Bengali Handwritten Character Recognition: A Comparative Study of CNN Variants and Lightweight Pretrained Models on Small Size Dataset

Project-I (MA47201) report submitted to
Indian Institute of Technology Kharagpur

in partial fulfilment for the award of the degree of

BS-MS in STATISTICS AND DATA SCIENCE

by

Bimal Gayali
(21MA25018)

Under the supervision of

Professor Debjani Chakraborty

Department of Mathematics
Indian Institute of Technology Kharagpur

Autumn Semester, 2024-25

November 26, 2024

Declaration

I certify that:

(a) The work contained in this report has been done by me under the guidance of my supervisor.

(b) The work has not been submitted to any other Institute for any degree or diploma.

(c) I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

(d) Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

Place: Kharagpur **Bimal Gayali**
Date: November 26, 2024 Roll: 21MA25018

DEPARTMENT OF MATHEMATICS
INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR
KHARAGPUR - 721302, INDIA

Certificate

This is to certify that the project report entitled **"Bengali Handwritten Character Recognition: A Comparative Study of CNN Variants and Lightweight Pretrained Models on Small Size Dataset"** submitted by **Bimal Gayali (Roll No. 21MA25018)** to Indian Institute of Technology Kharagpur towards partial fulfilment of requirements for the award of degree of Bachelor of Technology in Mathematics is a record of bona fide work carried out by him under my supervision and guidance during Autumn Semester, 2024-25.

Professor Debjani Chakraborty
Supervisor
Department of Mathematics
Indian Institute of Technology Kharagpur
Kharagpur - 721302, India

Place: Kharagpur
Date: November 26, 2024

Contents

Abstract

Name of the student: Bimal Gayali
Roll No: 21MA25018

Department: Department of Mathematics

Thesis title: Bengali Handwritten Character Recognition: A Comparative Study of CNN Variants and Lightweight Pretrained Models on Small Size Dataset

Thesis supervisor: Professor Debjani Chakraborty

Month and year of thesis submission: November 26, 2024

1 Introduction

Handwritten character recognition for Bengali script remains a challenging task due to the script's intricate structure, high variability in handwriting styles, and often limited availability of labeled data. Bengali characters consist of complex curves, multiple strokes, and subtle diacritics that make distinguishing between characters difficult. This complexity is compounded by the scarcity of large datasets, which are crucial for training high-performing deep learning models. Most state-of-the-art neural networks, such as Convolutional Neural Networks (CNNs), typically rely on extensive training data to achieve robust accuracy, making it difficult to adapt these models for Bengali character recognition, especially with limited size data.

The purpose of this research is to explore effective methods for Bengali handwritten character recognition using this limited dataset. We examine various neural network architectures, including standard CNNs, residual networks, and dilated CNNs, as well as the integration of Squeeze-and-Excitation (SE) blocks to see how well these approaches can capture the unique features of Bengali characters. Additionally, we compare the effectiveness of pretrained models like MobileNet, ResNet, DenseNet, and VGG16 in this context, as these models have demonstrated success in transfer learning for many recognition tasks and might offer an advantage when training data is limited.

By combining both custom and pretrained architectures, this study aims to identify models that achieve high accuracy with small datasets, thus addressing a significant gap in Bengali character recognition research. Our findings could inform future work in regional handwriting recognition and contribute to the development of practical, efficient systems for recognizing Bengali text in real-world applications.

2 Literature Review

Bengali handwritten character recognition has been a challenging task due to the complexity of the script and the variations in handwriting styles. Existing approaches, mostly based on convolutional neural networks (CNNs), have shown some success, but they often struggle when applied to small datasets due to overfitting, poor generalization, and limited data for feature learning.

Many studies have employed standard CNN architectures for Bengali handwritten character recognition, achieving reasonable accuracy with larger datasets. However, when these models are applied to smaller datasets, they often fail to generalize well.

Residual networks (ResNets) have been introduced to address this issue by allowing the training of deeper networks, improving the extraction of features from complex characters. While ResNets have shown better results than standard CNNs, they still face challenges when trained on small datasets, as they can also overfit and struggle to learn robust features from the limited data.

Dilated convolution networks, on the other hand, have been proposed as a solution to capture contextual information over larger receptive fields without reducing spatial resolution. While dilated convolutions improve the model's ability to capture global features, they still require large datasets for optimal performance.

Squeeze-and-Excitation (SE) blocks have been incorporated in some models to enhance feature recalibration and allow the network to focus on more important features. While SE blocks have improved recognition accuracy, their use in Bengali handwritten recognition with small datasets remains limited, and their impact has not been fully explored.

In this research, we aim to improve the recognition of Bengali characters on small-sized datasets by combining multiple advanced techniques. Specifically, we integrate SE blocks into existing CNN and ResNet architectures to enhance feature selection, and we explore dilated convolutions to improve the contextual understanding of the characters. Additionally, we compare the performance of popular pretrained models like MobileNet, ResNet50,

DenseNet121, and VGG16. This approach allows us to leverage the power of pretrained models while addressing the limitations of small dataset sizes, thus improving the overall performance of Bengali handwritten character recognition.

By introducing these techniques, this research contributes to the enhancement of Bengali character recognition accuracy, especially for smaller datasets, which have been a significant challenge in the existing literature.

3 Preliminaries

This section presents the models and techniques utilized in this study for Bengali handwritten character recognition. The techniques applied are based on Convolutional Neural Networks (CNNs), Residual Neural Networks (ResNets), Dilated Convolutions, Squeeze-and-Excitation (SE) blocks, and pretrained models. These methods are integrated and evaluated to improve the recognition accuracy on small Bengali handwritten datasets.

3.1 Dataset Description

The dataset used for Bengali Handwritten Character Recognition contains **50 distinct classes**, each representing a different Bengali character. It includes a total of **12,000 training images** and **3,000 test images**, with each image corresponding to one of the characters. These characters include a range of Bengali vowels, consonants, and additional symbols.

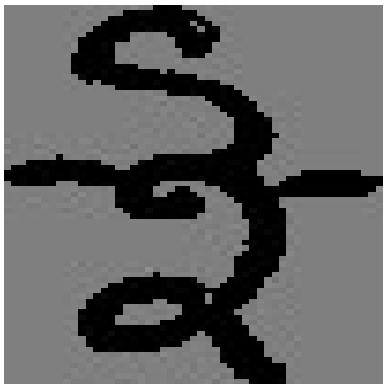


Figure 1: Example 1: Grayscale Bengali character image.

Figure 2: Example 2: Another grayscale Bengali character image.

Fig 1: Example of Bangla Characters

Key Features:

- **Classes:** The dataset covers 50 Bengali characters, such as basic vowels and consonants along with special symbols and variants.
- **Image Size:** Each image is resized to **32x32 pixels** to maintain consistency across the dataset, enabling efficient training and testing.
- **Color Format:** The images are in **grayscale** to simplify the learning task by reducing color complexity, allowing the model to focus on the character shapes and structure.
- **Variations:** The dataset captures a wide range of handwriting styles, including differences in stroke thickness, character spacing, and orientation. These variations are crucial for the model's ability to generalize and recognize characters in diverse handwriting styles.

Data Splits:

- **Training Set:** 12,000 images are used for model training, enabling the model to learn and optimize its parameters.
- **Test Set:** 3,000 images are used for evaluation to assess the model's performance and generalization after each epoch.

This dataset is specifically designed for Bengali handwritten character recognition, and its diversity in handwriting styles poses a challenge for models to achieve high accuracy. The variety within the dataset aids in developing models that can generalize well across unseen handwritten samples.

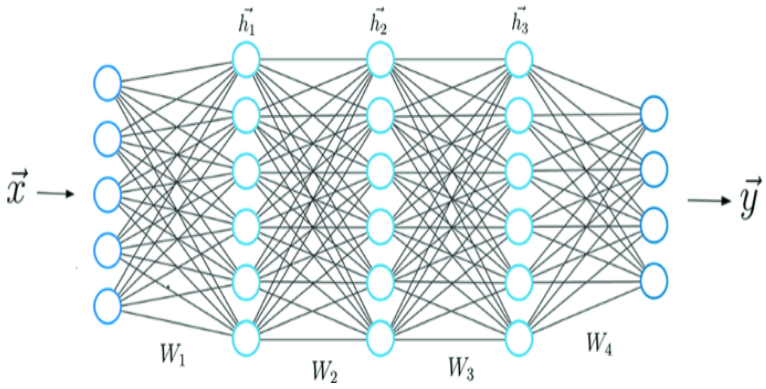


Figure 3: Deep Neural Network

3.2 Bengali Character Recognition Process

The recognition of Bengali characters using deep learning techniques involves several key steps, each contributing to the model's ability to correctly identify the characters. The following steps outline the general process used for Bengali character recognition:

1. **Convolutional Layers:** The first step in character recognition is the extraction of low-level features from the input images using convolutional layers. Convolutional filters slide over the image to detect edges, textures, and simple patterns such as lines or curves. These features help the model distinguish different shapes and structures within the Bengali characters.

2. **Max Pooling:** After the convolution operation, the model applies Max Pooling. This technique reduces the dimensionality of the image by selecting the maximum value from a pool of values, effectively down-sampling the image while retaining the most important features. Pooling helps the network focus on the most prominent features and reduces the computational load.

3. **Dropout:** To prevent overfitting, Dropout is applied during training. This technique randomly sets a fraction of the input units to zero at each update during training time, effectively "dropping out" some neurons. Dropout helps the model generalize better and prevents it from memorizing the training data, improving performance on unseen data.

4. **Fully Connected Layers and Classification:** After extracting relevant features, the model passes the data through fully connected layers.

These layers learn high-level abstract representations of the features. The final layer uses a softmax activation function to classify the input image into one of the 50 character classes.

5. **Evaluation and Fine-Tuning:** The model is evaluated on a test set of images. During evaluation, the model's hyperparameters, such as learning rate and layer configuration, are adjusted to optimize performance.

This process involves a series of operations, such as convolution, pooling, feature extraction, and classification, that work together to accurately recognize Bengali characters. The diagram below illustrates this general process, showing how the input image is processed and classified.

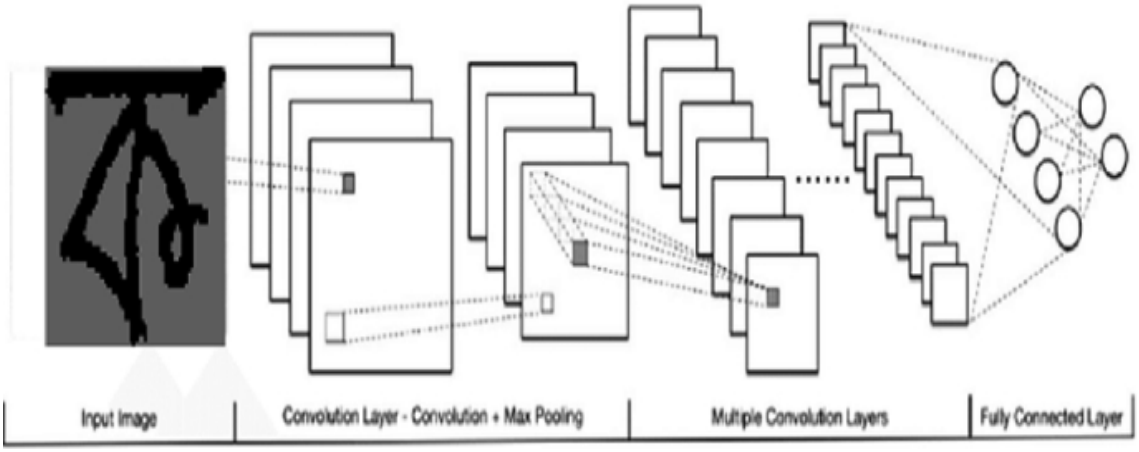


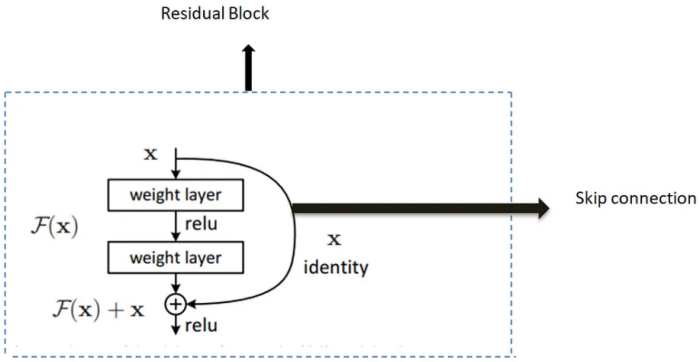
Figure 4: Bengali Character Recognition Process Using Neural Network

3.3 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are the backbone of many image recognition tasks due to their ability to automatically learn spatial hierarchies from input images. The architecture of CNNs typically involves multiple convolutional layers followed by pooling and fully connected layers. CNNs are effective at extracting local features, making them useful for character recognition tasks. However, CNNs can struggle with small datasets, leading to overfitting. Therefore, in this research, we enhance CNNs with additional techniques to improve generalization and accuracy.

3.4 Residual Neural Networks (ResNets)

Residual Neural Networks (ResNets) address the challenge of training deep neural networks by introducing skip connections between layers. These connections allow gradients to flow more easily, preventing the vanishing gradient problem and enabling the network to learn more complex features. While ResNets have demonstrated great success in large-scale datasets, their performance on smaller datasets like Bengali handwritten characters is less well explored. In this research, we investigate the effectiveness of ResNets for Bengali character recognition on small datasets.



3.5 Dilated Convolutions

Dilated convolutions are used to increase the receptive field of convolutional layers without adding extra parameters. This technique helps the model to capture global context information, which is beneficial for understanding the intricate features of handwritten characters. Dilated convolutions are particularly useful in tasks where large-scale context is crucial, and in this study, they are applied to CNN and ResNet architectures to improve their performance on the Bengali handwritten dataset.

3.6 Squeeze-and-Excitation (SE) Block

The Squeeze-and-Excitation (SE) block is a lightweight mechanism that recalibrates the feature maps by focusing on the most informative channels. This helps the model to emphasize important features, improving its ability to distinguish between subtle variations in character shapes. By integrating SE blocks into CNN and ResNet models, we aim to enhance the model's representational power, thereby improving its performance on Bengali character recognition.

3.7 Pretrained Models

Transfer learning using pretrained models has shown significant success in improving the performance of models, especially when training data is limited. In this study, we utilize popular pretrained models such as VGG16, ResNet50, DenseNet121, and MobileNet. These models have been trained on large datasets like ImageNet and are fine-tuned on the Bengali handwritten dataset. This approach helps leverage the learned features from large datasets, boosting recognition accuracy on the smaller Bengali dataset.

By combining these techniques, our research aims to improve the recognition accuracy of Bengali handwritten characters, particularly in the context of small datasets where traditional models might underperform.

4 Standard Optimize CNN

In this study, we implemented a Convolutional Neural Network (CNN) model to classify Bengali handwritten characters. The architecture of the CNN model is as follows:

- The first layer is a 2D convolutional layer with 128 filters of size (3, 3), followed by a max pooling layer with pool size (2, 2) and a dropout layer with a rate of 0.2.
- The second layer is another convolutional layer with 64 filters, followed by max pooling and dropout.
- The output from the convolutional layers is flattened and passed through a dense layer with 128 units, followed by a dropout layer.
- The final output layer has 50 units, corresponding to the 50 character classes, and uses the softmax activation function for classification.

This model was trained using the Adam optimizer and categorical cross-entropy loss function. The architecture is summarized as follows:

4.1 Model Training and Results

The model was trained using a dataset consisting of 12,000 training images and 3,000 test images, with 50 character classes. The training data was augmented using random shear and rotation, and the images were resized to 40x40 pixels. The model achieved a validation accuracy of approximately 87% after 10 epochs.

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	0.2643	0.6610	2.8051	1.2414
2	0.5803	0.7537	1.4549	0.8233
3	0.6612	0.7960	1.1688	0.6923
4	0.7029	0.8203	1.0105	0.5880
5	0.7261	0.8373	0.9141	0.5434
6	0.7433	0.8530	0.8521	0.5071
7	0.7606	0.8253	0.7780	0.5831
8	0.7694	0.8690	0.7517	0.4593
9	0.7854	0.8627	0.7106	0.4750
10	0.7976	0.8790	0.6671	0.4145

Table 1: Training and Validation Results per Epoch for Standard CNN Model

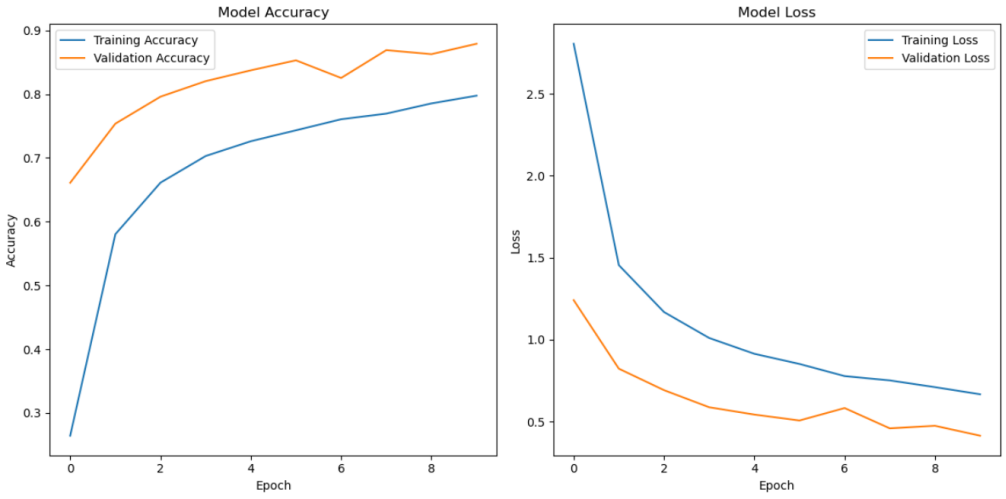


Figure 5: CNN Model Architecture for Bengali Handwritten Character Recognition

4.2 Model Performance

The CNN model's performance over the epochs is summarized in the following table:

The model showed significant improvements in accuracy during the training process, with validation accuracy reaching 87% by the 10th epoch.

5 Residual Neural Network

In this study, a custom Residual Neural Network (ResNet) model with multi-scale convolutional layers is implemented for Bengali handwritten character recognition. This architecture aims to enhance the model's feature extraction capabilities and generalization on a limited dataset.

5.1 Overview of the Model

The model incorporates multiple feature extraction techniques to handle the complex patterns of Bengali handwritten characters effectively. The architecture includes standard residual blocks, multi-scale convolutional blocks, and Spatial Dropout to increase robustness. The residual blocks facilitate the flow of gradients, aiding in efficient training and helping the model learn more complex features. Additionally, multi-scale convolutional layers allow for the extraction of different levels of spatial information from the characters, enabling the model to capture subtle nuances in character shapes.

5.2 Standard Residual Block

The standard residual block is used to prevent the vanishing gradient problem by adding a skip connection that bypasses the non-linear transformations. In our model, each residual block consists of two convolutional layers, each followed by Batch Normalization. The output of these layers is then added to the input of the block, creating a shortcut connection. This design aids in efficient feature learning, especially in deeper networks, which is critical for accurate character recognition.

5.3 Multi-Scale Convolution Block

The multi-scale convolution block extracts features at different scales by using separate convolutional layers with kernel sizes of 1x1, 3x3, and 5x5, followed by concatenation. This block enables the model to capture both fine-grained and coarse features of the input images, which improves its ability to recognize diverse and intricate patterns found in Bengali characters. The concatenated output from these multi-scale convolutions provides a richer representation of the input data.

5.4 Spatial Dropout Layer

To improve generalization, Spatial Dropout layers are added after each pooling layer. Spatial Dropout randomly drops entire channels, rather than individual neurons, during training, which encourages the network to learn redundant features across channels. This technique is particularly effective in image-based models as it enhances robustness and reduces overfitting.

5.5 Training Details

The model was trained for 10 epochs with the Adam optimizer and categorical cross-entropy loss function. Table 2 summarizes the accuracy and loss values across epochs, showing the model's performance improvements. The use of multi-scale features and residual connections contributed to increased accuracy and stability in training, with a final validation accuracy of 91.93%.

5.6 Variation in Test and Validation Results Across Epochs

During training, the validation and test results typically show a fluctuation across epochs. The training accuracy generally increases as the model learns the features from the dataset, while the validation accuracy may not always follow the same trend due to the model's exposure to new unseen data in the validation set. The results are affected by the following:

- **Overfitting/Underfitting:** Initially, the model may underfit, and as training progresses, the model's performance improves, reducing both training loss and validation loss. However, if the model becomes too complex or overfits, the validation accuracy may plateau or decrease while training accuracy continues to improve.
- **Learning Rate:** The learning rate plays a crucial role in how quickly the model converges. A high learning rate may cause the model to miss optimal solutions, while a low learning rate might lead to slow convergence and longer training times.
- **Generalization:** The model's ability to generalize is often tested on validation data. As the model improves through each epoch, the validation accuracy should ideally approach the training accuracy, but fluctuations may still occur due to the inherent difficulty of the task or small changes in the dataset during training.

The test and validation results may vary significantly as the model adapts and refines its understanding. The results in Table 2 illustrate the overall improvements in both training and validation accuracy across the epochs, with a final increase in validation accuracy of up to 91.93%.

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	0.0878	0.0483	4.8958	4.4037
2	0.3093	0.4147	3.1551	2.8499
3	0.5251	0.6993	2.4820	2.1313
4	0.6526	0.7520	2.1254	2.1826
5	0.7193	0.8237	1.9195	1.6335
6	0.7663	0.8760	1.7204	1.3709
7	0.7933	0.8773	1.5816	1.3202
8	0.8184	0.9130	1.4682	1.1291
9	0.8263	0.9060	1.4030	1.1208
10	0.8401	0.9193	1.3175	1.1124

Table 2: Training and Validation Results per Epoch for ResNet Architecture

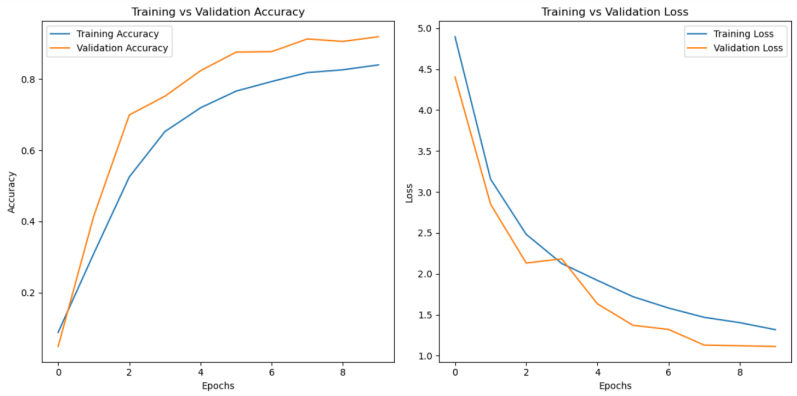


Figure 6: Training and Validation Results. (Left) Accuracy per epoch; (Right) Loss per epoch. The left side of the image shows training and validation accuracy, while the right side shows training and validation loss across epochs.

6 Dilated Residual Neural Network

The proposed model is an enhanced convolutional neural network (CNN) designed for Bengali handwritten character recognition. It incorporates a residual block with dilated convolutions and a multi-scale convolution block to capture various levels of feature abstraction. Below is the detailed architecture of the model:

1. Input layer with shape $(40, 40, 3)$.
2. Convolutional layer with 64 filters, kernel size (3, 3), and ReLU activation.
3. Batch normalization.
4. Residual block with dilated convolutions (dilation rate = 1, 2, or 3).
5. Max pooling layer.
6. Spatial dropout layer (rate = 0.3) for robustness.
7. Multi-scale convolution block with 128 filters, combining kernel sizes (1, 1), (3, 3), and (5, 5).
8. Flatten layer.
9. Fully connected layers with 128 units and ReLU activation.
10. Output layer with 50 units and softmax activation (for 50 classes).

6.1 Dilated Residual Blocks

The dilated residual block utilizes convolutions with dilation rates of 1, 2, and 3 to capture varying receptive fields without increasing computational complexity. The residual connections are added to preserve gradient flow during training.

6.2 Performance Comparison: Dilated Rate = 1

The training log below shows the performance of the enhanced model with a dilation rate of 1 during 10 epochs of training. The model achieved notable improvements in both training and validation accuracy, as summarized in the table and illustrated in the performance graph.

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	0.0814	0.1100	4.9784	4.2023
2	0.3188	0.6470	3.1753	2.0496
3	0.5300	0.7423	2.6121	1.8723
4	0.6562	0.8283	2.2226	1.5851
5	0.7321	0.8173	1.8528	1.6231
6	0.7657	0.8510	1.7181	1.8110
7	0.7959	0.9080	1.5666	1.1939
8	0.8223	0.8867	1.4725	1.3012
9	0.8323	0.9130	1.4036	1.1544
10	0.8392	0.9160	1.3891	1.2303

Table 3: Performance for Dilated Rate = 1

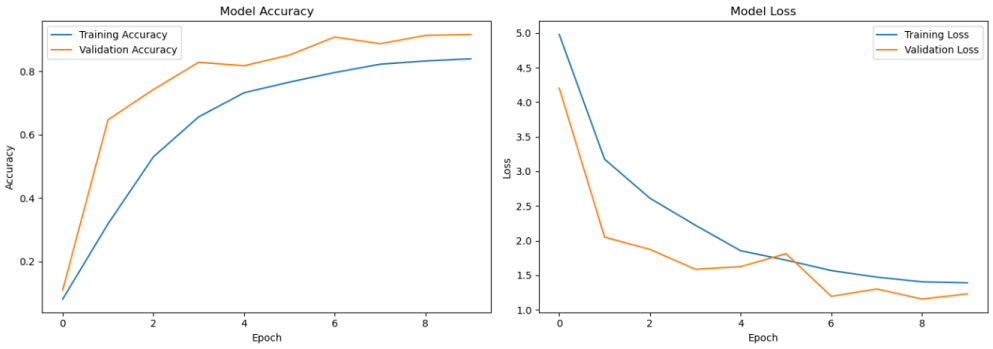


Figure 7: Training and Validation Accuracy for Dilated Rate = 1

6.3 Performance Comparison: Dilated Rate = 2

The training log for the enhanced model with a dilation rate of 2 is shown below. This configuration demonstrated strong performance with significant gains in validation accuracy.

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	0.0828	0.1277	4.7868	3.9457
2	0.3373	0.6163	3.1469	2.1044
3	0.5443	0.5600	2.5218	3.4820
4	0.6440	0.8187	2.2619	1.6709
5	0.7267	0.8390	1.8905	1.4517
6	0.7645	0.8710	1.7843	1.4738
7	0.7975	0.8897	1.6309	1.3143
8	0.8212	0.8943	1.5299	1.1922
9	0.8376	0.8723	1.4243	1.3751
10	0.8491	0.9250	1.3872	1.1070

Table 4: Performance for Dilated Rate = 2

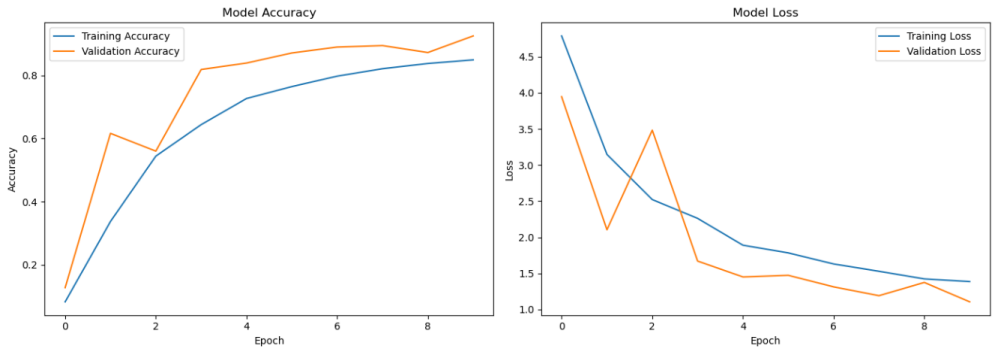


Figure 8: Training and Validation Accuracy for Dilated Rate = 2

6.4 Performance Comparison: Dilated Rate = 3

The training log for the enhanced model with a dilation rate of 3 is provided below. This configuration benefited from the largest receptive field, resulting in better generalization.

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	0.0227	0.0200	4.9284	4.1397
2	0.0280	0.0783	4.0612	4.0749
3	0.1128	0.3667	3.8989	3.1034
4	0.4059	0.6470	3.0379	2.0900
5	0.5974	0.7777	2.3073	1.5576
6	0.6833	0.8390	1.9195	1.2821
7	0.7373	0.8660	1.6748	1.2371
8	0.7729	0.8643	1.5512	1.1865
9	0.7977	0.8517	1.4367	1.4915
10	0.8152	0.8960	1.3864	1.2260

Table 5: Performance for Dilated Rate = 3

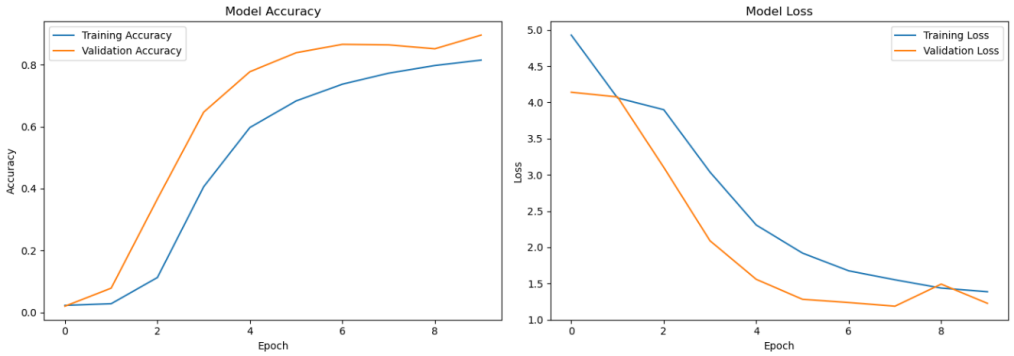


Figure 9: Training and Validation Accuracy for Dilated Rate = 3

6.5 Conclusion

The results indicate that varying the dilation rate has a notable impact on the model's ability to extract features and generalize effectively for Bengali handwritten character recognition.

For a dilation rate of **1**, the model exhibited steady improvements in training and validation accuracy, achieving a validation accuracy of **91.6%**. This configuration balances the depth of feature extraction and generalization ability. With a dilation rate of **2**, the model demonstrated slightly better feature abstraction, resulting in the highest validation accuracy of **92.5%**, while maintaining a lower validation loss, suggesting improved generalization and robustness. A dilation rate of **3** provided the largest receptive field. While this configuration allowed the model to capture a wider range of spatial features, it occasionally overfit the training data, reflected in a lower final validation accuracy of **89.6%** compared to other configurations.

The results highlight that a moderate dilation rate (**2**) achieves the best trade-off between feature abstraction and generalization for this dataset. Excessive dilation (rate = 3) may lead to diminished returns, particularly for datasets with smaller image dimensions, due to loss of fine-grained details. These insights underscore the importance of tuning the dilation rate based on the specific dataset characteristics and task requirements.

7 CNN Model with Squeeze-and-Excitation (SE) Blocks

The proposed model consists of the following components:

- **Input Layer**: The input is a 40x40 RGB image, suitable for small-sized datasets.
- **First Convolutional Block with SE**: A convolutional layer with 128 filters of size (3, 3) is followed by Batch Normalization and a Squeeze-and-Excitation (SE) block. The SE block enhances the model's focus on important features by re-weighting the feature maps.
- **Second Convolutional Block with SE**: A second convolutional layer with 64 filters (3, 3), Batch Normalization, SE block, max-pooling, and dropout layers.
- **Fully Connected Layers**: The output of the convolutional layers is flattened and passed through a dense layer with 128 units and dropout. The final softmax layer has 50 units, representing the character classes.

The Squeeze-and-Excitation block, which acts as a form of adaptive feature recalibration, is applied after each convolutional block. This block helps improve the model's capacity to focus on essential features, particularly important in Bengali script, where small nuances make a significant difference between characters.

The architecture can be summarized as follows:

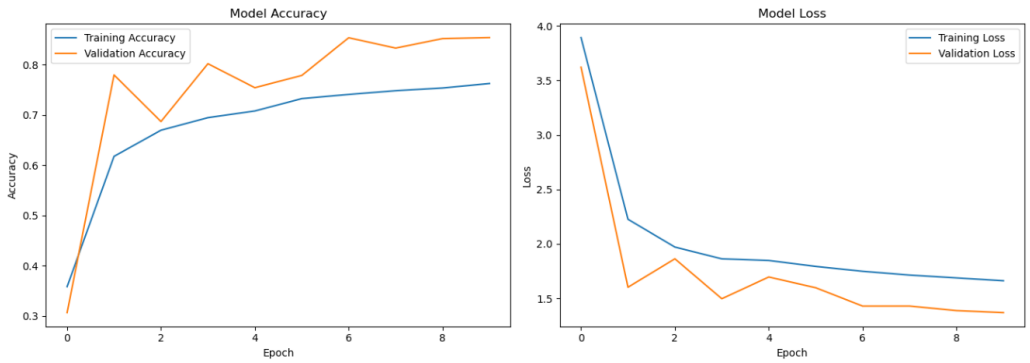


Figure 10: CNN Model with Squeeze-and-Excitation (SE) Blocks for Bengali Handwritten Character Recognition

7.1 Model Training

The model was trained using the Adam optimizer with categorical cross-entropy as the loss function. The dataset was augmented, and the training was carried out for 10 epochs. Below is a table summarizing the training and validation accuracy for each epoch.

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	0.3587	0.3070	3.8936	3.6218
2	0.6178	0.7797	2.2266	1.6028
3	0.6697	0.6870	1.9713	1.8636
4	0.6948	0.8020	1.8633	1.4974
5	0.7080	0.7543	1.8484	1.6967
6	0.7326	0.7787	1.7939	1.5983
7	0.7409	0.8537	1.7492	1.4300
8	0.7483	0.8330	1.7145	1.4302
9	0.7538	0.8520	1.6884	1.3885
10	0.7627	0.8540	1.6621	1.3696

Table 6: Training and Validation Results per Epoch for SE Block Model

7.2 Model Evaluation and Results

The model achieved a maximum validation accuracy of 83.73% on the test set after 10 epochs. The drop in validation accuracy in the later epochs could be due to overfitting.

7.3 Conclusion

The proposed CNN model with Squeeze-and-Excitation (SE) blocks successfully enhanced Bengali handwritten character recognition accuracy, demonstrating the effectiveness of the SE block in improving the model's focus on critical features. Future work can explore further optimizations or hybrid models combining CNNs with transformers.

8 Bengali Character Recognition using Pretrained Models

Bengali character recognition is an important task in the field of Optical Character Recognition (OCR). This study aims to explore the effectiveness of various pretrained models for Bengali handwritten character recognition. We used three popular models, namely MobileNetV2, DenseNet121, and VGG16, to evaluate their performance on a Bengali character dataset containing 12,000 training images and 3,000 test images across 50 classes.

8.1 Methodology

We used the following pretrained models for this study:

- **MobileNetV2**: A lightweight model designed for mobile and embedded vision applications. MobileNetV2 is known for its efficiency and is particularly well-suited for small datasets, making it an ideal choice for this study.
- **DenseNet121**: A densely connected convolutional network with improved gradient flow and fewer parameters. DenseNet models are designed to improve feature reuse and are effective for tasks requiring deeper networks.
- **VGG16**: A deep convolutional network known for its simplicity and effectiveness in image classification tasks. VGG16 has a relatively larger number of parameters compared to MobileNetV2 and DenseNet121, which can lead to overfitting on smaller datasets if not optimized properly.

We froze the base layers of each model and added custom layers to perform the character classification task. The models were trained using the same dataset, and the results were compared in terms of training accuracy, validation accuracy, and loss.

8.2 Results

The following table summarizes the performance of each model after 10 epochs of training:

Model	Training Accuracy		Validation Accuracy	
	Accuracy (%)	Loss	Accuracy (%)	Loss
MobileNetV2	73.8	0.8680	63.9	1.2938
DenseNet121	40.0	2.0693	56.4	1.5371
VGG16	48.08	1.7427	63.4	1.3023

Table 7: Performance Comparison of Pretrained Models for Bengali Character Recognition

8.3 MobileNetV2

MobileNetV2 showed the highest performance with a training accuracy of 73.8% and a validation accuracy of 63.9%. This model performed exceptionally well, likely due to its efficiency in handling small datasets. MobileNetV2 is designed to be computationally efficient with fewer parameters, which helps in reducing overfitting when working with limited data. Its lightweight architecture allows it to generalize well on small-scale datasets like the one used in this study, making it the most suitable choice for character recognition in Bengali.

8.4 DenseNet121

DenseNet121 achieved a training accuracy of 40.0% and a validation accuracy of 56.4%. Despite its ability to reuse features effectively, DenseNet121 showed lower performance compared to MobileNetV2. The lower accuracy is likely due to the fact that DenseNet121, with its deeper architecture and large number of parameters, requires a larger dataset to truly benefit from its dense connections. In this study, the limited size of the dataset may not have been sufficient to leverage the full potential of DenseNet121, leading to suboptimal results.

8.5 VGG16

VGG16 achieved a training accuracy of 48.08% and a validation accuracy of 63.4%. Although VGG16 is known for its simplicity and effectiveness, it showed a higher training loss compared to MobileNetV2. This suggests that VGG16 may be prone to overfitting when trained on small datasets due to its relatively larger number of parameters. However, its validation accuracy indicates that with proper regularization or fine-tuning, VGG16 can still perform well on the Bengali character recognition task.

8.6 Conclusion

Based on the results, MobileNetV2 proved to be the most effective model for Bengali character recognition, especially when working with small datasets. Its lightweight architecture and computational efficiency make it an ideal choice for tasks with limited data. DenseNet121 and VGG16, although capable models, require more data and optimization to fully demonstrate their potential. In future work, further fine-tuning and data augmentation techniques may improve the performance of these models on smaller datasets.

9 Overall Conclusion

This research presents a comparative study of various convolutional neural network (CNN) architectures and lightweight pretrained models for Bengali handwritten character recognition on a small dataset. The evaluation focused on standard CNN, residual neural networks (ResNet), dilated residual networks, models with Squeeze-and-Excitation (SE) blocks, and pretrained models such as MobileNetV2, DenseNet121, and VGG16.

The baseline CNN model showed a steady improvement, achieving a final validation accuracy of **87.90%**. However, its feature extraction capability was limited compared to more advanced models. The ResNet architecture, with residual connections, outperformed the standard CNN, achieving a final validation accuracy of **91.93%**.

Dilated residual networks enhanced performance further by capturing long-range dependencies. The model with **dilation rate 1** achieved a validation accuracy of **91.60%**, while the model with **dilation rate 2** achieved the highest performance with a validation accuracy of **92.50%** and a test accuracy of **84.91%**.

Incorporating Squeeze-and-Excitation (SE) blocks improved feature attention, leading to a final validation accuracy of **85.40%**, despite some fluctuation in validation accuracy during training.

Among the pretrained models, **MobileNetV2** showed the best performance, achieving a validation accuracy of **63.9%** and training accuracy of **73.8%**, making it suitable for small-scale character recognition tasks.

In conclusion, the study highlights the importance of choosing the right architecture for Bengali handwritten character recognition. Residual and dilated residual networks show the best performance, with dilated convolutions offering improvements in capturing the complexities of Bengali characters. SE blocks helped with feature discrimination, while MobileNetV2 showed good performance with a favorable balance between accuracy and efficiency.

References

1. *BornoNet: Bangla Handwritten Characters Recognition Using Convolutional Neural Network*. Rabby, A. S., Haque, S., Islam, S. M., Abujar, S., & Hossain, S. A.

2. Dataset source: https://rabby.dev/ekush/home.

3. Bangla Handwritten Character Dataset. Google Drive Repository : https://drive.google.com/drive/folders/1QRCpFwX4mc2EPlKUD-3MHwNSJYqQh$_1$*J*.

4. *BanglaNet: Bangla Handwritten Character Recognition using Ensembling of Convolutional Neural Network*, Published in: Saha, C., & Rahman, M. M.

5. *Bangla Handwritten Character Recognition Using Convolutional Neural Network with Data Augmentation* . Chowdhury, R. R., Hossain, M. S., Islam, R. U., Andersson, & Hossain. https://doi.org/10.1109/ICIEV.2019.8858545

6. *Bengali Handwritten Character Recognition Using Deep Convolutional Neural Network*,

   Purkaystha, B., Datta, T., & Islam, M. S.https://doi.org/10.1109/ICCITECHN.2017.8281853, Publisher: IEEE, Conference Location: Dhaka, Bangladesh