

TRANSFER LEARNING FOR PRINTED PASHTO ALPHABET RECOGNITION: A COMPARATIVE ANALYSIS

Mina Pashtana^{*1}, Abdul Basit², Azam Khan¹, Muhammad Asfand Yar Khan¹, Liaquat Ali¹,
Muhammad Saeed Kakar¹

^{*1}Department of Computer Science and Information Technology, University of Balochistan, Quetta, Pakistan

²Associate Professor, Department of Computer Science and Information Technology, University of Balochistan, Quetta, Pakistan. E-mail(s): drabasit@um.uob.edu.pk.

^{*1}kinnocent357@gmail.com

DOI: <https://doi.org/10.5281/zenodo.22931019>

Keywords

Pashto optical character recognition (OCR); transfer learning; convolutional neural networks (CNN); MobileNetV2; EfficientNetB0; ResNet50; low-resource language; printed character recognition.

Article History

Received: 01 June 2026

Accepted: 23 August 2026

Published: 24 September 2026

Copyright @Author

Corresponding Author: *

Mina Pashtana

Abstract

Optical character recognition (OCR) has made significant improvements thanks to deep learning, but scripts with few resources, like Pashto, are still not well covered in research. The Pashto script is written from right to left and has a cursive style. Also, some characters look very similar but are different because of small dots or marks. This makes it hard to recognize characters accurately. Most past studies have used custom-built convolutional neural network (CNN) models for recognizing Pashto script, but very few have compared common pretrained CNN models. This paper fills that gap by testing three pretrained CNN models—MobileNetV2, EfficientNetB0, and ResNet50—for recognizing printed Pashto letters across 44 different characters. All models were trained and tested under the same conditions, and their performance was measured using test accuracy, macro precision, macro recall, and macro F1-score. The results showed that ResNet50 performed the best, with a test accuracy of 87.39% and macro precision, recall, and F1-scores of 0.89, 0.87, and 0.87, respectively. This is better than MobileNetV2 (82.30%) and EfficientNetB0 (76.40%). ResNet50 also showed better generalization, performing best in both validation and test phases even though its training accuracy was lower than MobileNetV2. This suggests that ResNet50 learns more robust features for distinguishing similar Pashto characters. The study shows that choosing the right pretrained CNN model greatly affects recognition results for scripts with limited resources. It also shows that ResNet50 is the most effective model among those tested. This work sets up a clear benchmark for using transfer learning in printed Pashto character recognition and adds to the limited research on Pashto OCR.

INTRODUCTION

Optical character recognition (OCR) and handwritten character recognition (HCR) have made significant improvements in the last ten years, mostly because of deep learning methods used on well-supported writing systems like Latin, Arabic,

and Chinese. Convolutional neural networks (CNNs) are now the widely used method for these tasks because they can automatically learn important visual features from images without needing manual feature creation, which was common in older methods. Even with these advances, many regional

and low-resource languages are still not well covered in character recognition studies.

Pashto is one of these languages. It is spoken by about 50 million people in Pakistan and Afghanistan and is an official language in Afghanistan. It is used significantly in both spoken and written form. Pashto has some similarities with Arabic and Urdu, like a cursive, right-to-left script, but it also presents new challenges for recognition. Some Pashto characters look almost the same, differing only by small dots or marks, and the language has characters that are not found in Arabic or Urdu scripts. These traits make recognizing Pashto characters a challenging pattern recognition task. Also, the lack of large, standardized, and public datasets has slowed down research compared to increasingly commonly studied writing systems.

Early attempts at recognizing Pashto characters used traditional machine learning methods that needed manual feature extraction. More recently, deep learning methods have become increasingly common. Specialized CNN models made just for Pashto script have shown strong results on created datasets, but these models usually need significantly of design work and are only good for one specific dataset, which makes them less useful in other situations. At the same time, transfer learning has become a popular choice. This involves taking CNN models that were trained on large datasets like ImageNet and adjusting them for a new, usually smaller, task. This method saves time and data while often performing as well or better than custom models. Models like MobileNetV2, EfficientNetB0, and ResNet50 are commonly used here. Each model has its own balance between how much computation it needs and how accurate it is, and their performance can vary depending on the size of the dataset, how complex the images are, and the specific task.

Although transfer learning models have been widely tested in various image classification areas, there is not much research comparing them specifically for recognizing the Pashto alphabet. Most previous studies that focus on Pashto have mainly created one unique model, without checking how well different commonly used transfer learning models perform

against each other in this particular script. This lack of comparison is the main reason for this study.

This paper tackles this gap by testing and comparing the performance of three commonly used pretrained CNN models—MobileNetV2, EfficientNetB0, and ResNet50—for the task of classifying the Pashto alphabet. Each model is trained and tested under the same conditions, and their performance is measured using test accuracy, macro precision, macro recall, and macro F1-score, making the comparison between the models fair and straightforward. The goal is not just to find out which model works best for this task, but also to understand how the design of each model affects its ability to recognize characters in a visually challenging and less commonly used writing system. The rest of this paper is structured in the following way: the Literature Review section looks at existing research on Pashto character recognition and classification using transfer learning; the Methodology section explains the dataset used and the approach taken for the experiments; the Data Analysis and Results section shows the findings and compares the performance of the three models; and the Conclusion and Recommendations section highlights the key contributions of the study and suggests possible areas for future research.

Literature Review

A. Background Theory

Handwritten character recognition has improved significantly because of deep learning, but scripts with fewer resources, like Pashto, are still less extensively studied as bigger languages such as English, Arabic, and Chinese. Using transfer learning with pretrained models from ImageNet, such as MobileNetV2, ResNet50, and EfficientNetB0, is now a popular way to handle image classification when there's not much training data. Studies comparing different areas show mixed results, which depend on how big the dataset is and how complex the visuals are. This means the performance of these transfer learning models is not the same in every situation, so it is important to test them directly when working with new scripts or domains.

Previous Studies

Amin et al. (2020): To address the lack of resources for recognizing Pashto handwritten characters, the researchers created the “Poha” dataset, which includes 44 Pashto handwritten characters. They also developed a custom CNN model specifically designed for this script. The model was trained to identify the detailed features that differentiate visually similar Pashto characters, such as small dots and special marks. Their model achieved a test accuracy of 99.64%, which is better than several other deep learning models tested on the same dataset. This outcome shows that a well-designed CNN can perform much better than standard models when working with a script that has limited resources, and it positions “Poha” as an important tool for future studies.

Ullah et al. (2023): The research extended Pashto character recognition beyond just individual letters to include handwritten digits and special character combinations called ligatures, aiming to fill a larger gap in Pashto script recognition studies. To deal with the lack of labeled Pashto data, the authors created a dataset using a generative adversarial network (GAN) that included 60,000 digit images. This helped increase both the number and variety of training samples available. They tested how well standard CNN, LeNet, and Deep CNN models performed on this new dataset. The results showed that using GANs for data augmentation can significantly improve model training when there is not enough labeled data, providing a useful approach for scripts with limited resources like Pashto.

Khaliq et al. (2023): The study achieved better recognition results by using a customized CNN specifically designed for predicting character movement paths. They used the improved PHWD-V2 dataset as the basis for their experiments. Instead of using a standard pretrained model, the researchers built their network to handle the unique writing styles and stroke patterns found in Pashto script. This custom approach reached a training accuracy of 93.98% and a test accuracy of 92.99%, which is better than several commonly used deep learning models tested under the same conditions. These findings support a common observation in

Pashto script research: models designed with the script's specific features in mind generally perform better than general-purpose models.

Uddin et al. (2021): The researchers carried out a benchmark study on Pashto handwritten numerals, which has not been widely studied compared to research focused on the alphabet. They used a deep neural network with a rule-based activation function, which was created to better understand the unique visual features of Pashto numerals. This approach reached a recognition accuracy of 91.45%, supporting the idea that customizing the model's architecture usually gives better performance than using standard models for Pashto script. The study also provided a benchmark dataset that has been useful for later research on Pashto characters and numerals.

Fauzi and Haqdu D (2025): The study moved beyond focusing only on Pashto-specific literature to evaluate how well pretrained CNN models work in a different area. They tested MobileNetV2, EfficientNetB0, and ResNet50 for a traffic density classification task. The results showed that ResNet50 performed better than the other two models, achieving the highest validation accuracy and also showing better precision, recall, and F1-scores. This indicated that ResNet50's deeper residual structure gave it an advantage when dealing with complex visual patterns. These findings are also important for other tasks beyond traffic classification, as they show that the effectiveness of these three common architectures depends significantly on the visual complexity of the task.

Hajar et al. (2025): In another study, a similar benefit was seen for ResNet50 in a completely different area. The research focused on classifying leaves using ecoprint techniques and compared the same group of pretrained models used for transfer learning. The results again showed that ResNet50 performed better, and the researchers pointed out that this success was mainly because of its residual connections. These connections helped reduce the vanishing gradient problem, which usually makes it hard to train very deep networks. This result supported the pattern seen in other areas, where ResNet50's deep structure allowed it to learn more detailed and useful features during fine-tuning.

Along with the traffic density study, this work further supports the idea that ResNet50 might be especially good at handling complex visual classification tasks in various applications.

Poudel et al. (2025): In a Nepali Sign Language character recognition task, the opposite result was found, with MobileNetV2 performing better than ResNet50, even though ResNet50 usually does better in other areas. The researchers said this was because lighter models like MobileNetV2 are better suited for smaller datasets with limited resources. They explained that deeper models like ResNet50 might struggle with overfitting or need more data to work effectively. This is significant because it shows there's no one-size-fits-all model that works best in every situation. Instead, the performance of a model depends on the specific task and the type of data it is working with.

Siddiqui et al. (2023): The study tackled the ongoing issue of limited datasets in Pashto script research by creating a baseline database of individual printed characters that includes all forty-four Pashto letters in various font styles. Unlike previous work that mainly focused on handwritten text, this effort specifically aimed at printed character recognition, addressing a key gap in the resources available for Pashto script research. The resulting dataset established one of the first clear and standardized benchmarks for recognizing isolated printed Pashto characters, supporting earlier datasets that focused

on handwritten text like “Poha” and PHWD-V2. This dataset is directly relevant to the current study, which also examines transfer learning models using isolated printed Pashto alphabet images, making it a crucial resource for the present comparative analysis.

Conceptual Framework

Previous research shows two main approaches: creating custom CNN models for Pashto script can be very accurate, but they need significantly of time and effort to design. On the other hand, using transfer learning with pretrained models like MobileNetV2, EfficientNetB0, and ResNet50 is faster and more flexible, but how well they work can depend on the task. However, not many studies have carefully compared these standard transfer learning models specifically for recognizing Pashto letters. This study fills that gap by testing these three models under the same experimental conditions, expanding the existing transfer learning comparisons to include the Pashto script.

Methodology

This part introduces the suggested transfer learning approach for recognizing printed Pashto alphabet characters. The framework is divided into four main steps: preparing the dataset, processing the images, building the model using transfer learning, and assessing its performance.

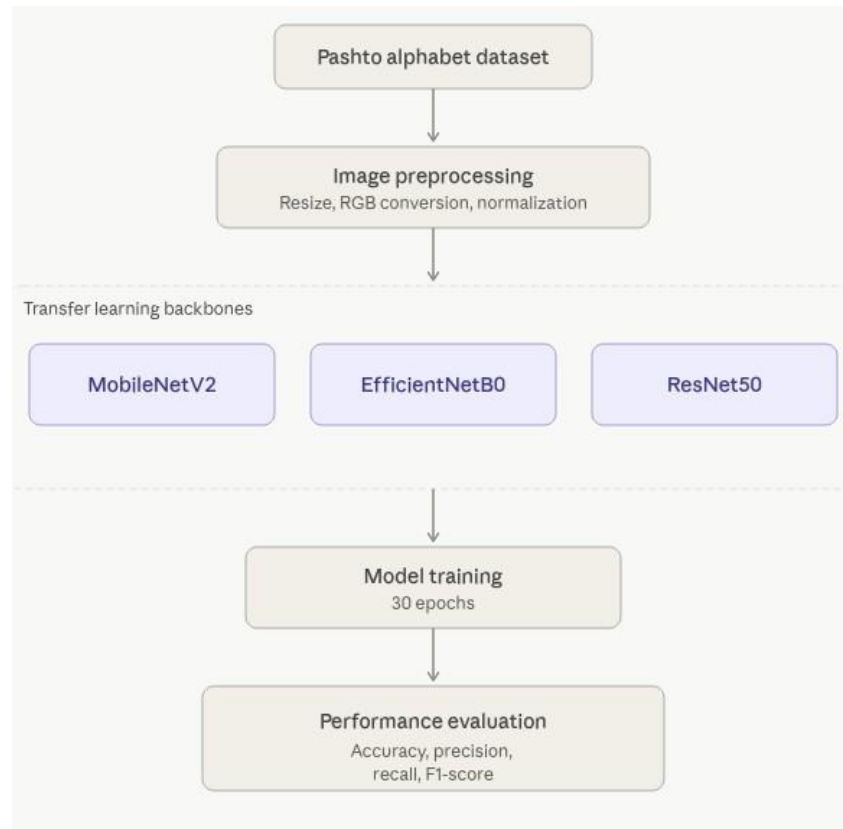


Fig. 1. The proposed method for recognizing the Pashto alphabet: preparing the input data, applying transfer learning with MobileNetV2, EfficientNetB0, and ResNet50, then training and evaluating each model.

A. DataSet

The experiments used a printed Pashto alphabet dataset that included 44 different character types. The original images were in grayscale and had a resolution that was not suitable for use with pretrained deep convolutional neural networks. To fix this, each image was resized to 224 by 224 pixels, which is the standard size needed for models like MobileNetV2, EfficientNetB0, and ResNet50. Since these pretrained models require three-channel RGB input, each grayscale image was converted into a three-channel format by duplicating the single grayscale channel across all three channels. The pixel values were then scaled to the range [0, 1] to help the model train more effectively and reach strong results faster. To check how well the models performed on new data, the dataset was split into three parts: training, validation, and testing. The same steps were followed for all images in every experiment to keep everything consistent.

B. Model Development

Instead of building deep CNNs from the beginning, this work uses transfer learning. This method involves using models that were previously trained on the large ImageNet dataset and then adjusting them for recognizing the Pashto alphabet. Transfer learning helps the networks use the detailed visual features learned from millions of natural images, which significantly cuts down the number of images needed to get strong results. Three commonly used CNN structures were chosen because they have different strengths: MobileNetV2 is good for being light and not using too much computing power; EfficientNetB0 is better at using parameters efficiently through a special scaling method; and ResNet50 uses residual learning to make it easier to train deeper networks effectively.

For each architecture, the original ImageNet classification layer was taken out and replaced with

a custom classifier. This new classifier included a global average pooling layer, a dropout layer to help prevent overfitting, and a fully connected softmax layer with 44 neurons. The 44 neurons matched the number of classes in the Pashto alphabet. During

training, only the parameters of this new classification layer were updated, while the pretrained convolutional part of the model stayed the same and was used as a fixed feature extractor.

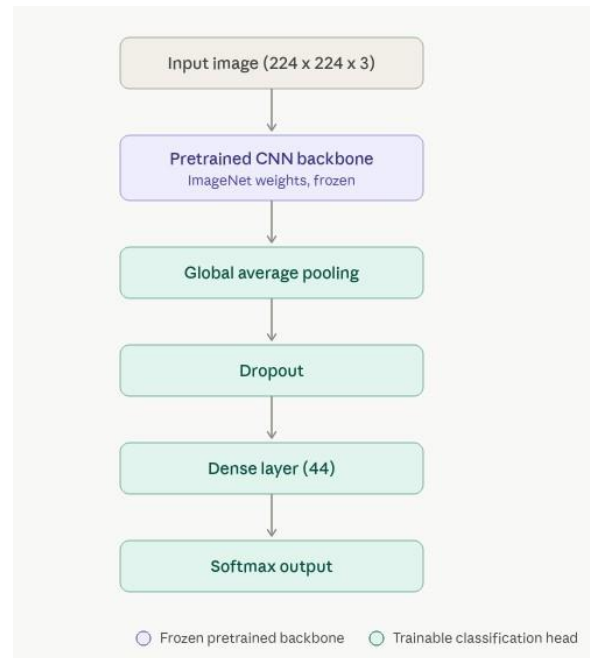


Fig. 2. Transfer learning architecture adopted for MobileNetV2, EfficientNetB0, and ResNet50.

C. Training Methodology

All the experiments were carried out using the TensorFlow/Keras deep learning framework. Each model was trained with the Adam optimizer, starting with a learning rate of 0.001. Since the class labels were given as integers instead of one-hot encoded vectors, the sparse categorical cross-entropy loss function was used. All three CNN models were trained for 30 epochs with the same hyperparameters to make the comparison fair. A learning rate scheduler called ReduceLROnPlateau was used to lower the learning rate automatically when the validation loss stopped improving for several epochs. This helped the models converge more steadily and avoid getting stuck in poor local minima. To make sure the performance differences were due to the model designs and not training setup changes, the same preprocessing steps, optimizer, learning rate, loss function, and training method were used for all three models.

The models were tested on a separate set of data that was not used during training or when tuning the model. Their performance was measured using four common metrics for multi-class classification: accuracy, precision, recall, and F1-score. Accuracy shows how many samples were correctly classified overall. Precision tells us how many of the samples that were predicted to be in a certain category were actually correct. Recall measures how well the model identifies all the samples that truly belong to each category. The F1-score, which is the average of precision and recall, gives a fairer picture of the model's performance, especially when some classes are easier to predict than others. Along with these numbers, detailed reports were used to look at how well the model performed for each character class, helping to spot which characters were consistently recognized and which ones were hard to tell apart because they look similar in Pashto script.

Data Analysis And Results

A. Results

The three transfer learning architectures exhibited distinct learning characteristics during the training process.

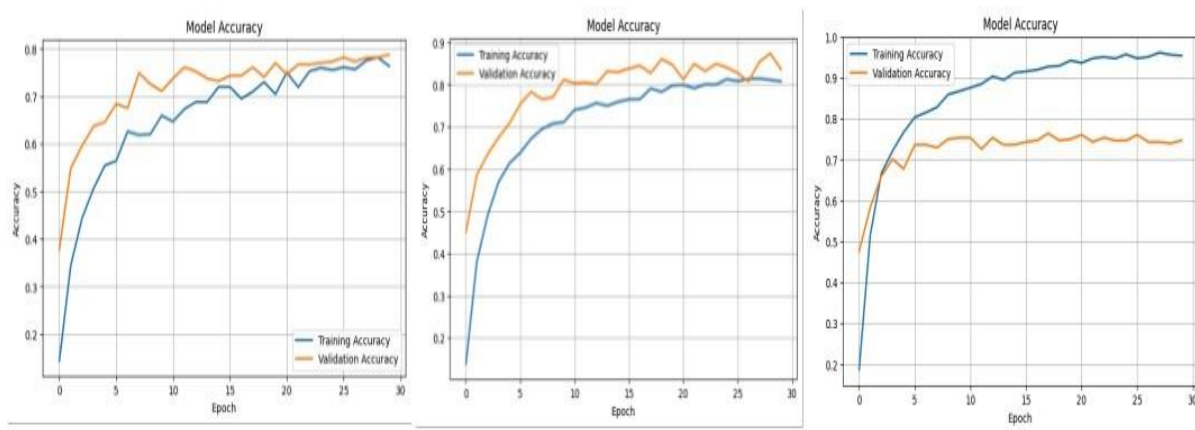


Fig. 3. Training and validation accuracy curves of MobileNetV2, EfficientNetB0, and ResNet50 during training.

MobileNetV2 showed the quickest learning among the models tested. During training, the accuracy rose quickly from 19.24% in the first round to nearly 95% by the end. Meanwhile, the validation accuracy saw significant improvements early on but then leveled off around 77 to 78%. Even though the learning rate was adjusted several times using the

adaptive scheduler, there was not much gain later in training, which means the model had probably reached its maximum learning potential. The big difference between how well it performed on training data versus new, unseen data suggests it might be overfitting—memorizing the training examples instead of learning to generalize.

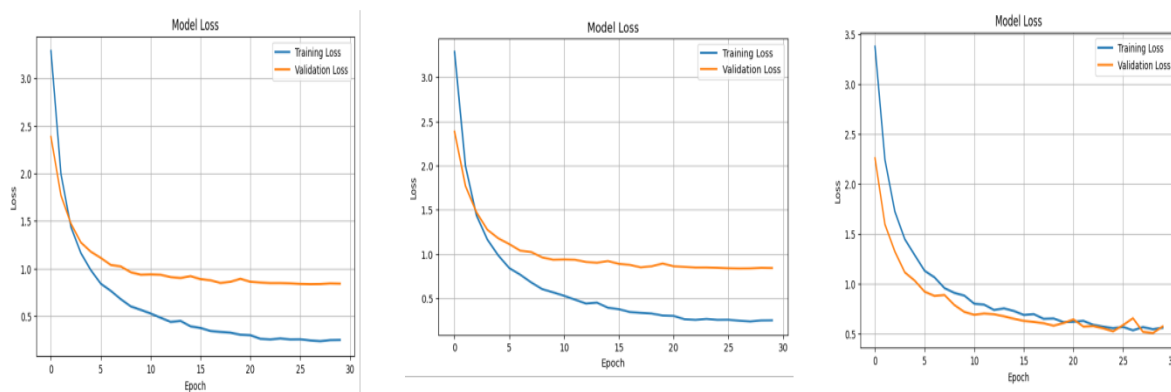


Fig. 4. Training and validation loss curves of MobileNetV2, EfficientNetB0, and ResNet50 during training.

EfficientNetB0 showed a slower learning process. Both the training and validation accuracy kept going up steadily during training without big ups and downs, showing consistent optimization. The difference between the training and validation results stayed small, which means the model generalized well. But both lines leveled off close to 78%, showing the model hit a performance limit even though it kept improving. Because of this, EfficientNetB0 had the worst recognition results compared to the other models tested.

ResNet50 showed a different way of learning. Even though its training accuracy grew slowly and stayed around 81% by the end of training, its validation accuracy kept getting better over time and reached

about 87% in the last few training stages. This suggests that the model learned more effective features while still being good at generalizing to new data. The small gap between training and validation results shows that ResNet50 did not overfit too much and had the most consistent learning performance compared to the other models.

Table 1 shows how well the three transfer learning models performed on the test set. Out of the models tested, ResNet50 had the best recognition accuracy with 87.39%, which is higher than MobileNetV2 at 82.30% and EfficientNetB0 at 76.40%. The same order was seen in macro precision, recall, and F1-score, which means ResNet50 is the most effective at classification.

Table 1. Performance comparison of the evaluated transfer learning architectures

Model	Test Accuracy	Precision	Recall	F1-score
MobileNetV2	82.30%	0.85	0.82	0.82
EfficientNetB0	76.40%	0.79	0.76	0.76
ResNet50	87.39%	0.89	0.87	0.87

Table 2 shows how well the three different models performed during training, validation, and testing. MobileNetV2 had the best performance during training with 95.2% accuracy, but its lower results on validation and testing suggest it might be overfitting. On the other hand, ResNet50 had the

highest accuracy on validation and testing, even though it did not perform as well during training, which means it generalized better to new, unseen data. EfficientNetB0 showed more consistent learning across all stages but had the lowest overall recognition results.

Table 2. Comparison of training, validation, and test accuracies for the evaluated architecture

Model	Training Accuracy	Best Validation Accuracy	Test Accuracy
ResNet50	80.8%	87.39%	87.39%
MobileNetV2	95.2%	77.78%	82.30%
EfficientNetB0	78.2%	78.76%	76.40%

B. Analysis

The results of the experiment show that the type of pretrained CNN used has a big effect on recognizing printed Pashto letters, even when all models are trained in the same way. Among the different models tested, ResNet50 performed the best overall, achieving the highest accuracy on both validation and test data, as well as the best macro precision, recall, and F1-score. Even though its training

accuracy (80.8%) was lower than MobileNetV2, its better performance on new, unseen data suggests it generalizes better and learns more useful features rather than just memorizing the training data. ResNet50's strong performance is because of its residual learning design, which allows for deeper feature extraction and helps prevent the vanishing gradient problem. This is especially helpful for

printed Pashto characters, where many letters are very similar and differ only by small dots or marks. In contrast, MobileNetV2 learned quickly and reached the highest training accuracy (95.2%), but its lower performance on validation and test data showed a bigger gap in generalization, indicating a tendency to overfit. Despite this, MobileNetV2 still provided good recognition accuracy and had a simple structure, making it a suitable choice for OCR tasks on devices with limited resources. EfficientNetB0 performed the worst among the models tested. Even though its training and validation results stayed fairly steady, it hit a performance limit during training and consistently scored lower than MobileNetV2 and ResNet50. This suggests that its default settings might need improvement or that more training data could help it better learn features for this script.

Overall, the results support a key point: having a high accuracy during training does not always mean a model will perform better in real-world situations. The real test for a model's effectiveness is how well it can handle new, unseen data. For challenging scripts like Pashto, which are both visually complex and have limited resources, models that can learn more detailed and distinct features, such as ResNet50, seem to work better for accurate character recognition. One thing to note is that all three models were tested with the same setup and data split, so changing factors like hyperparameters, data enhancement methods, or using more training data might affect how the models compare. Also, the study only looked at printed Pashto characters, so the results might not apply directly to handwritten text or to recognizing words and ligatures as whole units.

Conclusion And Recommendations

A. Conclusion

This study looked at and compared three commonly used pretrained convolutional neural network (CNN) models—MobileNetV2, EfficientNetB0, and ResNet50—for recognizing printed Pashto letters using transfer learning. All models were trained and tested under the same setup, using test accuracy, macro precision, macro recall, and macro F1-score to ensure a fair and consistent comparison. The

results showed clear differences in how well each model performed. ResNet50 had the best overall performance, with a test accuracy of 87.39% and macro precision, recall, and F1-scores of 0.89, 0.87, and 0.87 respectively. While MobileNetV2 had the highest training accuracy, ResNet50 consistently had the best validation and test results, showing it generalizes better to new data. On the other hand, MobileNetV2 offered a good balance between efficiency and performance, making it good for applications with limited resources. EfficientNetB0 performed the worst under the conditions of this study. The findings show that the choice of pretrained CNN model greatly affects the recognition of printed Pashto letters. Better generalization, rather than just high training accuracy, is a better sign of effective performance, especially for visually complex and low-resource scripts like Pashto. This study sets a systematic benchmark for widely used pretrained CNN models in printed Pashto character recognition and shows that ResNet50 is the most effective among the ones tested, giving a strong starting point for future Pashto OCR research.

B. Recommendation

Future studies should look into testing more pretrained models, increasing the dataset with bigger and more varied examples of printed and handwritten Pashto characters, and looking into better data enhancement and attention-based learning methods. Researchers and practitioners are encouraged to build on the existing framework by moving from recognizing individual characters to recognizing whole words and ligatures in Pashto OCR. They should also explore ways to make models lighter and more efficient, so they can work well on devices with limited resources and mobile platforms, helping with real-world applications for digitizing regional languages.

Data Availability

The Pashto alphabet dataset used in this study can be obtained from the corresponding author upon request, provided it is reasonable. There may be some limitations on accessing specific data because related research is still in progress.

Conflict Of Interest

The authors state that there is no conflict of interest.

Funding Source

None.

Acknowledgment

The authors would like to thank the Department of Computer Science and Information Technology at the University of Balochistan in Quetta, Pakistan, for their support of this research.

REFERENCES

- Amin, M. S., Yasir, S. M., & Ahn, H. (2020). Recognition of Pashto handwritten characters based on deep learning. *Sensors*, 20*(20), 5884. <https://doi.org/10.3390/s20205884>
- Ullah, M., Arslan, M. M., Abdullah, Young, X., & Othman, N. Q. (2023). Pashto handwritten character recognition based on generative adversarial network (GAN) and deep learning. *SSRN*. <https://doi.org/10.2139/ssrn.4333780>
- Khaliq, F., Shabir, M., Khan, I., Ahmad, S., Usman, M., Zubair, M., & Huda, S. (2023). Pashto handwritten invariant character trajectory prediction using a customized deep learning technique. *Sensors*, 23*(13), 6060. <https://doi.org/10.3390/s23136060>
- Uddin, I., Ramli, D. A., Khan, A., Bangash, J. I., Fayyaz, N., Khan, A., Kundi, M., & Khan, A. (2021). Benchmark Pashto handwritten character dataset and Pashto object character recognition (OCR) using deep neural network with rule activation function. *Complexity*, 2021*, Article 6669672. <https://doi.org/10.1155/2021/6669672>
- Fauzi, D. R., & Haqdu D, G. A. (2025). Comparison of CNN models using EfficientNetB0, MobileNetV2, and ResNet50 for traffic density with transfer learning. *Journal of Intelligent Systems Technology and Informatics*, 1*(1), 22–30. <https://doi.org/10.64878/jistics.v1i1.6>
- Hajar, S., Murinto, M., & Yudhana, A. (2025). Comparison of transfer learning strategies using MobileNetV2 and ResNet50 for ecoprint leaf classification. *Jurnal Teknik Informatika (JUTIF)*, 6*(5), 3251–3264.
- Poudel, B., Ghimire, S., Bhattarai, S., Bhandari, S., & Dahal, S. S. (2025). Nepali sign language characters recognition: Dataset development and deep learning approaches. *arXiv*. <https://arxiv.org/abs/2510.11243>
- Siddiqui, A., Basit, A., Noor, W., Khan, M. A., Kakar, M. S. H., & Khan, A. (2023). Baseline isolated printed text image database for Pashto script recognition. *Intelligent Automation & Soft Computing*, 37*(1), 875–885. <https://doi.org/10.32604/iasc.2023.036426>