Exploring cGANs for Urdu Alphabets and Numerical System Generation

Suleman Khalil[1], Syed Yasser Arafat[1], Fatima Bibi[1], Faiza Shafique[1]
[1]Department of CS & IT (Mirpur University of Science and Technology (MUST), Mirpur (AJK), Pakistan)
***Correspondence:** sulemankhalil12@gmail.com.

Citation | Khalil. S, Arafat. S. Y, Bibi. F, Shafique. F, "Exploring cGANs For Urdu Alphabets and Numerical System Generation", IJIST, Special Issue. pp 164-187, March 2025
DOI | https://doi.org/10.33411/IJIST/1226
Received | Feb 20, 2025 **Revised |** March 04, 2025 **Accepted |** March 10, 2025 **Published |** March 13, 2025.

Urdu ligatures play a crucial role in text representation and processing, especially in Urdu language applications. While extensive research has been conducted on handwritten characters in various languages, there is still a significant gap in studying raster-based generated images of Urdu characters. This paper presents a generative model designed to produce high-quality samples that closely resemble yet differ from existing datasets. Utilizing the power of Generative Adversarial Networks (GANs), the model is trained on a diverse dataset comprising 40 classes of Urdu alphabets and 20 classes of numerals (both modern and Arabic-style), with each class containing 1,000 augmented images to capture variations. The generator network creates synthetic Urdu character samples based on class conditions, while the discriminator network evaluates their similarity to real datasets. The model's effectiveness is assessed using key metrics such as the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), and Fréchet Inception Distance (FID).

The results confirm that the proposed GAN-based approach achieves high fidelity and structural accuracy, making it highly valuable for applications in text digitization and Optical Character Recognition (OCR).

Keywords: Generative Adversarial Network, Structural Similarity Index, Fréchet Inception Distance, Peak Signal-to-Noise Ratio, Optical Character Recognition.

Introduction:

The increasing reliance on digital text representations in document archiving, optical character recognition (OCR), and font design highlights the need for efficient methods to generate digital versions of various writing systems. Generative Adversarial Networks (GANs) have numerous practical applications, such as enhancing OCR system performance by training on diverse datasets to improve recognition accuracy. Additionally, GANs assist typographers in creating new fonts by generating diverse ligatures and numeral designs. Research has explored techniques like GANs for generating handwritten text in various languages, including Arabic [1], Bangla [2], Chinese [3], Nepali [4], and Urdu ligatures [5]. These studies primarily used handwritten datasets for model training. While models like Stable Diffusion [6][7] and DALL-E [8] are designed for general purposes, this study aims to bridge a gap by investigating the potential of using Conditional GANs (cGANs) to generate raster images of characters from Urdu, Arabic, and modern numerals.

Urdu belongs to the Indo-Aryan subgroup of the Indo-European language family. Approximately 70 million people speak Urdu as their mother tongue, while around 100 million others, primarily in Pakistan and India, use it as a second language [9]. It is recognized in India's constitution and serves as Pakistan's official language [9]. Significant Urdu-speaking communities exist in the United Arab Emirates, the United States, and the United Kingdom. Notably, Hindi and Urdu are mutually intelligible [10]. The Urdu script consists of 60 characters, derived from 28 Arabic letters and 32 Persian characters, written in Naskh and Nastaliq styles [11]. These two fonts are widely used in different languages: Nastaliq is primarily employed for Urdu, Punjabi, and Sindhi, while Naskh is used for Arabic, Persian, and Pashto [12]. Nastaliq follows an elegant, cursive, right-to-left writing style, featuring ligatures formed by both joining and non-joining alphabets [11][13][14].

Arabic numerals originate from the Hindu-Arabic numeral system, which includes both isolated (non-joining) and connected (joining) alphabets, reflecting its cursive nature [11][13][14]. This system, which originated in India and was later adopted by Arabic mathematicians [15], is often mistakenly considered "Western" or "Latin" digits. The numerals (0-9) are widely used worldwide, including in Urdu and Arabic scripts. Ensuring accuracy and consistency in generating these numeral shapes is crucial for OCR systems recognizing numbers within Urdu text. In this research, "modern numbers" may refer to specific numeral glyphs within certain scripts, such as the extended Arabic-Indic digits used in Urdu [16]. This study differs from previous research by using a raster-based dataset instead of handwritten images for cGAN training. Raster images are widely used in photographs, bitmap graphics, and scanned documents due to their ability to depict a broad range of colors and intricate details. We explore the application of Conditional Generative Adversarial Networks (cGANs), a specialized class of generative models [17]. These networks consist of two primary components: the generator (G) and the discriminator (D). The generator (G) creates synthetic images that mimic real data, while the discriminator (D) evaluates the images to distinguish between real and generated samples.

The primary goal of this research is to develop a system capable of generating realistic Urdu alphabets and numerals to enhance existing datasets. Figure 1(a) presents research papers on GANs that we reviewed. HiGAN+ [18] introduces a framework that separates latent space into style and content factors, allowing independent control over handwriting form while preserving its core content. StackGAN++ [19] generates high-resolution images from text descriptions. VQGAN [20] integrates an autoencoder with a GAN architecture, using vector quantization to capture intricate image details. Mirrorgan [21] introduces a novel text-to-image generation technique based on redescription. TiGAN [22] provides an innovative framework for interactive text-driven image creation and modification. AttnGAN [23] focuses on generating images from text descriptions, while JoinFontGAN [24] employs few-shot learning

to produce high-quality fonts. TeDiGAN [25] uses text descriptions to create and manipulate face images with high realism and control.

Figure 1(a). A recent study of GANS.

Novelty:

The key contributions of our research are as follows:

- We created a novel dataset for training Generative Adversarial Networks (GANs) to address a significant gap in resources for Urdu script generation. This dataset can further aid in developing robust Urdu Optical Character Recognition (OCR) systems.
- We successfully demonstrated the generation of raster images for Urdu characters, Arabic numerals, and modernized numerals using a Conditional Generative Adversarial Network (GCN). To the best of our knowledge, this is the first instance of cGAN being applied to Urdu characters.
- We introduced an innovative approach to data augmentation by generating synthetic raster-based images, potentially transforming dataset creation for various applications.
- We evaluated the effectiveness of a machine learning model trained on synthetic data generated by GCC. The model demonstrated strong performance when tested on real-world data.

Objectives:

Our study aimed to achieve several objectives. First, we sought to address the scarcity of resources for Urdu text, ligatures, and numeral images required for training cGAN models. To bridge this gap, we developed a dataset comprising 40,000 Urdu ligature images and 20,000 numeral images.

Second, our goal was to develop a robust cGAN model capable of generating an unlimited number of realistic Urdu text images. This objective was successfully realized using our custom-built dataset.

Additionally, we assessed the model's robustness using various evaluation metrics to ensure its reliability and effectiveness.

Related Work:

In 2014, Ian Goodfellow and his colleagues introduced the foundational research paper on Generative Adversarial Networks (GANs) [26]. This paper presents a framework where two neural networks—a generative model (G) and a discriminative model (D)—are trained together in an adversarial process. The objective of G is to model the data distribution, while D evaluates the probability that a given sample comes from the training data rather than

being generated by G. This minimax game framework enables effective training through backpropagation and allows the generation of high-quality samples without requiring Markov chains. This approach leverages adversarial networks to generate competitive samples, offering potential solutions to challenges in generative modeling.

One of the notable studies on GAN-based handwritten English text generation was conducted by Eloi Alonso et al. [27]. They proposed an adversarial approach for generating handwritten word images using a bidirectional long short-term memory (LSTM) recurrent neural network to extract the words for generation. These extracted words were then fed as conditional input, along with noise, into the generator networks. An additional recognizer was also incorporated into the network. While the numerical results were promising, the generated images of French and Arabic words were significantly blurry. Wu et al. [28] employed the DCGAN architecture to generate images of tomato leaf disease, thereby augmenting the dataset of diseased leaves. This method, known as dataset augmentation using DCGAN, provided an alternative to traditional augmentation techniques such as translation, rotation, and flipping, which do not always generalize well. Their DCGAN-based augmentation approach improved model recognition accuracy compared to conventional techniques. Furthermore, their findings indicated that the DCGAN-generated results were more convincing in both the visual Turing Test and t-distributed Stochastic Neighbor Embedding analysis.

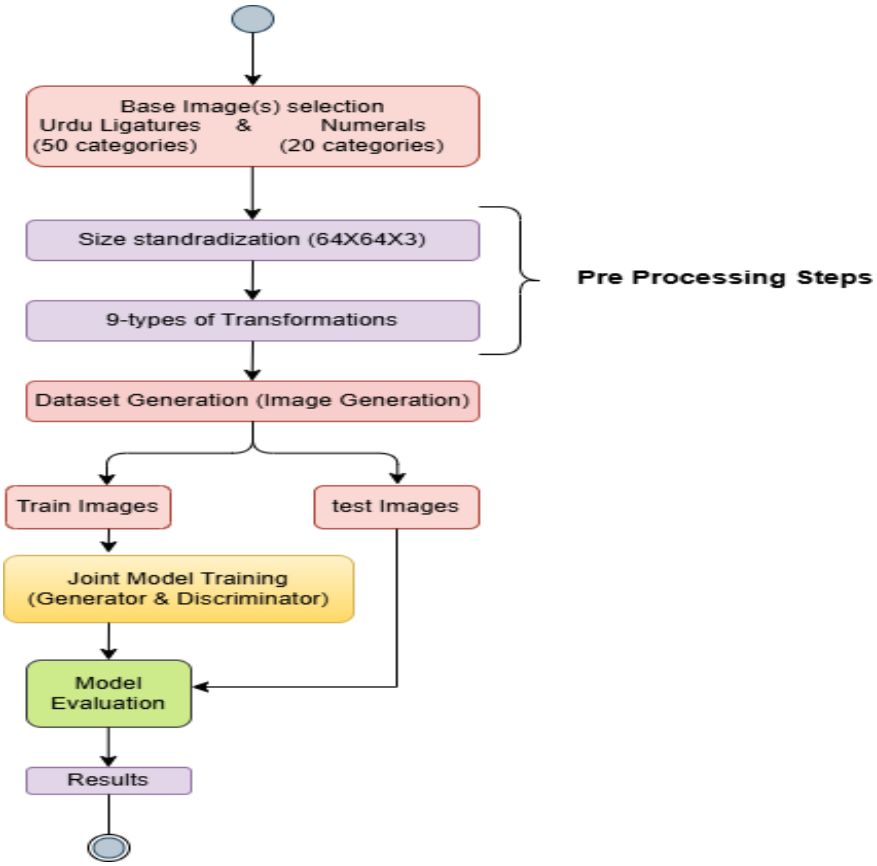

Figure 1(b). The workflow of this research study

In [1], Mustapha et al. used the DCGAN architecture to generate handwritten Bangla digits. They utilized three widely recognized handwritten Bangla datasets—Ekush, their dataset, BanglaLekha-Isolated, and CMATERdb—to achieve their objective. Since the proposed DCGAN efficiently generates Bangla digits, it serves as a reliable model for producing handwritten Bangla digits from random noise. Their study aims to apply the DCGAN architecture to generate Arabic characters. The dataset comprises handwritten

images of 33 Arabic alphabets, from "alef" to "yeh," with each character having 480 images, each sized 32x32 pixels. Recent advancements in generative models for text and image synthesis have significantly enhanced the ability to generate high-quality synthetic data for low-resource languages. While GANs have been widely used for handwritten character generation and font synthesis, newer approaches, such as transformer-based diffusion models, have demonstrated superior results in text-to-image generation. Studies by Ramesh et al. [8] on DALL-E and Rombach et al. [6] on Stable Diffusion indicate that diffusion models trained on large-scale datasets outperform GANs in generating visually coherent text-based images. However, these models require substantial computational resources and large-scale pretraining, making them impractical for low-resource languages like Urdu. In contrast, conditional GANs (cGANs) provide a computationally efficient alternative, enabling fine-grained control over script-specific character generation while maintaining high visual quality.

Urdu Optical Character Recognition (OCR) and script synthesis present unique challenges due to complex ligature formations, cursive structure, and varying diacritic placements. Unlike Latin-based scripts, where characters remain distinct, Urdu letters change shape depending on their position (isolated, initial, medial, or final), making OCR-based training datasets highly diverse and difficult to standardize. Prior research by Arafat & Iqbal [29][30] highlighted that traditional machine learning and neural network-based OCR models struggle with segmentation errors due to overlapping ligatures. Recent studies have explored GAN-based data augmentation to enhance Urdu OCR performance; however, most existing approaches rely on basic GAN architectures (e.g., DCGAN, Pix2Pix) rather than more sophisticated conditional architectures like cGANs. This study addresses this gap by leveraging cGANs for Urdu character and numeral generation, providing a more adaptable framework for low-resource script synthesis and OCR model training.

Despite the success of GANs in generating realistic images, their application in low-resource language processing remains underexplored, particularly in text-based image generation. Previous studies have successfully employed CycleGAN for domain adaptation [22]. Additionally, research by Guan et al. [31] on GAN-based data augmentation for handwritten datasets underscores the potential of generative models in enhancing dataset diversity for OCR tasks. By implementing cGANs with controlled conditioning on Urdu characters and numerals, this study introduces a scalable and effective generative framework for script-based OCR improvement, filling a crucial research gap in Urdu script synthesis.

Dataset:

The dataset used in this study is based on the Urdu Ligatures dataset [29], originally comprising 45,000 unique ligatures. These ligatures included nine distinct transformation types, along with a standard set of ligatures. To enhance the dataset for this study, the following steps were taken:

- **Resolution Standardization:** Using the cv2 library, all images were resampled to a resolution of 64x64 pixels, ensuring consistency across the dataset.
- **Data Augmentation:** Based on techniques outlined in [32], ten different transformation methods were applied to the ligatures. This augmentation process significantly expanded the dataset, generating 1,000 images for each letter and numeral using transformations such as scaling, rotation, and flipping.

As a result, the dataset increased to 60,000 images, comprising 40,000 Urdu characters and 20,000 numerals (both Urdu and Modern Indo-Arabic). The general steps involved in creating the Urdu dataset are shown in Figure 1(b).

This refined and standardized dataset provides a diverse and well-balanced representation of Urdu characters and numerals, making it ideal for machine learning applications such as text recognition and computational linguistics. Furthermore, it addresses

a significant gap in resources for Urdu script generation, laying the foundation for future research in this area.

Developed Methodology:

Model Development:

Conditional Generative Adversarial Networks (cGANs) are a variation of the GAN architecture in which the generator receives an additional conditional input alongside the latent noise. This configuration enables the model to generate data based on specific inputs, making it suitable for tasks such as generating images corresponding to predefined labels or categories. In this research, we employed a cGAN to generate vector-based images of Urdu alphabets, Arabic numerals, and modern digits. The GCC consists of two neural networks: a generator (G) and a discriminator (D), as illustrated in Figure 2. This work is the first to apply cGANs specifically for low-resource languages, such as Urdu, for ligature and numeral generation, highlighting the model's effectiveness in producing diverse and high-quality images.

Generator:

The generator network takes a label and a random array as input, producing images that align with the structure of the training data for that label.

Discriminator:

The discriminator evaluates labeled data batches, combining real data from the training set with generated data, classifying each sample as either "real" or "generated."

Generator Model:

Our generator model takes a latent input and generates an image. Its architecture includes fully connected layers, transposed convolutional layers, batch normalization, and activation functions such as ReLU and Tanh. An input layer is used to define the input layer for latent variables, and a concatenation layer integrates the conditional information with the generated features.

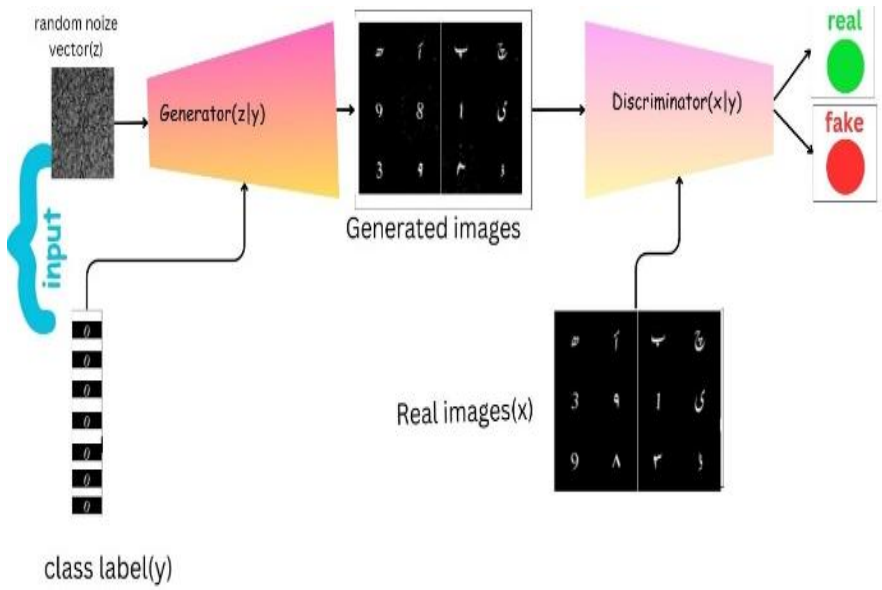

Figure 2. Proposed Methodology

The function layer is a custom layer designed to apply a transformation that converts an appearance into an image. To keep the pixel values of the output image within the range of [-1, 1], a tanh layer is used. Figure 3 illustrates the structure of our generator.

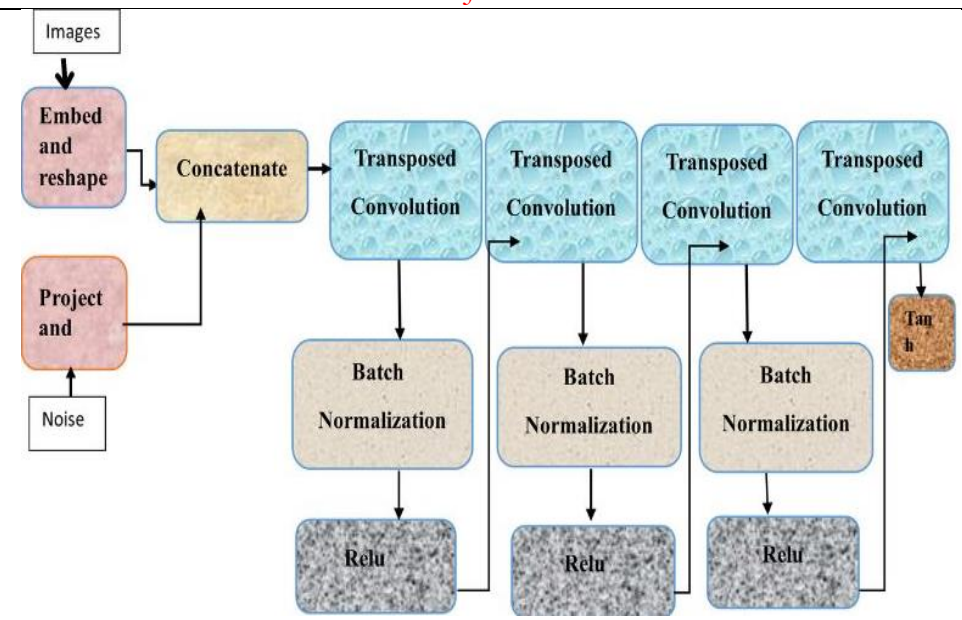

Figure 3. Structure of Generator network.

The generator's loss function is given in Equation 1.

$$\text{lossGenerator} = -\text{mean}(\log(\hat{y}_{\text{Generated}})) \quad (1)$$

The generator's score is determined by the average feasibility assigned by the discriminator to the generated data, as shown in Equation 2.

$$\text{scoreG} = \text{mean}(\text{sigmoid}(\hat{y}_{Generated})) \quad (2)$$

Discriminator Model:

The discriminator model is designed to evaluate the authenticity of an input image, determining whether it is real or generated. Its architecture includes key components such as convolutional layers, batch normalization, dropout, Leaky ReLU activation, and concatenation layers. The Image Input Layer serves as the entry point for image data, while the Dropout Layer helps reduce overfitting by applying regularization. The Leaky ReLU activation function provides a small gradient for negative inputs, aiding efficient learning. Additionally, the Concatenation Layer combines conditional data with image features, improving the model's ability to distinguish between real and generated images. The discriminator's architecture is illustrated in Figure 4.

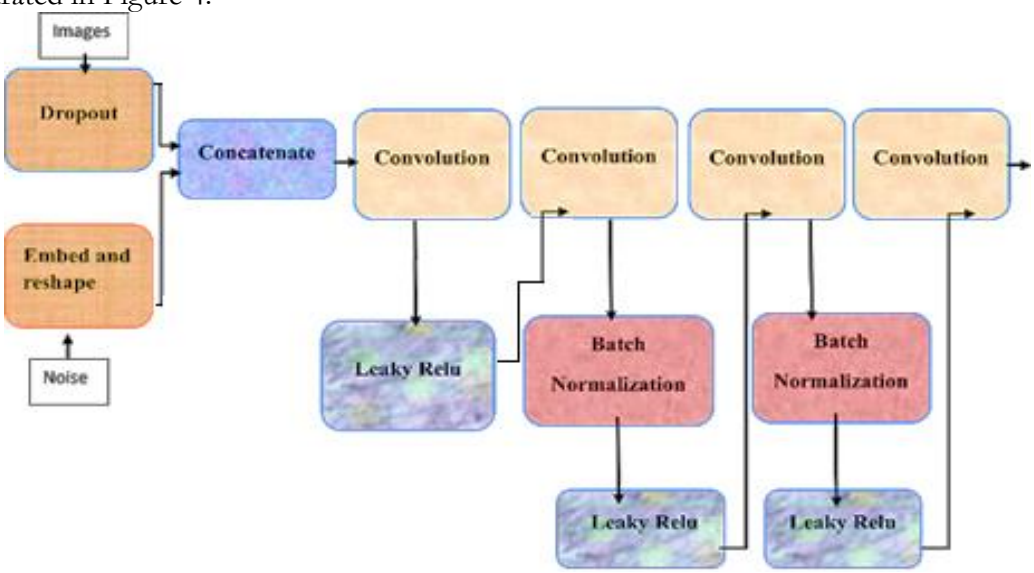

Figure 4. Structure of Discriminator network.

Equation 3 defines the loss function for the discriminator.

$$\text{lossDiscriminator} = -\text{mean}(\log(\hat{y}_{\text{real}})) - \text{mean}(\log(1 - \hat{y}_{\text{Generated}})) \quad 3$$

The discriminator's score is calculated as the average contingency assigned to real and generated data by the discriminator, as shown in Equation 4.

$$\text{scoreD} = 1/2(\text{mean}(\text{sigmoid}(\hat{y}_{real})) + \text{mean}(1 - \text{sigmoid}(\hat{y}_{Generated}))) \quad 4$$

Training Parameters:

With a learning rate of 0.0002, a gradient decay rate of 0.5, and a squared gradient decay rate of 0.999, we used the Adam optimizer to train both the discriminator and generator networks. Adam is widely recognized for its efficient optimization and adaptive learning rate capabilities.

The training process included 45 epochs for the Urdu characters dataset and **81 epochs** for the Urdu numerals dataset, where each epoch represents a complete pass through the entire training data. To balance computational efficiency with the ability to capture diverse gradients from the data, a batch size of 128 samples was used.

The model's performance was evaluated on a validation set every 100 iterations to monitor training progress and reduce the risk of overfitting. Training progress graphs were updated accordingly, and the performance graphs for the generator and discriminator on the Urdu characters and numerals datasets are shown in Figure 5 and Figure 6, respectively.

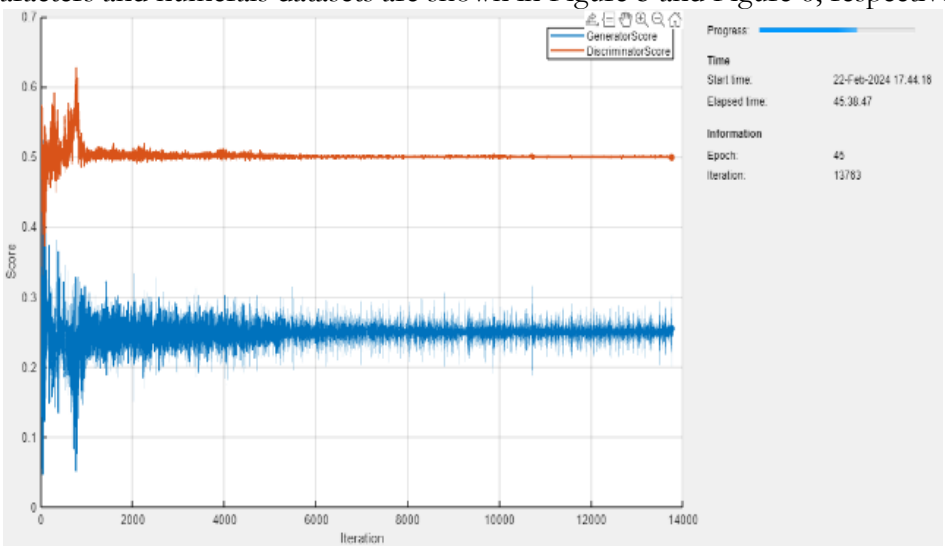

Figure 5. Generator and discriminator score graph for Urdu characters dataset

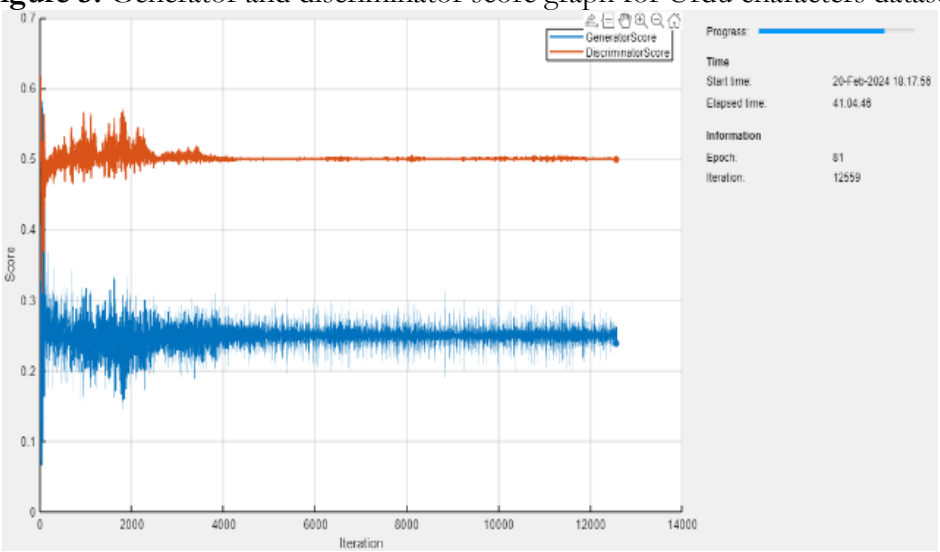

Figure 6. Generator and discriminator score graph for numbers dataset

Evaluation Metrics:

We use PSNR (Peak Signal-to-Noise Ratio) [33], where a higher PSNR value indicates better image quality, and SSIM (Structural Similarity Index) [34], where values closer to 1 represent higher similarity, while values closer to 0 indicate lower quality. A low PSNR score suggests significant numerical differences between images.

Additionally, we evaluate our model using FID (Fréchet Inception Distance) [35]. A perfect FID score of **0.0** means the two sets of images are identical, while lower FID values indicate greater similarity or a closer statistical alignment between them. The formulas for these metrics are provided in Table 1.

Table 1. Formulas for measurements.

Measurements	Formulas
FID	$d^2 = \|\|mu_1 - mu_2\|\|^2 + Tr(C_1 + C - 2 * sqrt(C_1 * C_2))$ 5
SSIM	$\frac{(2\mu_x 2\mu_y + c1)(2\sigma_{xy} + c2)}{(\mu_x^2 + \mu_y^2 + c1)(\sigma_x^2 + \sigma_y^2 + c2)}$ 6
PSNR	$10\log_{10}(\frac{R^2}{MSE})$ 7

Results and Discussion:

The results strongly indicate that the cGAN-based model can generate high-quality and visually realistic characters, including both Urdu alphabets and numerals (Arabic and modern).

When assessing generative models, key evaluation metrics such as Peak Signal-to-Noise Ratio (PSNR), Fréchet Inception Distance (FID), and Structural Similarity Index Measure (SSIM) serve as essential benchmarks. Each metric captures a different aspect of generation quality:

FID (Fréchet Inception Distance).

A lower FID value indicates that the generated images closely resemble the real dataset in terms of feature distributions. The reported top FID scores—0.0100 for the Urdu alphabet dataset and 0.0006 for the numeral dataset (as shown in Table 2)—are exceptionally low. This suggests that the synthetic characters are nearly indistinguishable from real ones at a high-level statistical representation. The average FID values—0.0055 for the alphabet set and 0.0035 for the numerals—are still highly promising. However, the top scores emphasize the model's potential under optimal training configurations or specific hyperparameter settings.

A few sample outputs are illustrated in Figures 7(a), 7(b), and 7(c).

PSNR (Peak Signal-to-Noise Ratio): Measures how closely the generated images resemble a reference image at the pixel level. Higher PSNR values indicate less distortion and better image quality. The top PSNR scores—25.699 for Urdu alphabets and 28.844 for numerals—demonstrate that, at their best, the generated characters exhibit minimal noise and high pixel-level fidelity.

While the average PSNR values are slightly lower, they remain strong (23.4098 for alphabets and 25.9774 for numerals, as shown in Table 2). This indicates that, on average, the generated outputs are clean, detailed, and well-defined.

Some sample outputs are illustrated in Figures 7(a), 7(b), and 7(c).

SSIM (Structural Similarity Index Measure): evaluates structural and perceptual similarity, ensuring that the generated images preserve the shapes, edges, and patterns characteristic of the original character forms.

The top SSIM scores—0.9056 for alphabets and 0.9480 for numerals (as shown in Table 2)—are remarkably high, indicating that the generated characters closely resemble real examples in terms of structural integrity.

The average SSIM values—0.8320 for alphabets and 0.8347 for numerals—are also strong, demonstrating consistent structural fidelity across different generation conditions.

Some sample outputs are illustrated in Figures 7(a), 7(b), and 7(c).

MS-SSIM (Multi-Scale Structural Similarity Index Measure): is an enhanced version of SSIM that evaluates images at multiple levels of detail, from fine to coarse. This approach improves the traditional SSIM metric by comparing structural features such as brightness, contrast, and patterns across different resolutions, making it more aligned with how humans perceive visual information.

Our experimental results reveal promising outcomes, demonstrating that cGANs effectively address training data sparsity for low-resource languages. The model successfully generated coherent and contextually relevant Urdu text and numerals, emphasizing the potential of cGANs for text generation in resource-constrained environments.

This capability is significant for several reasons:

1. **Expanding Linguistic Diversity** – cGANs facilitate automated content generation, supporting underrepresented languages.
2. **Enhancing NLP Applications** – Synthetic text can augment training datasets, improving performance in text classification, sentiment analysis, and machine translation for low-resource languages.

Additionally, our findings contribute to the broader field of language modeling and GAN-based research. By demonstrating that cGANs can generate meaningful Urdu text, our study paves the way for future research on GAN-based NLP models, particularly for languages that are underrepresented in digital spaces.

The results of our various experiments, including top and average scores for each evaluation metric, are illustrated in graphs and figures throughout the study.

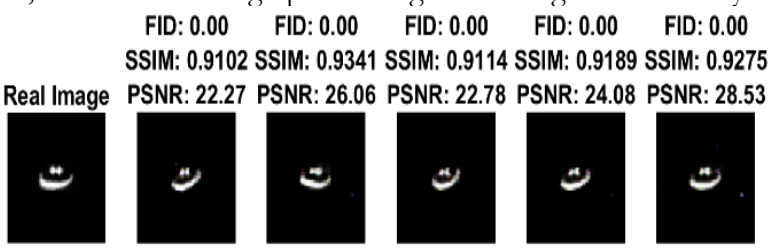

Figure 7(a). Evaluation scores for Urdu character 'stay'

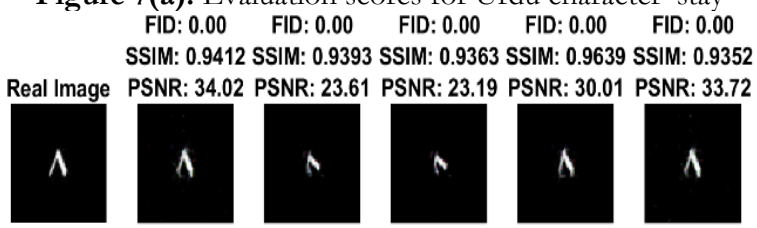

Figure 7(b). Evaluation scores for Arabic number '8'

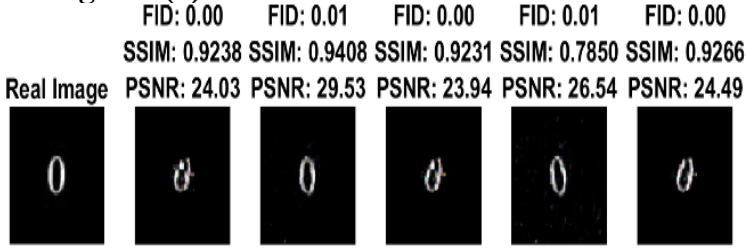

Figure 7(c). Evaluation scores for the modern number '0'

We calculated both the average scores and top scores for PSNR, FID, and SSIM for each class in the numbers dataset. The results are presented in Figure 8(a), Figure 8(b), and Figure 8(c).

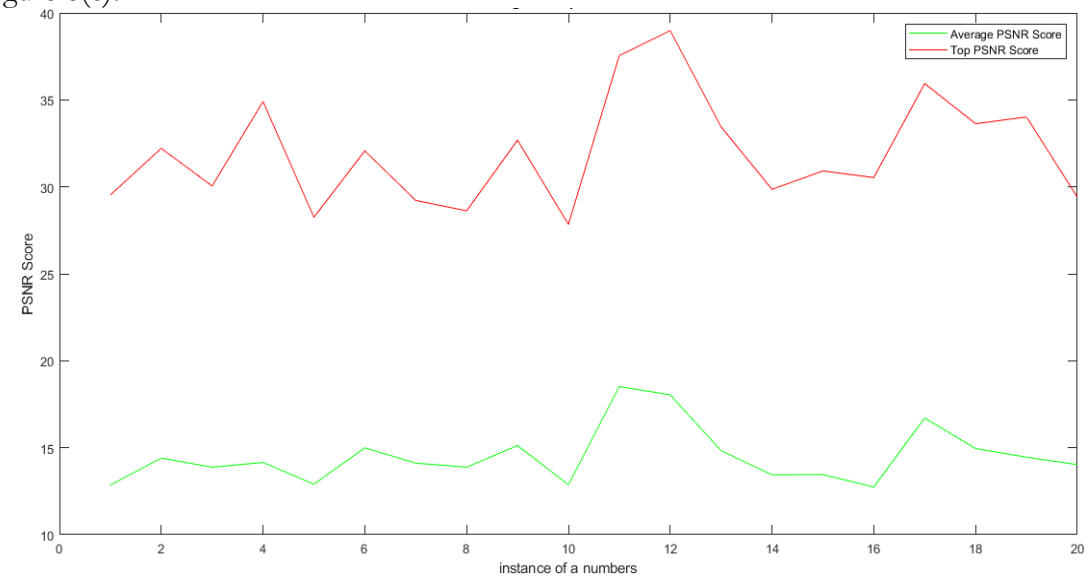

Figure 8(a). Average vs top PSNR score of numbers dataset

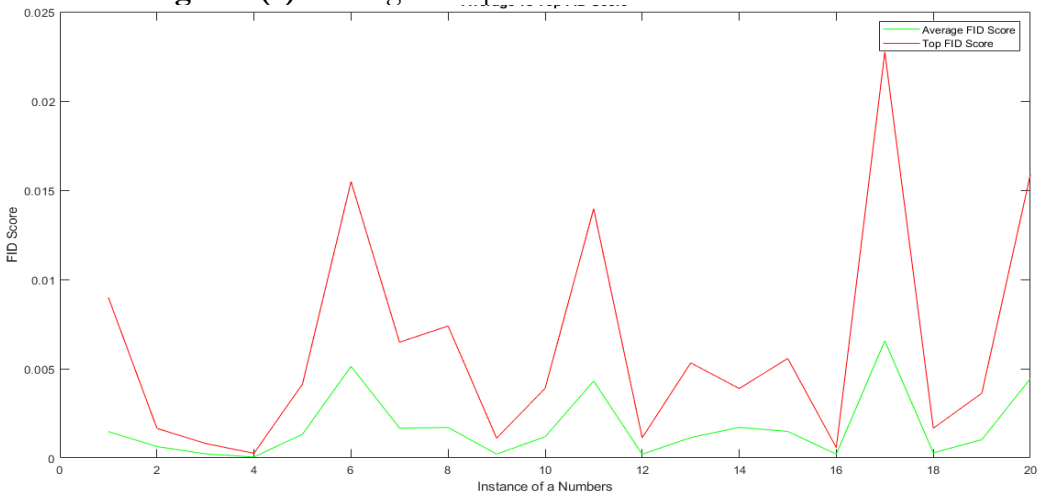

Figure 8(b). Average vs top FID score for numbers

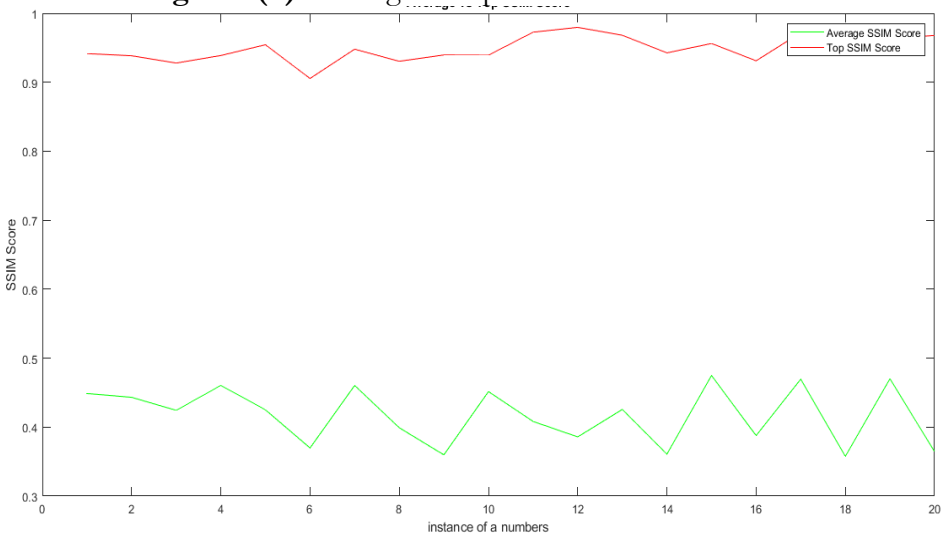

Figure 8(c). Average vs top SSIM score for numbers

The average scores and top scores for PSNR, FID, and SSIM for each class in the Urdu characters dataset are shown in Figure 9(a), Figure 9(b), and Figure 9(c).

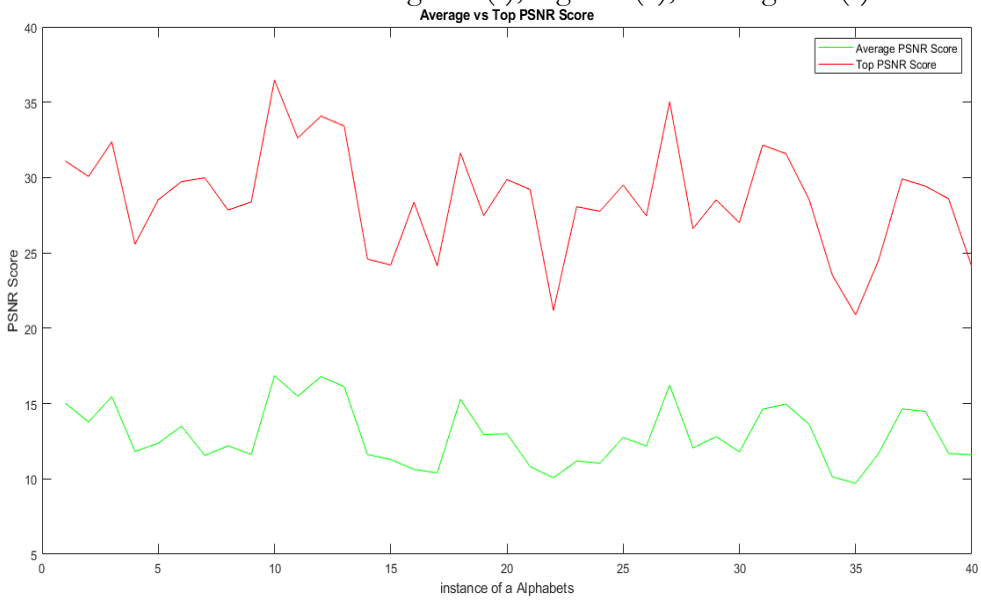

Figure 9(a). Average vs PSNR score for Urdu alphabets

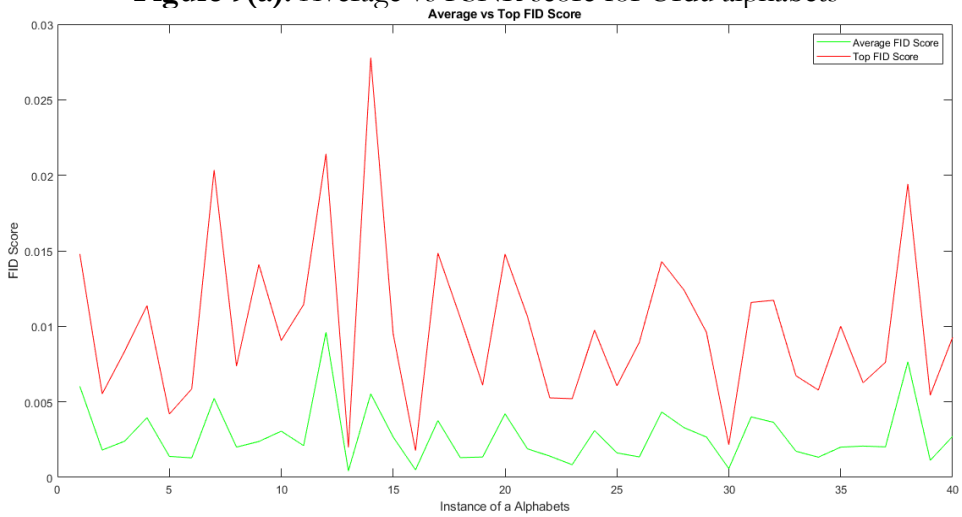

Figure 9(b) Average vs top FID score for Urdu alphabets

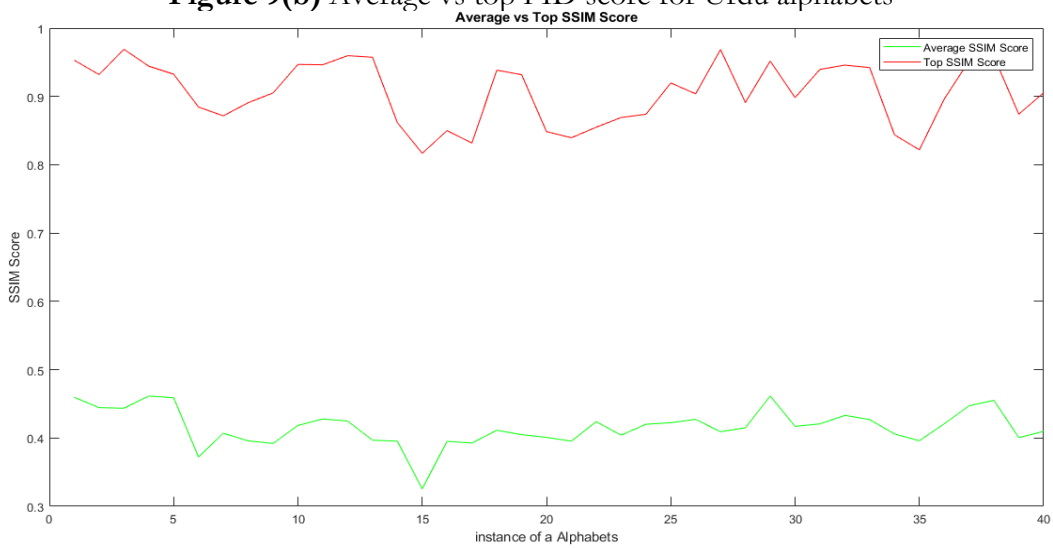

Figure 9(c). Average vs top SSIM score for the Urdu alphabet

We calculated the average scores and top scores for both datasets, as presented in Table 2.

Table 2. Various Evaluation Results

Dataset	FID		PSNR		SSIM	
Urdu alphabets	Average score	Top score average	Average score	Top score average	Average score	Top score average
	0.0055	0.0100	23.4098	25.6999	0.8320	0.9056
Numbers	0.0035	0.0062	25.9774	28.8484	0.8347	0.9480

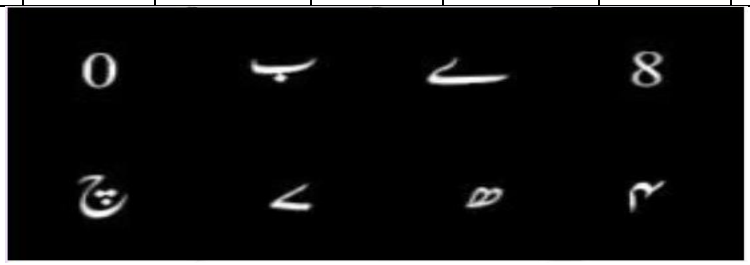

Figure 10. Samples of real images

Figure 11. samples of generated images

The generated images in Figure 11 closely resemble the real images shown in Figure 10.

- Figure 12(a) presents the average PSNR score bar chart, while Figure 12(b) displays the top PSNR score for each class of Urdu characters.
- Figure 13(a) and Figure 13(b) illustrate the average and top SSIM scores, respectively.
- Figure 14(a) and Figure 14(b) show the average and top FID scores, respectively.

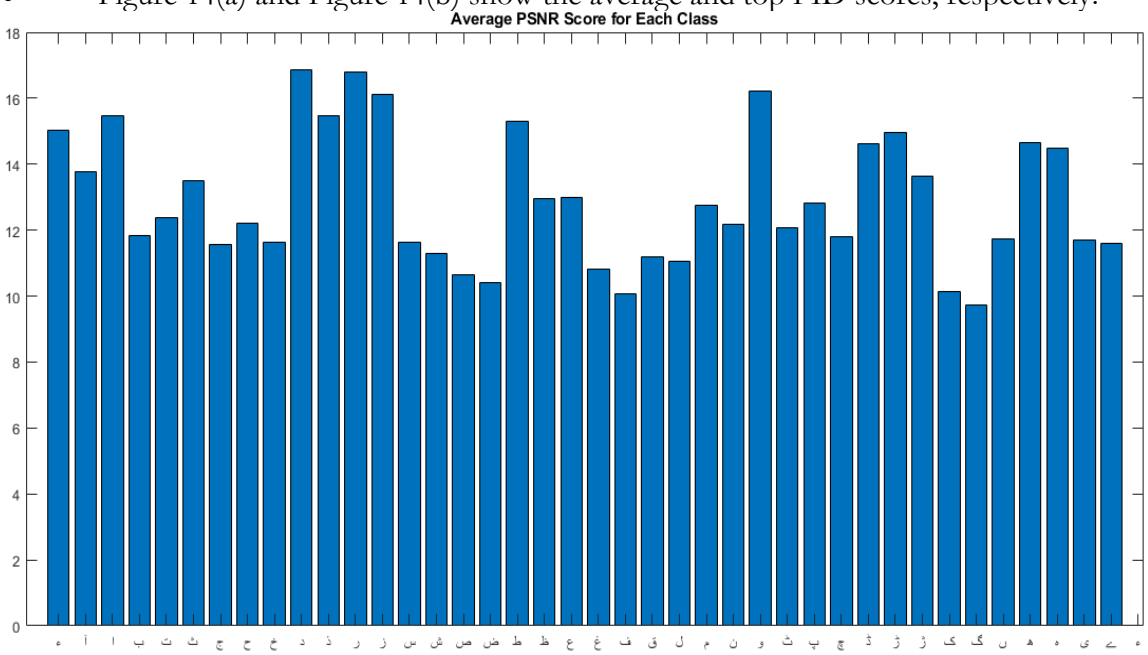

Figure12(a). Average PSNR score bar chart for each class of Urdu Characters.

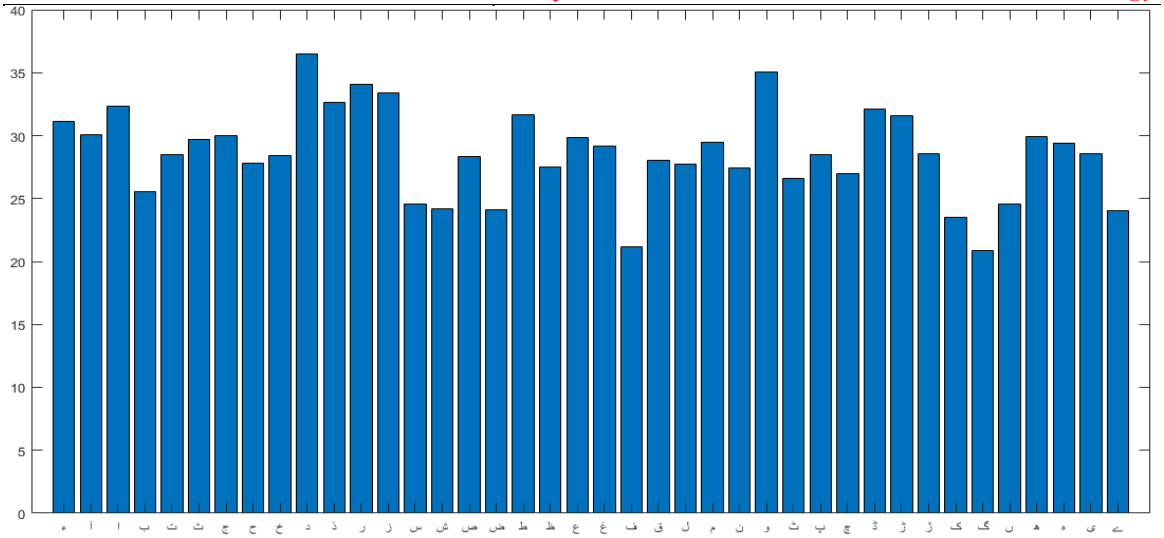

Figure12 (b). Top PSNR score bar chart for each class of Urdu characters

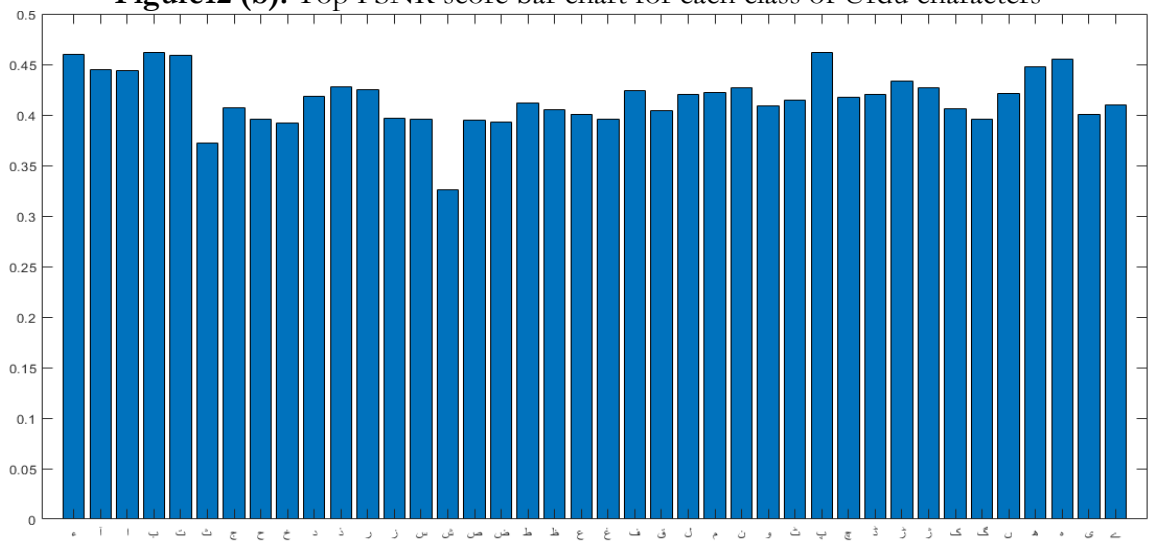

Figure13(a). Average SSIM score bar chart for each class of Urdu characters

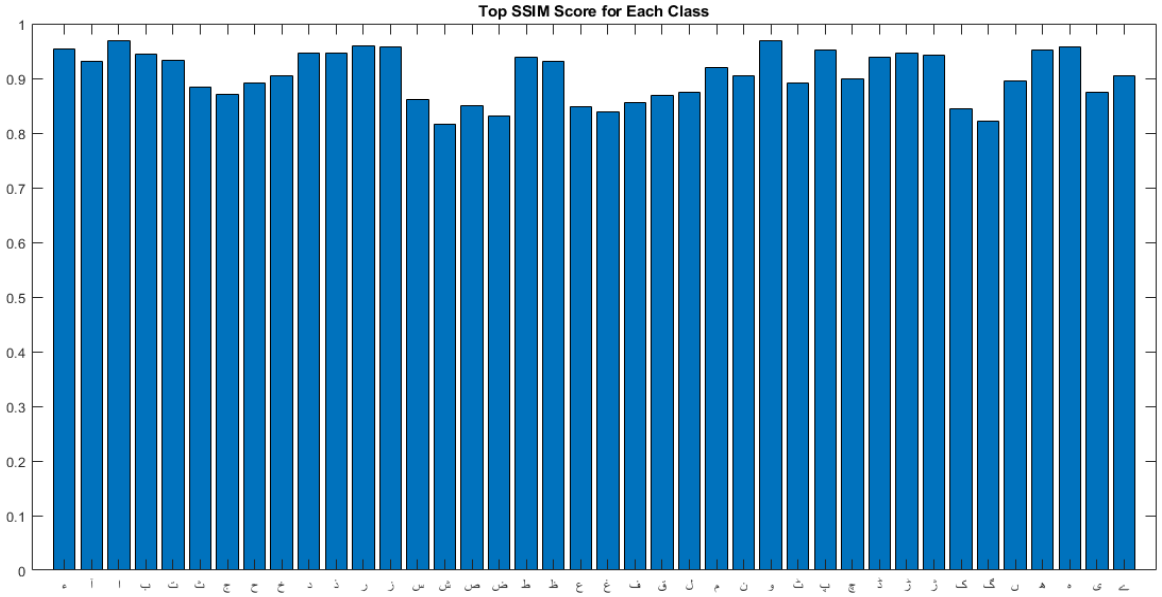

Figure13 (b). Top SSIM score bar chart for each class of Urdu characters

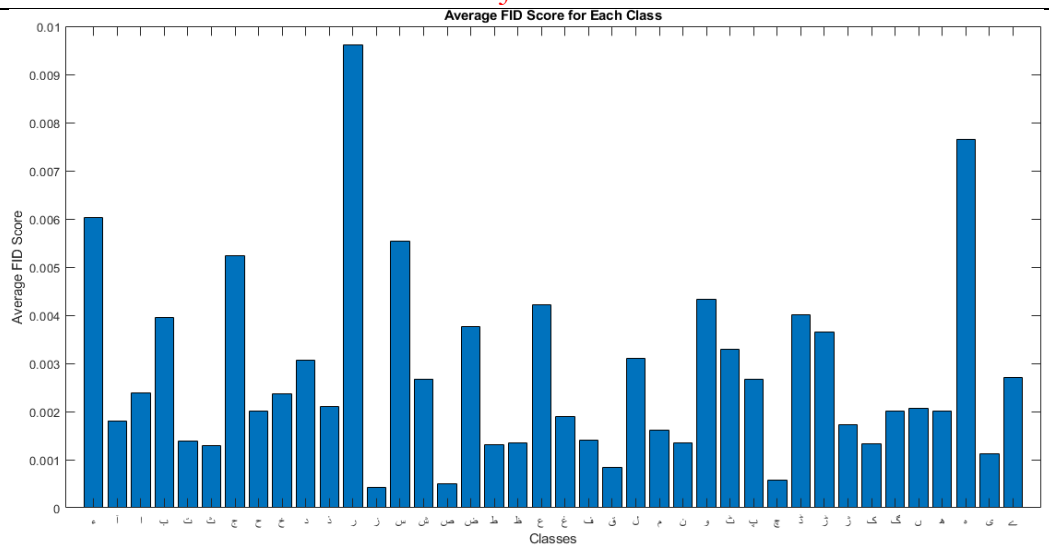

Figure14(a). Average FID score bar chart for each class of Urdu characters

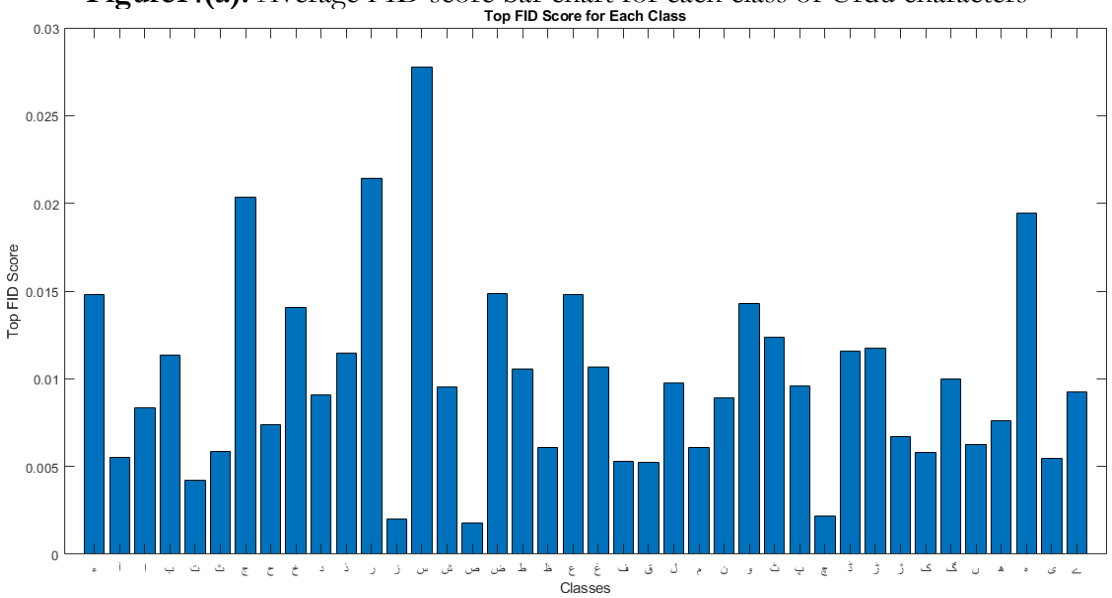

Figure14 (b). Top FID score bar chart for each class of Urdu characters

Figure 15(a) and Figure 15(b) present the average and top FID scores for each class of numbers, respectively.

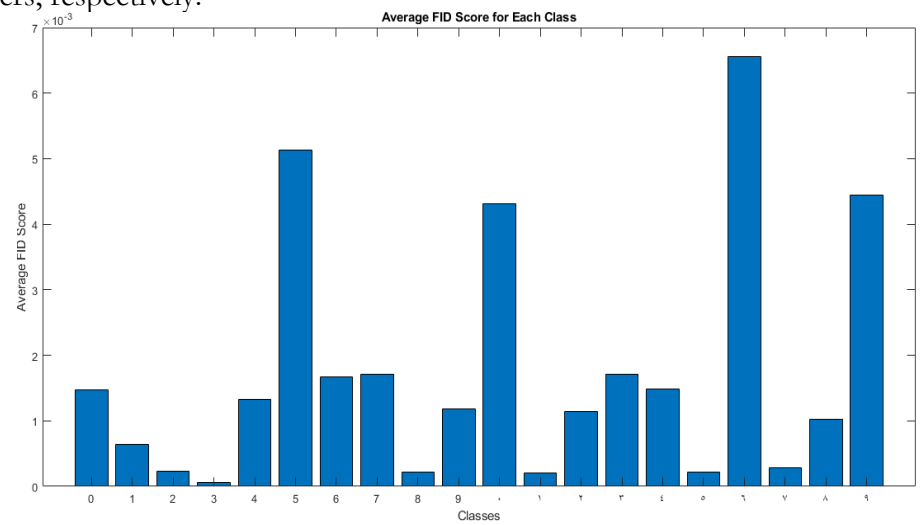

Figure15 (a). Average FID score bar chart for each class of numbers

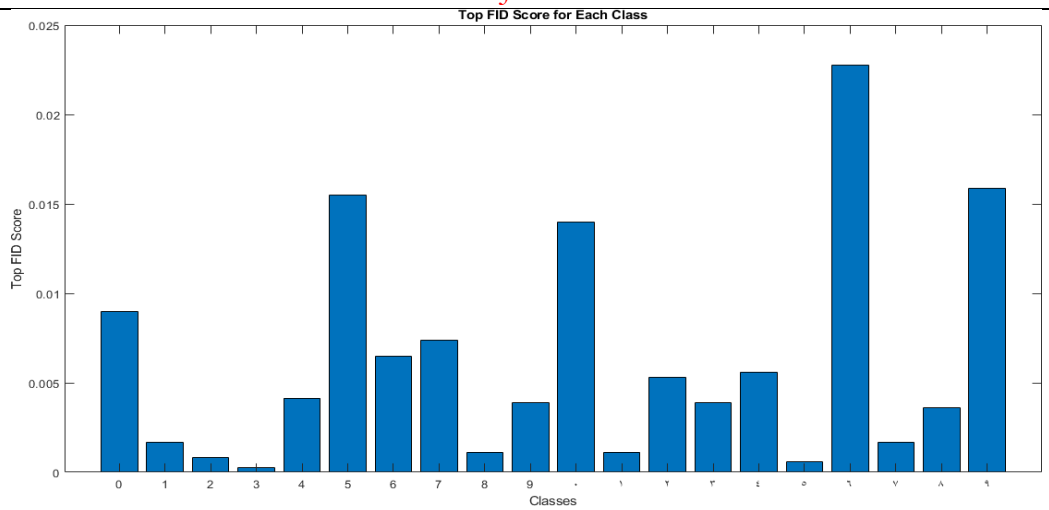

Figure15 (b). Top FID score bar chart for each class of numbers

Figure 16(a) and Figure 16(b) display the average and top PSNR scores for each class of numbers, respectively.

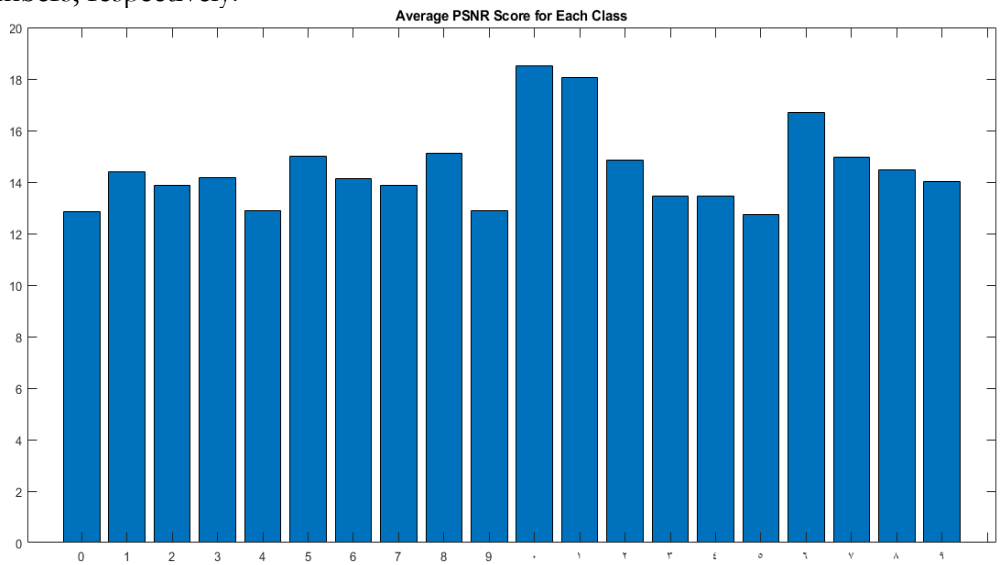

Figure16 (a). Average PSNR score bar chart for each class of numbers

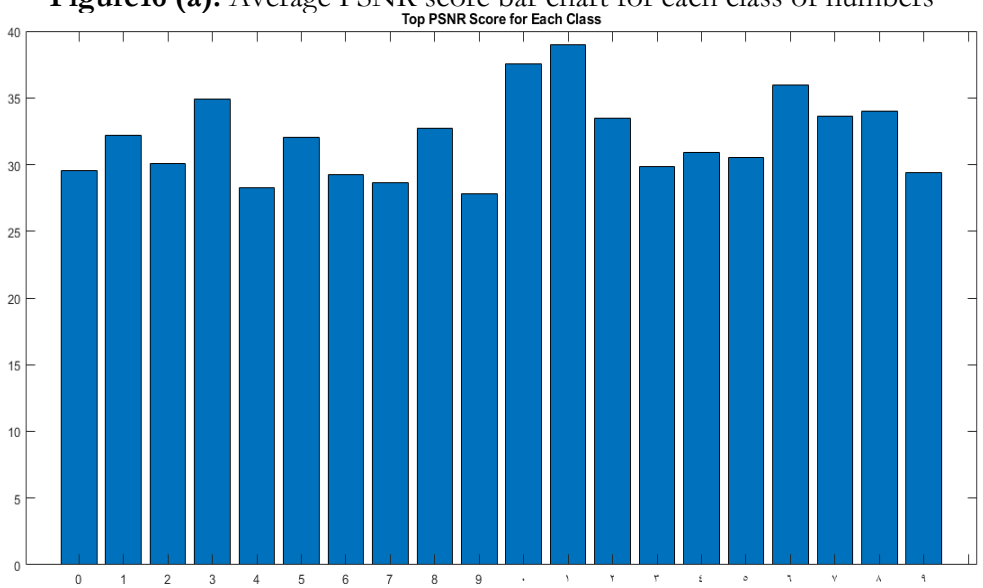

Figure16 (b). top PSNR score bar chart for each class of numbers

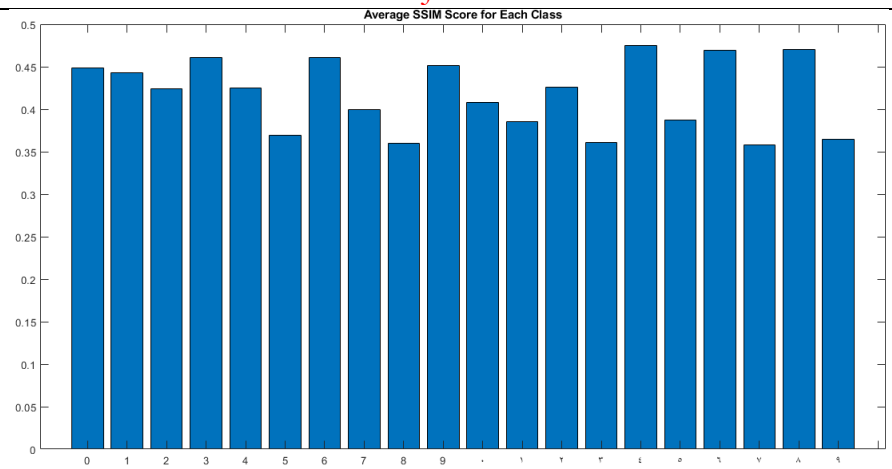

Figure17 (a). Average SSIM score bar chart for each class of numbers

Figure 17(a) and Figure 17(b) present the average and top SSIM scores for each class of numbers**,** respectively.

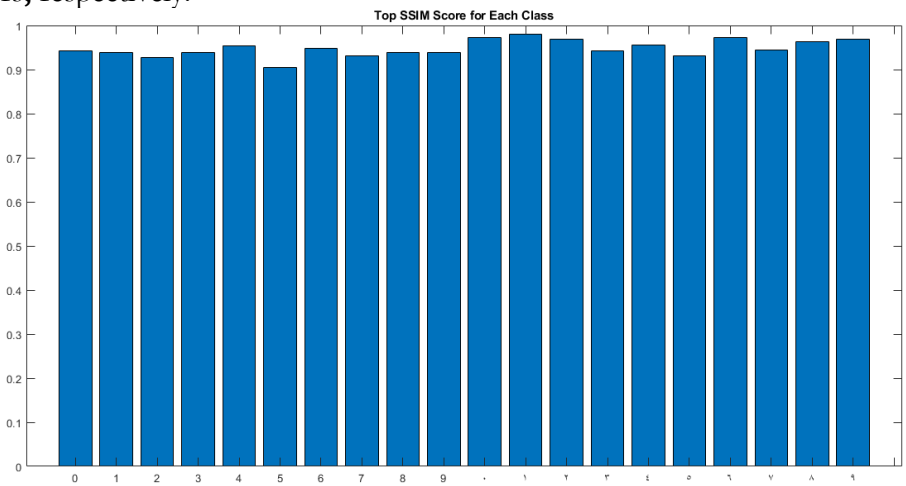

Figure17 (b) Top SSIM score bar chart for each class of numbers

We also compared our results with the Arabic cGAN for character generation using the MS-SSIM metric and obtained encouraging results**.** The findings are presented in Table 3**.**

Table 3. Comparison of MS-SSIM scores for Urdu and Arabic Characters

Dataset	Classes	MS-SSIM
Arabic characters dataset [1]	28	0.6350
Our Urdu characters dataset	40	0.6683

Discussion:

The findings of this study highlight that Conditional Generative Adversarial Networks (cGANs) offer an effective method for generating high-quality Urdu alphabet and numeral images. This contributes significantly to dataset augmentation for OCR training and script synthesis**.** The proposed model effectively captures the unique stylistic characteristics of Urdu script**,** addressing key challenges such as ligature complexity, cursive nature, and contextual character shape variations**.** Performance evaluations using PSNR, SSIM, and FID confirm that cGAN-generated images exhibit high visual fidelity and structural accuracy, making them well-suited for real-world OCR applications**.**

In comparison to traditional handwritten dataset augmentation methods**,** cGANs provide a scalable and automated solution for expanding Urdu script datasets—eliminating the need for extensive manual annotation. A key finding of this study is the superiority of cGANs over Variational Autoencoders (VAEs) and diffusion models in Urdu text generation. While diffusion models have gained popularity in text-to-image synthesis, their high

computational cost and training instability make them less practical for low-resource languages.

Our results show that cGANs achieve significantly lower FID scores (0.0055 for alphabets, 0.0035 for numerals), demonstrating superior image realism and text coherence compared to VAEs. However, despite their strong performance, cGANs still face challenges in handling highly stylized ligatures and complex diacritic placements. These limitations could be addressed through hybrid architectures incorporating transformers or attention mechanisms, enhancing the model's ability to generate more accurate and detailed Urdu text.

While this study successfully generates high-quality synthetic Urdu characters, certain deployment challenges remain unaddressed:

- **Computational Efficiency** – Further research is needed to explore the feasibility of cGANs in edge computing environments, as real-world OCR systems require lightweight, low-latency models.
- **Adversarial Robustness** – GAN-generated text images may be susceptible to perturbations, which could negatively impact OCR model performance.

Future Work:

To address these challenges, future studies should focus on:

- **Optimizing model efficiency** for real-time applications.
- **Integrating adversarial defense mechanisms** to enhance robustness against distortions and occlusions.

Comparative Analysis:

In comparison to Diffusion Models [36] and VAEs [37], which offer alternative text generation solutions:

- **Diffusion Models** are well-suited for high-quality text generation when abundant data is available.
- **VAEs** perform better in scenarios where data is sparse, but diversity is less critical.
- **cGANs**, however, provide a balanced approach, generating contextually aware, accurate, and diverse text, even with limited data availability.

Conclusion:

Overall, can emerge as a compelling solution for low-resource language generation, such as Urdu, striking a balance between diversity, contextual relevance, and practical applicability.

Importance of These Metrics and Their Strong Values:

These evaluation metrics and their high scores are crucial for several reasons:

1. **Assessing Image Quality**
   - Peak Signal-to-Noise Ratio (PSNR) measures how closely the generated images match reference images at the pixel level.
   - Higher PSNR values indicate less distortion, ensuring that the generated text remains clear and readable.
2. **Evaluating Structural Similarity:**
   - Structural Similarity Index Measure (SSIM) ensures that the shapes, edges, and patterns in the generated images preserve the structural integrity of real Urdu characters.
   - High SSIM scores confirm that the generated text remains visually and contextually accurate.
3. **Measuring Statistical Alignment**
   - Fréchet Inception Distance (FID) evaluates the similarity between real and generated datasets in terms of feature distributions.
   - Lower FID values suggest that the synthetic images are nearly indistinguishable from real data, enhancing the reliability of generated text.
4. **Enhancing OCR Training and Recognition:**

- High-quality synthetic images contribute to better dataset augmentation for OCR models.
- This improves the accuracy of text recognition systems, especially in low-resource languages like Urdu.

5. **Supporting NLP Applications**
   - These strong values validate the effectiveness of cGAN-generated text for applications in machine translation, sentiment analysis, and text classification.
   - A well-generated Urdu script dataset helps improve performance in natural language processing (NLP) tasks.
6. **Ensuring Practical Use for Real-world Applications**
   - Maintaining visual fidelity ensures that synthetic text can be used for digital typography, font generation, and educational tools.
   - This broadens the application of AI-driven text synthesis **in** linguistic research, publishing, and digital content creation.

These metrics collectively demonstrate that cGANs are a powerful tool for Urdu text generation**,** offering high-quality, structurally accurate, and statistically reliable synthetic images.

Validation of Dataset Quality:

Significance of High FID, PSNR, and SSIM Scores:

Achieving strong FID, PSNR, and SSIM scores serves as a validation of the newly developed dataset and training protocols. These high scores confirm several key aspects:

1. **Dataset Quality & Representativeness**
   - A well-curated dataset is crucial for training generative models.
   - High scores indicate that the dataset contains diverse and high-quality samples, making it effective for training robust generative models.
2. **Realism & Fidelity of Generated Images**
   - Low FID scores suggest that the generated images closely match real-world samples in feature space.
   - High PSNR values confirm minimal pixel-level distortion, ensuring that the generated characters retain clarity and detail.
   - High SSIM scores validate that the structural properties of the generated characters remain true to the original script.
3. **Effectiveness of Training Protocols**
   - High metric scores reflect the efficacy of the training pipeline**,** including data preprocessing, augmentation, and model optimization techniques.
   - This confirms that the model **learns efficiently** and generalizes well across different character classes.
4. **Suitability for OCR & NLP Applications**
   - A high-quality generative model enhances OCR dataset augmentation, making it useful for real-world Urdu script recognition.
   - Text-based AI applications, such as handwriting recognition and font generation, benefit from synthetic yet highly realistic text samples.
5. **Potential for Future Research & Development**
   - Strong metric scores indicate that the training methodology can be used for other low-resource languages and extended to various generative AI tasks.
   - This opens doors for further refinements, including the integration of transformers, diffusion models, or hybrid architectures for improved Urdu text generation.

By achieving strong FID, PSNR, and SSIM scores, this study validates the dataset's robustness and confirms that the training protocols effectively guide the model toward generating realistic, high-quality Urdu script images.

Benchmarking Performance.

Establishing Baseline Performance Metrics for a New Dataset:

When working with a new dataset, setting baseline performance metrics such as FID, PSNR, and SSIM provides a critical reference point for evaluating and improving model performance. These baselines serve several key functions:

1. **Benchmark for Future Improvements**
   - Establishing initial scores allows researchers to compare future model iterations and measure progress over time.
   - Any enhancements in preprocessing, augmentation, or architecture can be directly evaluated against the baseline.
2. **Standardized Performance Evaluation**
   - A baseline provides a consistent framework to assess different models or training strategies.
   - This is especially important for comparing cGANs, VAEs, Diffusion Models, and hybrid architectures on the same dataset.
3. **Dataset Suitability for Generative Tasks**
   - If the baseline scores are too low, it may indicate that the dataset requires better quality control, balancing, or augmentation.
   - Strong baseline scores suggest the dataset is sufficiently diverse and informative for training high-quality generative models.
4. **Guiding Hyperparameter Tuning**
   - Researchers can use baseline metrics to fine-tune learning rates, batch sizes, and regularization techniques.
   - This prevents unnecessary adjustments and provides a data-driven approach to optimizing performance.
5. **Comparability Across Studies**
   - When published, baseline scores enable other researchers to reproduce results and compare new techniques against an established reference.
   - This fosters scientific rigor and promotes collaboration in generative AI research.

By setting baseline FID, PSNR, and SSIM scores, researchers create a solid foundation for evaluating generative models, ensuring that progress is measurable, reproducible, and meaningful.

Practical Applications.

In domains such as typography design**,** calligraphy digitization, and OCR (Optical Character Recognition) pre-training, the quality of generated characters plays a critical role in determining their usability.

Why Character Quality Matters?

1. **Typography & Font Design**
   - High-quality synthetic characters help in designing new fonts with stylistic consistency**.**
   - cGANs enable the automatic generation of script-specific typefaces, reducing manual effort.
2. **Calligraphy Digitization**
   - Many historical and artistic scripts lack digital representation.
   - AI-generated characters preserve intricate calligraphic details, making them usable in modern applications.
3. **OCR Pre-Training & Dataset Augmentation**
   - OCR models require large, diverse datasets for high accuracy.
   - High-fidelity synthetic text improves OCR performance by providing additional training samples, especially for low-resource languages.

By ensuring high visual fidelity and structural accuracy, cGAN-generated text enhances usability across digital design, linguistic research, and AI-driven text recognition.

Generalization Potential:

Significance of Strong Average Scores:

While peak results highlight the best-case performance of a model, strong average scores indicate consistency across different samples, making the model more reliable in real-world applications.

1. **Robustness Across Variability**
   - A high average PSNR, SSIM, and FID suggest that the model performs well on **most samples**, not just a few ideal cases.
   - This is essential for text generation tasks, where variations in handwriting, font styles, and distortions can affect recognition.
2. **Scalability & Generalization**
   - A model with strong average performance can generalize well across different datasets, character styles, and script variations**.**
   - This is particularly valuable for low-resource languages, where real-world **dataset augmentation** is required.
3. **Foundation for Further Research**
   - The presented results confirm that cGAN-based text generation is effective and provides a benchmark for future improvements.
   - The datasets and methodologies established here can be expanded and refined for broader linguistic and AI applications.

Conclusion:

By achieving both top-performing and high-average scores, this study reinforces that cGANs offer a reliable and scalable approach **for** text and character-based image synthesis**,** paving the way for further advancements in generative AI for script-based languages.

Conclusion:

This research demonstrated the potential of using a cGAN to generate raster images of different writing systems, possibly marking the first successful attempt for Urdu characters and numerals. The GCN successfully produced realistic and recognizable representations of Urdu script, Arabic numerals, and modern numerals, demonstrating its effectiveness in font generation, optical character recognition, and data augmentation. Further study and development can explore multiple GAN designs, increase the quality and consistency of produced pictures, and look into particular applications in a range of language processing and design domains. This research opens doors for further exploration in generating raster-based representations for diverse writing systems.

This research demonstrated the potential of using cGANs to generate raster images for different writing systems, possibly marking the first successful attempt at generating Urdu characters and numerals. The model effectively produced realistic and recognizable representations of:

- Urdu script
- Arabic numerals
- Modern numerals

Key Contributions and Implications:

1. **Applications in Font Generation & OCR**
   - The study highlights cGANs' effectiveness in font design by generating high-quality, script-aware text.
   - It enhances Optical Character Recognition (OCR) systems by providing diverse synthetic data for training.
2. **Advancements in Data Augmentation**

- By generating realistic synthetic text, cGANs can expand low-resource datasets without manual annotation.
- This helps improve NLP and OCR models for underrepresented scripts.

3. **Future Research Directions**

- Exploring multiple GAN architectures (e.g., StyleGAN, BigGAN) to refine quality and consistency.
- Hybrid models with transformers to enhance context-aware text generation.
- Extending to other scripts such as Persian, Pashto, and Sindhi to broaden the research impact.

This research lays the foundation for future exploration in generating raster-based representations for diverse writing systems, paving the way for breakthroughs in computational linguistics, typography, and AI-driven text synthesis.

Acknowledgment: We are very thankful to Mirpur University of Science and Technology for the cordial environment for research.

Author's Contribution:

Suleman Khalil: Conceptualization, Methodology, Software, investigation, Writing – review & editing. *Syed Yasser Arafat:* Conceptualization, Methodology, Software, investigation, Writing – review & editing, Project Administration. *Fatima Bibi:* Conceptualization, Methodology, investigation, Writing – review. *Faiza Shafique:* Methodology, Software, investigation, Writing – review

Conflict of interest: There is no Conflict of Interest.

References:

[1] I. B. Mustapha, S. Hasan, H. Nabus, and S. M. Shamsuddin, "Conditional Deep Convolutional Generative Adversarial Networks for Isolated Handwritten Arabic Character Generation," *Arab. J. Sci. Eng.*, vol. 47, no. 2, pp. 1309–1320, Feb. 2022, doi: 10.1007/S13369-021-05796-0/METRICS.

[2] S. Haque, S. A. Shahinoor, A. K. M. S. A. Rabby, S. Abujar, and S. A. Hossain, "OnkoGan: Bangla Handwritten Digit Generation with Deep Convolutional Generative Adversarial Networks," *Commun. Comput. Inf. Sci.*, vol. 1037, pp. 108–117, 2019, doi: 10.1007/978-981-13-9187-3_10.

[3] B. Chang, Q. Zhang, S. Pan, and L. Meng, "Generating Handwritten Chinese Characters Using CycleGAN," *Proc. - 2018 IEEE Winter Conf. Appl. Comput. Vision, WACV 2018*, vol. 2018-January, pp. 199–207, May 2018, doi: 10.1109/WACV.2018.00028.

[4] A. K. Basant Babu Bhandari, Aakash Dhakal, Laxman Maharjan, "Nepali Handwritten Letter Generation using GAN," *J. Sci. Eng.*, vol. 9, no. 49–55, 2021, doi: 10.3126/jsce.v9i9.46308.

[5] M. Sharif, A. Ul-Hasan, and F. Shafait, "Urdu Handwritten Ligature Generation Using Generative Adversarial Networks (GANs)," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 13639 LNCS, pp. 421–435, 2022, doi: 10.1007/978-3-031-21648-0_29.

[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2022-June, pp. 10674–10685, Dec. 2021, doi: 10.1109/CVPR52688.2022.01042.

[7] P. A. Jonathan Ho, Ajay Jain, "Denoising Diffusion Probabilistic Models," *arXiv:2006.11239*, 2020, doi: https://doi.org/10.48550/arXiv.2006.11239.

[8] I. Ramesh, A., Chen, M., & Sutskever, "DALL·E 2:A New Generative Model for Creating Images from Text Descriptions," *OpenAI*, 2022.

[9] M. G. A. Malik, C. Boitet, and P. Bhattacharyya, "Analysis of Noori Nasta'leeq for

major Pakistani languages," *Work. Spok. Lang. Technol. Under-resourced Lang.*, 2010.
[10] A. A. Kaifi Azmi, "Urdu language," *Britannica*, 2025, [Online]. Available: https://www.britannica.com/topic/Urdu-language
[11] S. A. Husain, "A multi-tier holistic approach for Urdu Nastaliq recognition," *Int. Multi Top. Conf. 2002. Abstr. INMIC 2002., Karachi*, pp. 84–84, Aug. 2005, doi: 10.1109/INMIC.2002.1310191.
[12] N. N. Kashfi, S.M. and A. Khair, "Revolution in Urdu Composing," *Elite*, 2008, [Online]. Available: http://elite.com.pk/epl/wp-content/uploads/2011/11/N.-Nastaliq-Book-Pages.pdf
[13] S. Panwar, M. Ahamed, and N. Nain, "Ligature segmentation approach for Urdu handwritten text documents," *ACM Int. Conf. Proceeding Ser.*, vol. 11-16-November-2014, Nov. 2014, doi: 10.1145/2677855.2677856.
[14] F. Shafait, Adnan-ul-Hasan, D. Keysers, and T. M. Breuel, "Layout analysis of urdu document images," *10th IEEE Int. Multitopic Conf. 2006, INMIC*, pp. 293–298, 2006, doi: 10.1109/INMIC.2006.358180.
[15] and I. M. David Bellos, E. F. Harding, Sophie Wood, "THE UNIVERSAL HISTORY OF NUMBERS," *Harvill London*, 2008, [Online]. Available: https://ia600205.us.archive.org/28/items/TheUniversalHistoryOfNumbers/212027005-The-Universal-History-of-Numbers_text.pdf
[16] D. Anderson, "7. Digital Vitality for Linguistic Diversity," *Glob. Lang. Justice*, pp. 220–243, Nov. 2023, doi: 10.7312/LIU-21038-010/HTML.
[17] S. O. Mehdi Mirza, "Conditional Generative Adversarial Nets," *arXiv:1411.1784*, 2014, doi: https://doi.org/10.48550/arXiv.1411.1784.
[18] X. G. Ji Gan, Weiqiang Wang, Jiaxu Leng, "HiGAN+: Handwriting Imitation GAN with Disentangled Representations," *ACM Trans. Graph.*, vol. 42, no. 1, pp. 1–17, 2022, doi: https://doi.org/10.1145/355007.
[19] H. Zhang *et al.*, "StackGAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1947–1962, Aug. 2019, doi: 10.1109/TPAMI.2018.2856256.
[20] Y. W. Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, "Vector-quantized Image Modeling with Improved VQGAN," *arXiv:2110.04627*, 2022, doi: https://doi.org/10.48550/arXiv.2110.04627.
[21] D. T. Tingting Qiao, Jing Zhang, Duanqing Xu, "MirrorGAN: Learning Text-to-image Generation by Redescription," *arXiv:1903.05854*, 2019, doi: https://doi.org/10.48550/arXiv.1903.05854 Focus to learn more.
[22] R. Z. Yufan Zhou, "TiGAN: Text-Based Interactive Image Generation and Manipulation," *AAAI Conf. Artif. Intell.*, 2022, doi: 10.1609/aaai.v36i3.20270.
[23] T. Xu *et al.*, "AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1316–1324, Dec. 2018, doi: 10.1109/CVPR.2018.00143.
[24] Xi, Y. et al, "JointFontGAN: Joint Geometry-Content GAN for Font Generation via Few-Shot Learning," *MM '20 Proc. 28th ACM Int. Conf. Multimed.*, pp. 4309–43, 2020, doi: https://doi.org/10.1145/3394171.3413705.
[25] B. W. Weihao Xia, Yujiu Yang, Jing-Hao Xue, "TediGAN: Text-Guided Diverse Face Image Generation and Manipulation," *arXiv:2012.03308*, 2021, doi: https://doi.org/10.48550/arXiv.2012.03308.
[26] I. J. Goodfellow *et al.*, "Generative Adversarial Nets," *Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, Accessed: Oct. 02, 2023. [Online]. Available: http://www.github.com/goodfeli/adversarial

[27] E. Alonso, B. Moysset, and R. Messina, "Adversarial generation of handwritten text images conditioned on sequences," *Proc. Int. Conf. Doc. Anal. Recognition, ICDAR*, pp. 481–486, Sep. 2019, doi: 10.1109/ICDAR.2019.00083.

[28] Y. C. and J. M. Q. Wu, "DCGAN-Based Data Augmentation for Tomato Leaf Disease Identification," *IEEE Access*, vol. 8, pp. 98716–98728, 2020, doi: 10.1109/ACCESS.2020.2997001.

[29] S. Y. Arafat and M. J. Iqbal, "Two Stream Deep Neural Network for Sequence-Based Urdu Ligature Recognition," *IEEE Access*, vol. 7, pp. 159090–159099, 2019, doi: 10.1109/ACCESS.2019.2950537.

[30] S. Y. Arafat and M. J. Iqbal, "Urdu-Text Detection and Recognition in Natural Scene Images Using Deep Learning," *IEEE Access*, vol. 8, pp. 96787–96803, 2020, doi: 10.1109/ACCESS.2020.2994214.

[31] M. Guan, H. Ding, K. Chen, and Q. Huo, "Improving Handwritten OCR with Augmented Text Line Images Synthesized from Online Handwriting Samples by Style-Conditioned GAN," *Proc. Int. Conf. Front. Handwrit. Recognition, ICFHR*, vol. 2020-September, pp. 151–156, Sep. 2020, doi: 10.1109/ICFHR2020.2020.00037.

[32] D. Yorioka, H. Kang, and K. Iwamura, "Data Augmentation for Deep Learning Using Generative Adversarial Networks," *2020 IEEE 9th Glob. Conf. Consum. Electron. GCCE 2020*, pp. 516–518, Oct. 2020, doi: 10.1109/GCCE50665.2020.9291963.

[33] A. Horé and D. Ziou, "Image quality metrics: PSNR vs. SSIM," *Proc. - Int. Conf. Pattern Recognit.*, pp. 2366–2369, 2010, doi: 10.1109/ICPR.2010.579.

[34] H. R. S. and E. P. S. Zhou Wang, A. C. Bovik, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004, doi: 10.1109/TIP.2003.819861.

[35] Y. D. Yu Yu, Weibin Zhang, "Frechet Inception Distance (FID) for Evaluating GANs," China University of Mining Technology Beijing Graduate School: Beijing, China. Accessed: Mar. 13, 2025. [Online]. Available: https://www.researchgate.net/publication/354269184_Frechet_Inception_Distance_FID_for_Evaluating_GANs

[36] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," *Adv. Neural Inf. Process. Syst.*, vol. 2020-December, Jun. 2020, Accessed: Oct. 01, 2023. [Online]. Available: https://arxiv.org/abs/2006.11239v2

[37] M. W. Diederik P Kingma, "Auto-Encoding Variational Bayes," *arXiv:1312.6114*, 2013, doi: https://doi.org/10.48550/arXiv.1312.6114.