GAN model. Both models are based on WGAN-GP and have the same number of blocks in the generator and discriminator. Additionally, they share a similar configuration of filters and convolutional layers. However, the baseline GAN lacks the residual connections, attention mechanisms, multi-scale feature extraction, and the squeeze-and-expand blocks in the discriminator, which are integrated into BrahmiGAN. The comparison is conducted using the same number of images. Table 2 presents a comparison between the baseline GAN and the proposed BrahmiGAN using feature-based metrics (FID and KID) and raster-based metrics (MSE and SSIM).

Table 2 Comparison of results using feature-based and raster-based metrics for images generated by the baseline GAN (BS) and proposed BrahmiGAN (BR). Values for KID and SSIM are scaled by 10^{-1}.

Letter	FID↓		KID↓		MSE↓		SSIM↑	
	BS	BR	BS	BR	BS	BR	BS	BR
a	133.17	**102.37**	1.15	**0.88**	31.26	**26.53**	4.46	**4.74**
ba	**150.92**	184.34	**1.40**	1.95	47.48	**36.61**	**3.01**	2.90
ca	141.39	**110.42**	0.90	**0.58**	28.42	**24.75**	4.98	**5.04**
da	110.16	**110.01**	0.78	1.11	23.26	**21.14**	5.32	**5.58**
ḍa	195.93	**154.23**	1.41	**1.01**	**20.25**	20.36	5.87	**6.13**
ga	167.88	**113.60**	2.05	**1.33**	31.32	**29.58**	3.82	3.76
ha	113.63	**84.13**	0.82	**0.56**	34.09	**29.29**	4.28	**4.62**
ja	**205.21**	220.20	**1.68**	2.05	43.76	**31.06**	**3.12**	2.89
jha	196.70	**183.79**	1.30	**1.29**	27.10	**24.90**	4.27	4.20
ka	105.06	**101.37**	**0.94**	1.18	26.39	**25.03**	4.77	**4.83**
la	153.42	**99.08**	1.43	**0.79**	30.61	**28.36**	4.04	**4.35**
ma	132.18	**114.64**	**1.22**	1.35	40.49	**37.05**	3.08	3.03
na	**149.31**	164.05	**1.39**	1.73	26.64	**25.17**	4.79	**5.07**
ṇa	**142.44**	226.32	**1.09**	2.96	37.06	**35.21**	2.99	**3.11**
pa	116.79	**105.81**	1.17	**1.08**	27.05	**23.14**	4.95	4.66
ra	161.48	**92.41**	1.52	**0.64**	17.29	**16.74**	7.09	7.09
sa	186.72	**96.81**	1.92	**0.85**	32.55	**32.19**	3.14	**3.28**
śa	212.51	**147.37**	2.59	**1.64**	31.20	**27.86**	3.82	**3.92**
ta	219.84	**103.93**	2.57	**1.04**	24.67	**24.23**	4.65	**4.91**
ṭa	255.83	**185.45**	3.27	**2.03**	37.34	**31.92**	3.53	**3.95**
u	146.59	**123.96**	0.93	**0.82**	23.24	**20.86**	5.67	**5.92**
va	141.32	**134.38**	1.13	1.54	28.05	**24.47**	5.61	5.61
ya	165.95	**139.00**	1.65	**1.25**	33.70	**31.01**	3.61	**4.00**

The experimental results demonstrate that, for the majority of classes, the proposed BrahmiGAN outperforms the baseline GAN in terms of the feature-based metrics. Furthermore, BrahmiGAN significantly outperforms the baseline GAN by a large margin across many classes in both raster-based metrics. In the next evaluation stage, the geometric and structural accuracy of the generated letters is assessed using vector-based metrics for both letter outlines and corresponding skeletons. A comparison with the baseline GAN is presented in Table 3.

Based on the experimental results, it is evident that the proposed BrahmiGAN generates more realistic letters, significantly outperforming the baseline GAN in terms of Procrustes distance for both letter outlines and skeletons. Additionally, BrahmiGAN demonstrates superior performance for the majority of classes based on Hausdorff distance. A visual comparison of letters accepted and rejected by human experts for both the proposed and baseline GANs is presented in Figure 7.

Based on the generated letter images, it is evident that BrahmiGAN produces more natural-looking characters with smoother curves, whereas the baseline GAN often generates images with broken strokes. The acceptance rates of human experts for the proposed BrahmiGAN and the baseline GAN are summarized in Table 4.

Based on the acceptance rate of human experts, it is evident that letters generated by BrahmiGAN are accepted by archaeologists at a significantly higher rate compared to those generated by the baseline GAN. In the next stage of evaluation, the letter recognition accuracy of the proposed ensemble Transformer classifier is assessed using the test set comprising real Early Brahmi letters, and the results are summarized in Table 5. The corresponding confusion matrix is presented in Figure 8.

Based on the precision, recall, and F1-score, the proposed ensemble classifier demonstrated excellent performance, achieving an average classification accuracy of 96.06%. The performance of the proposed model is also compared with similar Early Brahmi character recognition studies, as shown in Table 6.

Based on the above comparison, it is evident that the proposed recognition model significantly outperforms similar approaches while demonstrating superior generalizability in terms of the number of images used and the variety of letter classes considered.

Table 3 Comparison of results using vector-based metrics for stroke vectors derived from outlines (HO: Hausdorff outline, PO: Procrustes outline) and skeletons (HS: Hausdorff skeleton, PS: Procrustes skeleton). Metrics are evaluated for letters generated by the baseline GAN (BS) and the proposed BrahmiGAN (BR). Values for these metrics are scaled by 10^{-1}.

| Letter	HO↓		PO↓		HS↓		PS↓	
	BS	BR	BS	BR	BS	BR	BS	BR
a	2.18	**1.94**	4.27	**1.99**	2.42	**2.38**	4.28	**0.23**
ba	**1.81**	1.90	1.63	**1.38**	**1.75**	1.86	1.22	**0.11**
ca	2.26	**2.18**	1.60	**0.30**	**2.32**	2.43	1.26	**0.48**
da	**1.84**	1.89	2.86	**0.54**	2.54	**1.94**	2.74	**0.24**
ḍa	**2.59**	2.67	6.26	**3.79**	2.64	**2.72**	5.83	**0.40**
ga	**1.97**	2.01	2.84	**0.75**	2.20	**1.94**	2.42	**0.64**
ha	2.60	**1.73**	3.24	**0.42**	2.68	**1.61**	2.57	**0.31**
ja	3.03	**2.67**	6.85	**5.03**	3.35	**2.83**	6.66	**0.51**
jha	**2.61**	2.69	6.21	**3.64**	2.70	**2.66**	5.21	**0.37**
ka	**2.42**	2.58	1.80	**0.64**	2.25	**2.49**	1.84	**0.26**
la	2.77	**2.31**	3.98	**1.05**	2.92	**2.38**	3.07	**0.88**
ma	2.32	**1.54**	3.31	**1.29**	2.33	**1.93**	3.91	**0.37**
na	**2.13**	2.15	2.78	**0.84**	**2.13**	2.49	2.26	**0.93**
ṇa	2.19	**1.34**	3.30	**0.52**	**2.38**	2.49	1.22	**0.26**
pa	**2.42**	2.58	1.80	**0.64**	**1.27**	1.29	0.56	**0.02**
ra	1.53	**1.34**	2.64	**0.52**	**3.72**	3.75	4.45	**0.25**
sa	**3.23**	3.33	4.61	**2.08**	2.95	**2.75**	5.67	**0.12**
śa	**2.63**	2.68	2.51	**0.51**	2.69	**2.79**	2.41	**0.58**
ta	2.90	**2.43**	2.96	**1.79**	2.85	**2.54**	3.15	**0.23**
ṭa	1.38	**1.37**	2.12	**0.26**	1.38	**1.39**	1.01	**0.11**
u	1.77	**1.61**	1.43	**0.11**	1.59	**1.59**	0.65	**0.09**
va	3.31	**2.80**	3.61	**1.39**	3.62	**2.94**	3.00	**1.61**
ya	**2.52**	2.58	4.09	**1.30**	2.42	**2.38**	4.28	**0.23**
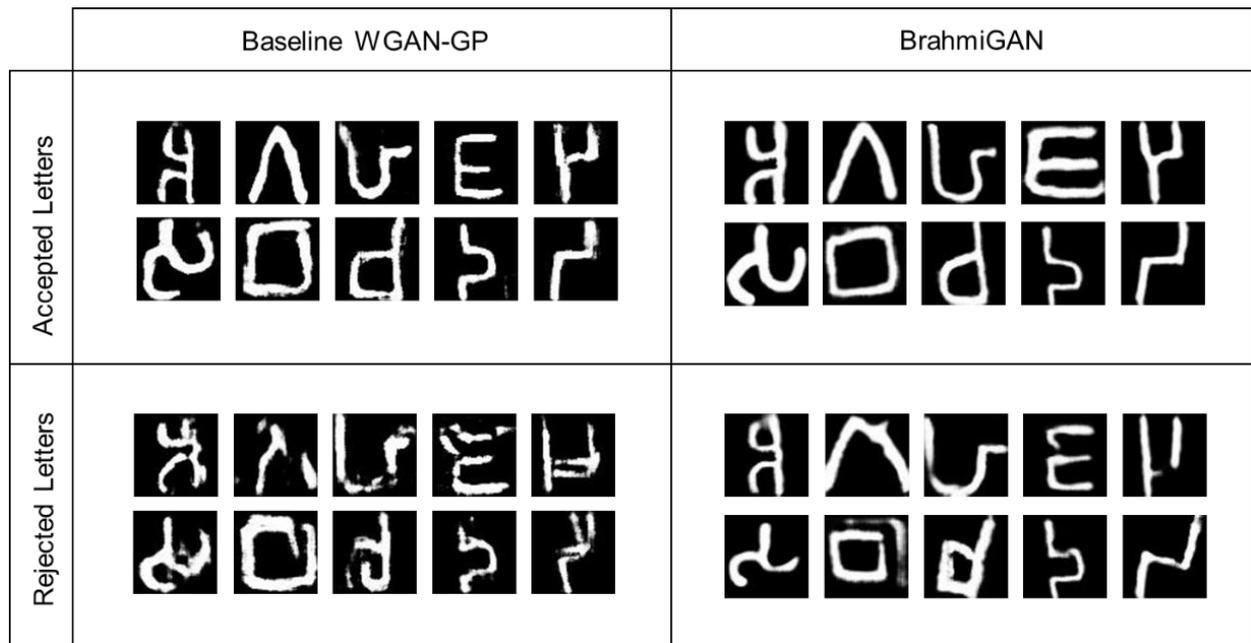
Figure 7 Visual comparison of letters accepted and rejected by human experts for both the proposed BrahmiGAN and the baseline GAN.

Table 4 Comparison of letter acceptability rates evaluated by human experts for letters generated by the baseline GAN (BS) and the proposed BrahmiGAN (BR).

Letter	a	ba	ca	da	ḍa	ga	ha	ja
BS	45.3	56.5	57.3	44.2	53.2	58.5	46.3	40.5
BR	**97.0**	**96.1**	**82.4**	**96.5**	**97.5**	**95.0**	**95.2**	**68.3**
Letter	jha	ka	la	ma	na	ṇa	pa	ra
BS	44.4	50.2	38.7	44.3	46.6	41.4	43.8	61.2
BR	**83.9**	**97.8**	**87.0**	**99.0**	**99.9**	**76.7**	**99.3**	**99.8**
Letter	sa	śa	ta	ṭa	u	va	ya	
BS	54.1	52.9	51.7	59.1	43.3	56.2	41.1	
BR	**82.3**	**93.7**	**97.5**	**95.9**	**98.8**	**93.7**	**86.2**	

Table 5 Classification performance of the proposed ensemble Transformer classifier on real test images.

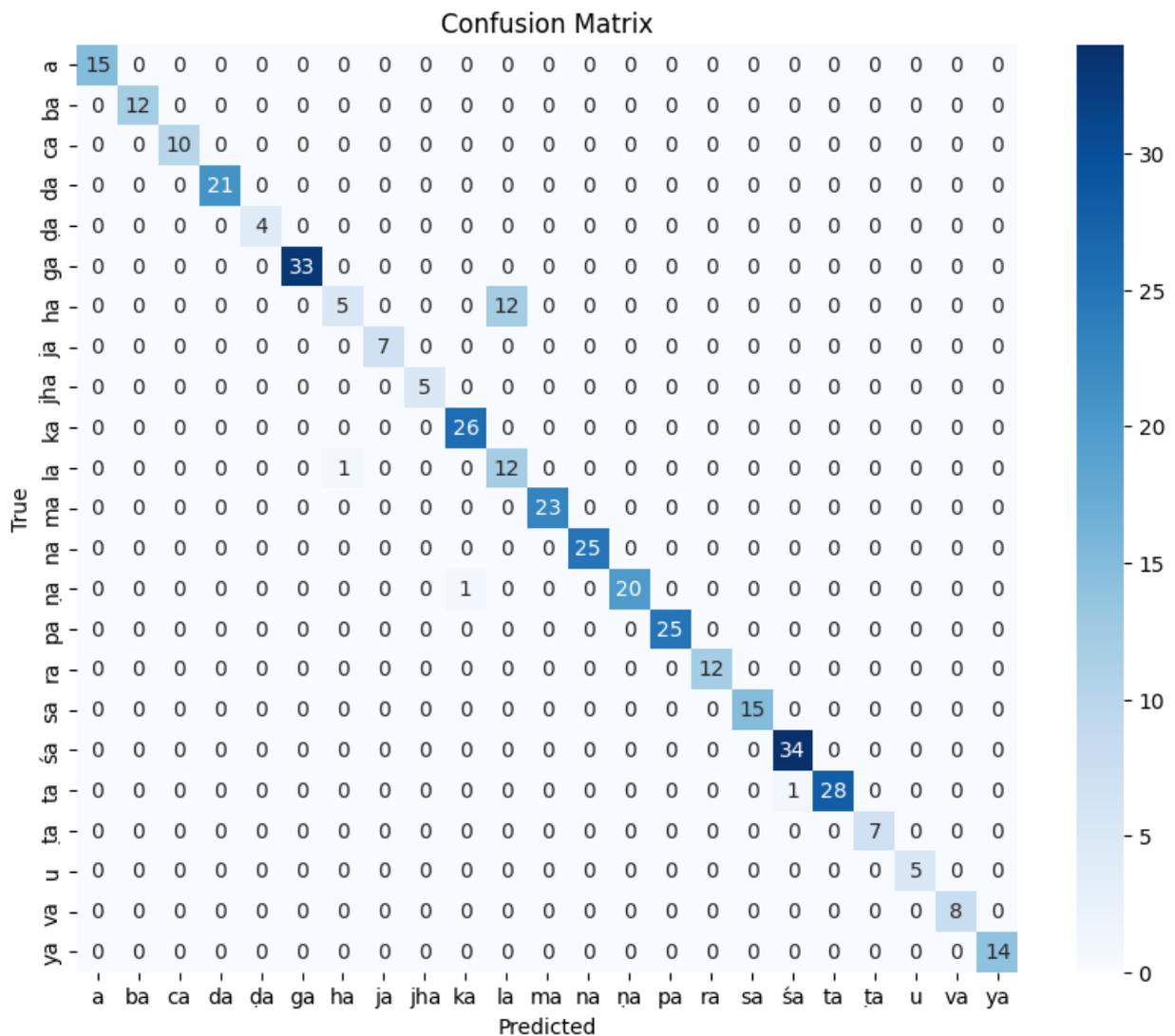| Letter	Precision	Recall	F1-score	Accuracy (%)
a	1.00	1.00	1.00	
ba	1.00	1.00	1.00	
ca	1.00	1.00	1.00	
da	1.00	1.00	1.00	
ḍa	1.00	1.00	1.00	
ga	1.00	1.00	1.00	
ha	0.83	0.29	0.43	
ja	1.00	1.00	1.00	
jha	1.00	1.00	1.00	
ka	0.96	1.00	0.98	
la	0.50	0.92	0.65	
ma	1.00	1.00	1.00	96.06
na	1.00	0.95	0.98	
ṇa	1.00	1.00	1.00	
pa	1.00	1.00	1.00	
ra	1.00	1.00	1.00	
sa	0.97	1.00	0.99	
śa	1.00	0.97	0.98	
ta	1.00	1.00	1.00	
ṭa	1.00	1.00	1.00	
u	1.00	1.00	1.00	
va	1.00	1.00	1.00	
ya	1.00	1.00	1.00	

Figure 8 Confusion matrix of the proposed ensemble Transformer classifier on real test images.

Table 6 Comparison with similar Early Brahmi character recognition studies. N/G denotes Not Given.

Study	Number of Classes	Number of Images Used	Quantitative Outcomes
Gunasekara et al. [6]	15	300	N/G
Wickramarathna and Ranathunga [7]	N/A	100	Word detection, error correction and translation accuracies: Around 86% to 98%
Wijerathna et al. [8]	10	1,800	Period Prediction and Character Recognition Accuracies: 60.00% and 93.33%
Vincen and Samsuryadi [22]	170	8,500	Classification Accuracy: 96.00%
Agrawal et al. [23]	214	3,500	Classification Accuracy: 95.94%
Gautam et al. [24]	170	6,475	Word Recognition Accuracy: 92.47%
Our Study	23	22,464	Letter Classification Accuracy: 96.06%

4.5 Paleographic validation and historical fidelity

In addition to the evaluation based on computational metrics, the paleographic authenticity of the generated Early Brahmi letters is evaluated and presented in this section.

The Early Brahmi script found in Sri Lankan cave inscriptions exhibits distinctive morphological characteristics influenced by the physical act of carving letters into rock surfaces with metal chisels during the 3rd century BCE to 1st century CE [36]. As shown in Figure 2, several consistent features can be observed in real Early Brahmi letter images. The letterforms are predominantly angular rather than curved, with sharp corners and straight strokes. The strokes maintain a uniform thickness throughout each letter, and their terminations are abrupt rather than tapered. Moreover, most characters display a strong vertical orientation with limited horizontal extension.

These consistent features of real Early Brahmi letters are also evident in the images generated by the proposed BrahmiGAN. As shown in Figure 7, the letters accepted by archaeology research officers closely resemble the real inscriptions. The generated letters exhibit unbroken strokes that preserve the structural integrity characteristic of carved writings. Furthermore, the angular features remain crisp and well-defined, the stroke thickness is uniformly maintained throughout each character, and the terminal points display the distinctive abrupt endings typical of chisel-carved scripts.

In addition, the visual comparison between the letters accepted from BrahmiGAN and those generated by the baseline GAN model, as illustrated in Figure 7, underscores the importance of preserving paleographic authenticity. Although some baseline GAN images were accepted by experts, they exhibited softened angles, variable stroke widths, and less crisp terminals, resulting in reconstructions that, while recognizable, were paleographically less accurate.

This paleographic validation has important implications for epigraphic research and the preservation of cultural heritage. The high expert acceptance rate (92.15%) demonstrates that the synthetically generated letters adhere to authentic Early Brahmi morphological principles, indicating that GAN-generated training data can effectively support recognition systems without introducing paleographically invalid patterns.

4.6 Ablation studies

We conducted an ablation study to validate the design of the proposed GAN model and demonstrate the effectiveness of the ensemble-based classifier. The BrahmiGAN model integrates residual connections, a dual attention module, and multi-scale feature concatenation in the generator, while the discriminator incorporates SqueezeNet-based blocks [10], dual attention, and residual dense connections. To justify the design choices of the generator and discriminator, we conducted an ablation experiment on a specific letter class, with the results presented in Table 7.

Table 7 Comparison of the proposed generator and discriminator for the letter class 'ta' across all metrics. Values for KID, SSIM, and all vector-based metrics (HO, PO, HS, PS) are scaled by 10^{-1}. Abbreviations: HO (Hausdorff outline), PO (Procrustes outline), HS (Hausdorff skeleton), PS (Procrustes skeleton), LA (Letter acceptability).

Model	FID↓	KID↓	MSE↓	SSIM↑	HO↓	PO↓	HS↓	PS↓	LA↑
Baseline GAN	219.8	2.57	24.67	4.65	2.90	2.96	2.85	3.15	51.7
Baseline Generator + Proposed Discriminator	118.3	1.06	25.17	5.07	2.54	1.81	2.68	1.24	63.2
Proposed Generator + Baseline Discriminator	198.7	2.29	25.21	5.00	2.58	1.96	2.69	1.37	58.8
BrahmiGAN	103.9	1.04	24.23	4.91	2.43	1.79	2.54	0.23	97.5

Based on the ablation study, it is evident that the proposed generator and discriminator jointly contribute significantly to the performance of the proposed BrahmiGAN. The architectural choices in BrahmiGAN are specifically tailored to address the unique morphological characteristics of Early Brahmi script. To systematically evaluate the contribution of each component in BrahmiGAN, we conducted an ablation study by progressively adding components to the baseline GAN model. Table 8 presents the average performance metrics across all 23 Early Brahmi letter classes, calculated by taking the average of individual letter results for each metric. The baseline configuration represents a standard WGAN-GP without architectural enhancements, while subsequent rows show cumulative additions: channel-spatial attention mechanisms, residual connections, and multi-scale feature concatenation.

The ablation results demonstrate substantial improvements across all metrics when comparing the fully configured BrahmiGAN architecture with the baseline model based on feature-based, raster-based, and vector-based metrics. These improvements, particularly the significant gains in Procrustes distance metrics, which measure overall shape similarity, confirm that our architectural design works synergistically to generate geometrically precise and epigraphically faithful Early Brahmi letters.

Table 8 Comparison of the proposed BrahmiGAN with architectural components (Average performance across all letters, and all quantitative metrics). Values for KID, SSIM, and all vector-based metrics (HO, PO, HS, PS) are scaled by 10^{-1}. Abbreviations: HO (Hausdorff outline), PO (Procrustes outline), HS (Hausdorff skeleton), PS (Procrustes skeleton).

Model	FID↓	KID↓	MSE↓	SSIM↑	HO↓	PO↓	HS↓	PS↓
Baseline GAN	161.06	1.49	30.57	4.39	2.36	3.34	2.48	3.03
+ Channel-Spatial Attention	153.54	1.43	30.23	4.42	2.32	2.95	2.46	2.19
+ Residual Connections	143.21	1.38	28.90	4.44	2.25	2.18	2.39	1.31
+ Multi-scale Features (BrahmiGAN)	134.68	1.29	27.28	4.50	2.19	1.33	2.33	0.40

To validate the design choice of the proposed ensemble classification model, an additional ablation study was conducted. The individual classification performance of the PVT and Swin Transformers was evaluated across various training sets. Their performance was then measured using a test set comprising exclusively real images. The experimental results are summarized in Table 9.

Table 9 Comparison of average classification accuracy (%) for individual models across various training sets.

Training Images	PVT		Swin	
	Raw	Skeletonized	Raw	Skeletonized
Real	82.15	86.87	81.62	81.89
Generated	89.23	91.60	87.66	95.54
Real + Generated	**93.70**	92.38	91.86	**95.80**

Based on the ablation study results, it is evident that incorporating generated images into the training set significantly improves the classification performance of both Transformer-based models. Furthermore, it can be observed that the PVT model achieves higher accuracy for raw images compared to skeletonized images, whereas the Swin Transformer demonstrates superior performance for skeletonized images across all three types of training sets. The ablation experiment further confirms that ensembling the classification scores of individual models improves accuracy, achieving 96.06%, compared to the best individual performances of the PVT and Swin Transformers, which reached 93.70% and 95.80%, respectively.

4.7 Discussion and future directions

This study represents a significant advancement in Sri Lankan epigraphical research by addressing the persistent challenge of data scarcity in the analysis of ancient scripts and the recognition of challenging Early Brahmi letters. The proposed BrahmiGAN generates realistic and culturally accurate Early Brahmi letters by incorporating residual connections, attention mechanisms, and multi-level and multi-scale features. The high acceptance rate by human experts (92.15% overall) clearly indicates that the generated Early Brahmi letters are not only visually realistic but also culturally authentic. This underscores the practical usability of generated letters for training recognition models, educational tools, and cultural preservation efforts.

Although the proposed BrahmiGAN demonstrated outstanding performance, the occasional outperformance of the baseline GAN on feature-based metrics suggests that further refinement may be needed to achieve better alignment between high-level feature distributions and real data. Additionally, the slightly lower Hausdorff distance observed in certain classes likely results from rotation and augmentation effects, as this metric is particularly sensitive to such transformations. Moreover, while most letter classes achieved an acceptance rate exceeding 80% among human experts, the letters 'ja' and 'ṇa' exhibited slightly lower rates due to their diverse shapes and intricate structural details, which make accurate replication challenging and increase the likelihood of confusion with other letter classes.

The proposed ensemble Transformer-based recognition model demonstrated outstanding performance on the test set containing only real Early Brahmi letters. However, confusion matrix (Figure 8), indicates a significant misclassification between letters 'ha' and 'la'. Letter 'ha' achieves only 29% recall despite 83% precision, with 12 out of 17 test instances misclassified as 'la'. Conversely, 'la' maintains 92% recall but only 50% precision. As shown in Figure 9, these letters are nearly mirror images of each other along the horizontal axis and share identical structural skeletons, angular features, and stroke patterns, making them difficult to distinguish. In future work, additional training samples will be collected for these letters to overcome this limitation. Furthermore, incorporating contextual information from surrounding letters in inscriptions represents another promising direction for distinguishing visually similar letter classes.

Figure 9 Similarity between letters 'ha' and 'la'.

Although the proposed study generates synthetic images of Early Brahmi letters and significantly enhances recognition accuracy, it is essential to acknowledge the potential risks associated with creating artificial representations of historical artifacts. Synthetic letters may inadvertently introduce subtle stylistic biases or morphological variations not present in authentic inscriptions, which could lead to misinterpretation of paleographic characteristics or an inaccurate understanding of script evolution. To mitigate these risks, this

study implements several precautionary measures, including validation of the generated letters by archaeological research officers, evaluation of both letter generation and recognition performance exclusively on real inscription images to ensure real-world applicability, and assessment of model performance using a comprehensive set of quantitative metrics.

Future work in this study will focus on expanding the dataset by collecting and annotating additional real Early Brahmi letter images to enhance the diversity and quality of the training data. Additionally, exploring multi-modal generation could extend the model's capabilities to generate not only individual letters but also complete words or inscriptions, incorporating contextual information to improve coherence and realism. Finally, generalizing the framework to other ancient scripts will enable the proposed approach to address similar challenges in generating and preserving other low-resource historical scripts, thereby contributing to broader cultural heritage preservation efforts.

5. Conclusion

In this study, the challenge of data scarcity in Sri Lankan Early Brahmi script recognition is addressed through BrahmiGAN, a novel generative model that significantly advances the synthesis of realistic and culturally accurate Early Brahmi letters. By integrating residual connections, attention mechanisms, multi-level, and multi-scale features, the proposed model effectively overcomes critical issues including mode collapse, geometric inaccuracy, and perceptual quality degradation. Through feature-based, raster-based, and vector-based evaluations, BrahmiGAN demonstrates superior performance in letter generation. Notably, the model achieves an overall acceptance rate of 92.15% as evaluated by human experts, a substantial improvement over the baseline GAN's rate of 49.16%, underscoring its effectiveness. Furthermore, unlike most previous ancient character recognition models, this study employs Transformer-based architectures for letter recognition, integrating PVT and Swin Transformers in an ensemble framework. The proposed approach achieves an impressive classification accuracy of 96.06% on a test set composed exclusively of real inscription images. Although this ensemble approach demonstrates outstanding performance, it still faces challenges in distinguishing visually similar letter classes, a limitation that will be addressed in future work by incorporating contextual information from surrounding letters. This research provides a solid foundation for computationally assisted epigraphical studies, integrating traditional archaeological methods with modern machine learning techniques to preserve and enhance the understanding of Sri Lanka's ancient textual heritage.

6. Acknowledgment

7. References

[1] Vadineanu S, Kalayci T, Pelt DM, Batenburg KJ. Convolutional neural networks and their activations: an exploratory case study on mounded settlements. J Comput Appl Archaeol. 2024;7(1):262-82.

[2] Sobotkova A, Kristensen-McLachlan RD, Mallon O, Ross SA. Validating predictions of burial mounds with field data: the promise and reality of machine learning. J Doc. 2024;80(5):1167-89.

[3] Aioanei AC, Hunziker-Rodewald RR, Klein KM, Michels DL. Deep Aramaic: towards a synthetic data paradigm enabling machine learning in epigraphy. PLoS ONE. 2024;19(4):e0299297.

[4] Violatti C, Dharma P. Brahmi script [Internet]. 2016 [cited 2024 Dec 10]. Available from: https://www.worldhistory.org/Brahmi_Script/

[5] Bandara D, Warnajith N, Minato A, Ozawa S. Creation of precise alphabet fonts of early Brahmi script from photographic data of ancient Sri Lankan inscriptions. Can J Artif Intell Mach Learn Pattern Recognit. 2012;3(3):33-9.

[6] Gunasekara S, Lafir MH, Dulaj C, Haputhanthri L, Alwis D. Deep learning-powered mobile app for early Brahmi Script decipherment in Sri Lanka. 2024 International Research Conference on Smart Computing and Systems Engineering (SCSE); 2024 Apr 4; Colombo, Sri Lanka. USA: IEEE; 2024. p. 1-6.

[7] Wickramarathna S, Ranathunga L. Data driven approach to Brahmi OCR error correction and Sinhala meaning generation from Brahmi character array. 2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer); 2019 Sep 2-5; Colombo, Sri Lanka. USA: IEEE; 2019. p. 1-6.

[8] Wijerathna KASAN, Sepalitha R, Indika T, Athauda H, Suranjini PD, Silva JADC, et al. Recognition and translation of ancient Brahmi letters using deep learning and NLP. 2019 International Conference on Advancements in Computing (ICAC); 2019 Dec 5-7; Malabe, Sri Lanka. USA: IEEE; 2019. p. 226-31.

[9] Surasinghe P, Thanikasalam K. An automated period prediction system for Sinhala epigraphical scripts using ensemble CNNs and attention modules. ECTI Trans Comput Inf Technol. 2024;18(4):555-67.

[10] Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size [Internet]. arXiv [Preprint]. 2016 [cited 2024 Dec 10]. Available from: https://doi.org/10.48550/arXiv.1602.07360.

[11] Wang W, Xie E, Li X, Fan DP, Song K, Liang D, et al. Pyramid vision transformer: a versatile backbone for dense prediction without Convolutions. International conference on computer vision, ICCV 2021; 2021 Oct 11-17; USA: IEEE; 2021. p. 568-78.

[12] Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, et al. Swin transformer: hierarchical vision transformer using shifted windows. International conference on computer vision, ICCV 2021; 2021 Oct 11-17; USA: IEEE; 2021. p. 10012-22.

[13] Yuan Z, Kamata SI. Data augmentation for ancient characters via Semi-MixFontGan. 2020 Joint 9th International Conference on Informatics, Electronics & Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision & Pattern Recognition (icIVPR); 2020 Aug 26-29; Kitakyushu, Japan. USA: IEEE; 2020. p. 1-6.

[14] Vögtlin L, Drazyk M, Pondenkandath V, Alberti M, Ingold R. Generating synthetic handwritten historical documents with OCR constrained GANs. 16th International Conference on Document Analysis and Recognition, ICDAR 2021; 2021 Sep 5-10; Lausanne, Switzerland. Cham: Springer; 2021. p. 610-25.

[15] Perrin S, Cudilla L, Xie Y, Mouchère H, Marthot-Santaniello I. Homer restored: virtual reconstruction of Papyrus Bodmer 1. HIP '23: 7th International Workshop on Historical Document Imaging and Processing; 2023 Aug 1; San Jose, USA. USA: ACM; 2023. p. 37-42.

[16] Vidal-Gorène C, Camps JB, Clérice T. Synthetic Lines from historical manuscripts: an experiment using GAN and style transfer. Image Analysis and Processing, ICIAP 2023 Workshops; 2023 Sep 11-15; Udine, Italy. Cham: Springer; 2023. p.477-88.

[17] Fogel S, Averbuch-Elor H, Cohen S, Mazor S, Litman R. ScrabbleGAN: semi-supervised varying length handwritten text generation. CVF conference on computer vision and pattern recognition, CVPR 2020; 2020 Jun 14-19; USA: IEEE; 2020. p. 4324-33.

[18] Zhu JY, Park T, Isola P, Efros AA. Unpaired image-to-image translation using cycle-consistent adversarial networks. International conference on computer vision, ICCV 2017; 2017 Oct 22-29; Venice, Italy. USA: IEEE; 2017. p. 2223-32.

[19] Preethi P, Mamatha HR. Intelligent recognition of ancient Brahmi characters using transfer learning. 2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT); 2023 Jan 5-6; Bhilai, India. USA: IEEE; 2023. p. 1-6.

[20] Devi KD, Supraja G, Vanathi A, Vidhyalakshmi M. Digitization and electronic translation of Brahmi stone inscriptions. AIP Conf Proc. 2023;2917(1):050017.

[21] Nagane AS, Patil CH, Mali SM. Classification of Brahmi script characters using HOG features and multiclass error-correcting output codes (ECOC) model containing SVM binary learners. 2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE); 2023 Jan 27-28; Bengaluru, India. USA: IEEE; 2023. p. 448-51.

[22] Vincen V, Samsuryadi S. Brahmi script classification using VGG16 architecture convolutional neural network. Comput Eng Appl J. 2022;11(2):127-36.

[23] Agrawal Y, Balasubramanian S, Meena R, Alam R, Malviya H, Rohini P. Optical character recognition using convolutional neural networks for Ashokan Brahmi inscriptions [Internet]. arXiv [Preprint]. 2024 [cited 2024 Dec 10]. Available from: https://doi.org/10.48550/arXiv.2501.01981.

[24] Gautam N, Chai SS, Jose J. Recognition of Brahmi words by using deep convolutional neural network [Internet]. 2020 [cited 2025 Sep 30]. Available from: https://www.preprints.org/manuscript/202005.0455/v1.

[25] Bhuvaneswari S, Kathiravan K. Enhancing epigraphy: a deep learning approach to recognize and analyze Tamil ancient inscriptions. Neural Comput Appl. 2024;36:19839-61.

[26] Murugan B, Visalakshi P. Ancient Tamil inscription recognition using detect, recognize and labelling, interpreter framework of text method. Heritage Sci. 2024;12:430.

[27] Lalitha E, Mondal A, Jawahar. Enhancing accuracy in Indic handwritten text recognition. 9th International Conference on Computer Vision and Image Processing, CVIP 2024; 2024 Dec 19-21; Hyderabad, India. Cham: Springer; 2026. p. 234-48.

[28] Mostafa MT, Rhythm ER, Mehedi MHK, Rasel AA. Advancements in optical character recognition for Bangla scripts. 2023 Innovations in Intelligent Systems and Applications Conference (ASYU); 2023 Oct 11-13; Sivas, Turkiye. USA: IEEE; 2023. p. 1-5.

[29] Cheema MDA, Shaiq MD, Mirza F, Kamal A, Naeem MA. Adapting multilingual vision language transformers for low-resource Urdu optical character recognition (OCR). PeerJ Comput Sci. 2024;10:e1964.

[30] Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville A. Improved training of Wasserstein GANs. 31st International Conference on Neural Information Processing Systems; 2017 Dec 4-9; Long Beach, California. USA: NeurIPS; 2017. p. 5767-77.

[31] Optuna. A hyperparameter optimization framework [Internet]. 2025 [cited 2025 Sep 30]. Available from: https://optuna.org/.

[32] Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. 31st International Conference on Neural Information Processing Systems; 2017 Dec 4-9; Long Beach, California. USA: NeurIPS; 2017. p. 6627-37.

[33] Bińkowski M, Sutherland DJ, Arbel M, Gretton A. Demystifying MMD GANs. 6th International Conference on Learning Representations, ICLR 2018; 2018 Apr 30 - May 3; Vancouver, Canada. USA: ICLR; 2018. p. 1-36.

[34] Huttenlocher DP, Klanderman GA, Rucklidge WJ. Comparing images using the Hausdorff distance. IEEE Trans Pattern Anal Mach Intel. 1993;15(9):850-63.

[35] Andreella A, De Santis R, Vesely A, Finos L. Procrustes-based distances for exploring between-matrices similarity. Stat Meth Appl. 2023;32(3):867-82.

[36] Paranavitana S. Inscriptions of Ceylon. Vol. 1. Colombo: Department of Archaeology; 1970.