Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-*378***

A CNN Braille Character Recognition System for Empowering Visually Impaired Children through Enhanced Parental Involvement and Interactive Learning

Dr. Dalvin vinoth Kumar A[1] , Dr. Stephen R[2] , Arya Anil[1] , Royce Rose K RV[1]
Department of statistics and Data Science[1]
Department of Computer Science[2]
Christ University

To Cite this Article
Dr. Dalvin vinoth Kumar A , Dr. Stephen R , Arya Anil , Royce Rose K RV**," A CNN Braille Character Recognition System for Empowering Visually Impaired Children through Enhanced Parental Involvement and Interactive"** ***Musik In Bayern, Vol. 90, Issue 2, Feb 2025, pp13-30***

Article Info
Received: 26-01-2025 Revised: 02-02-2025 Accepted: 10-02-2025 Published: 25-02-2025

Abstract

Early onset of severe, irreversible vision loss in young children can lead to lifelong impacts, including delays in motor skills, language, emotional, social, and cognitive growth. Vision-impaired children of school-going age may also face challenges in achieving educational success. Given that visual learning comprises more than 85% of a child's educational experience, any impairment in their visual comprehension could significantly impact their academic performance. Furthermore, visual impairment in children prevents their parents from being part of their child's educational journey. Studies have shown parental involvement is crucial to a child's academic success. In light of these challenges, In this paper a Braille Recognition System using Convolutional Neural Networks for accurately recognizing Braille Alphabets. An interactive learning system for visually impaired kids where they would be guided by their parents, enhancing the parent-child bond.
Keywords: Braille Character Recognition, Deep Learning for Assistive Technology , Convolutional Neural Networks (CNNs), BrailleNet Architecture , Visually Impaired Children Education, Optimization of Deep Learning Models, Adam vs. Nadam Optimizer , Hyperparameter Tuning in CNNs, Activation Functions (ELU, ReLU, GELU, SELU, Leaky ReLU)

1. Introduction

The International Agency of Blindness, an estimated 450 million children globally require treatment for vision-related conditions, with approximately 90 million children experiencing some form of sight impairment. Visually impaired kids face a lot of challenges in terms of

education. Unlike children with sight, these kids cannot visualize the concepts. This slows down the learning process. Moreover, parents of Blind kids find it hard to participate in their child's educational journey due to a lack of knowledge in Braille. It's not easy to study the intricate patterns of Braille. Studies have shown that Parental involvement in their child's education has long-lasting consequences. Parental involvement in early childhood education is linked to enhanced academic aspirations, increased school completion rates, and improved overall well-being among children. Active parental involvement, including assisting with academic tasks at home, reading together, aiding with homework, and utilizing teacher-recommended resources, significantly contributes to children's educational development[1]. Henderson and Berla concluded that the strongest determinant of a student's academic success is not their socioeconomic background or income, but "The extent to which that student's family is able to: (a) create a home environment that encourages learning; (b) communicate high, yet reasonable, expectations for their children's achievement and future careers; and (c) become involved in their children's education at school and in the community"[2].
These challenges must be addressed to promote inclusivity in education for all children. To do this, we have created an interactive system for Braille learning for Blind kids. This interactive system would aid the child's overall cognitive development by making learning fun and letting parents become part of their child's academic journey. Braille is a tactile writing system comprising patterns of raised dots used by blind and visually impaired individuals to read, enabling access to written information through touch. Deep learning has significantly advanced the field of image recognition. For detecting Braille characters from images, we utilized Convolutional Neural Networks (CNNs). The CNN architecture was chosen using rigorous trial and error. To identify the most suitable model for our dataset, we evaluated three models: ResNet18, VGG16, and Inception. All these models were modified to tailor to the data. The modified inception model was suitable for the Braille Dataset and gave the highest accuracy. This modified version, named BrailleNet, featured reduced size in convolutional layers, lesser number of denser layers and inception modules. With these adaptations, BrailleNet was efficient in recognizing Braille characters from A to Z. The model was further evaluated based on various metrics to ensure it performed well in classifying the Braille characters. The model parameters were fine-tuned to ensure maximum efficiency.

2. Literature Survey

This study[1] introduces an innovative method for automatic recognition of Braille characters. The authors have used several pre-processing techniques for image alignment and enhancement. After reviewing several state-of-the-art CNN architectures, they have modified the original CNN architecture by replacing few of the modules with an inverted residual block, which in turn has less computational cost. This is how they have introduced novelty in their work. They achieved an impressive accuracy of 95.2% using the English Braille dataset. This paper gives an excellent overview of the various CNN architectures and the advantages and disadvantages of each. The authors discuss the two main challenges of CNN: increasing accuracy and decreasing computational complexity. For rectifying the alignment issues, PCA has been used. From their work, it's clear that image processing can help increase a model's accuracy. The image processing techniques used here are Wiener filtering, thresholding, and morphological operations. The authors have used VGG16, ResNet50, DenseNet201, and InceptionV3 as the backbone of the CNN models. Depending on the specific model, the final few modules are substituted with IRB modules. The different processes to be done before modeling that have helped increase the accuracy are: Correcting alignment using PCA, Data Augmentation(rotation,

Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-*378***
width height, height shift, brightness correction), and image downsampling(bilinear interpolation). The overall findings indicate that the model exhibited superior performance, particularly when accounting for all image preprocessing, augmentation, and downsampling techniques. The model achieved an overall F score of 95.2%. Among the various models used for experimentation, DenseNet had less computational cost when incorporated with the IRB modules. The most important thing to note is how the authors have reduced the computational costs by employing lightweight CNN models.

In this paper[12], the authors have used the Adaptive Bezier Curve Network (ABCNet) to detect word-based text from natural scenes. They have implemented a character-based translator for the Braille language. Here, they have implemented three versions of ResNet - ResNet-18, ResNet-34, and ResNet-50. They have performed different processing techniques like augmentation and normalization, and resized the images. They set the learning rate as 0.001 and each model has 20 epochs of iterations. The loss function used was the cross-entropy loss function and the optimizer for models was Adam optimization algorithm. ResNet-18 has the highest performance metrics values. The F1 values of ResNet-34 and ResNet50 are lower than that of ResNet-18. ResNet-18 has better performance compared to the deeper networks and this is because compared to a shallow model, the deeper model would have more parameters. ResNet-50 and ResNet-18 have 2,355k parameters and 1,118k parameters respectively.

This paper[16] introduces a novel deep learning model designed for optical Braille character detection and word recognition from embossed Braille documents. The proposed model combines both a convolutional neural network and a transformer. The former is utilized for character recognition, while the latter is employed for word recognition. The proposed model was evaluated on datasets of Braille documents considering various factors like the height of each dot, quality of the paper, and lighting conditions. Mask R-CNN and YOLOv2 were the techniques employed to achieve the result. The R-CNN will first identify the region of interest for potential bounding boxes and then run a classifier on these regions. The results show a high detection accuracy for Braille characters and for word recognition. These accuracies have surpassed all the inaccuracies of the existing techniques. This study primarily focuses on enhancing optical Braille character detection and word recognition.

This paper[19] uses the HOMUS database. This database consists of 15200 samples of handwritten music notation. They were produced by 100 musicians. The samples are divided into 32 classes. Each class represents a different musical symbol. The authors have compared three Convolutional neural networks: LeNet, AlexNet, and GoogleNet. The LeNet is known for solving the MNIST dataset. AlexNet and GoogleNet have gained high results in classifying many other images. These three networks were compared based on different parameters. During the training, the dataset was divided into training set, validation set, and testing set. The LeNet got an accuracy of 79.51%, and AlexNet with an accuracy of 95.35% presented better results than the LeNet. However, the best result was achieved by using GoogleNet. It achieved an overall accuracy of 96.46%. The proposed methodology can be used for solving complex tasks like adding new symbols, etc.

This paper[17] proposes a system, Eyeris, which aims to aid visually impaired people without needing a guardian. The proposed system detects and recognizes objects, individuals, and signs in the frame of the user's vision and notifies them regarding the same. This proposed system also

ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-*378***
considers the communication between blind and deaf people by even transmitting sign language to blind people through audio. Objects can be detected using the SSD model (Single Shot MultiBox Detector); it is one of the fastest yet highly accurate models. The architecture of SSD is built on the VGG-16 network, thus enabling feature extraction at multiple scales. The object detection model is trained using COCO (Common Object in Context) Dataset. The network architecture for image and face recognition is built using ResNet-34. The reason for using the ResNet model to recognize faces is due to the very low error rate. ResNet possesses sufficient depth to extract intricate and easily discernible features, enabling the mapping of complex functions. Also, ResNet helps to achieve high precision in the results. The proposed system also alerts the Guardian if it detects threats like fire. Thus, the system also ensures people's safety and makes them independent.

3. Proposed Methodology

The proposed system has a CNN architecture that accurately recognizes the Braille alphabet. To train the model for the same, it must be trained on Braille images. The dataset was trained on three models: Resnet18, VGG16 and Inception. All these models were fine tuned to tailor to the dataset. The proposed BrailleNet model to improve the prediction accuracy of the braille images. input image for Inception is 224*224*3 and the dimensions of the input image Braille characters are 28*28*3, we had to customize the architecture accordingly. Fig 1 shows the architecture of BrailleNet. In the figure, labels such as A, B, C, till J, correspond to different layers of the proposed model. These labels denote the transformations applied to the image as it progresses through the network, capturing the feature extraction process at each convolutional layer.

A - Conv layer with 64 filters and a (3,3) kernel
B - MaxPool layer with (3,3) window and (2,2) strides
C - Conv Layer with 192 filters and a (3,3) kernel
D - MaxPool layer with (3,3) window and (2,2) strides
E - INCEPTION MODULE I
F - INCEPTION MODULE II
G - Average Pooling layer with (5,5) window and (1,1) strides
H - Fully Connected Layer Elu Activation Function
I - Dropout Layer with 0.4 dropout rate
J - Output Layer with 26 units, Softmax Function

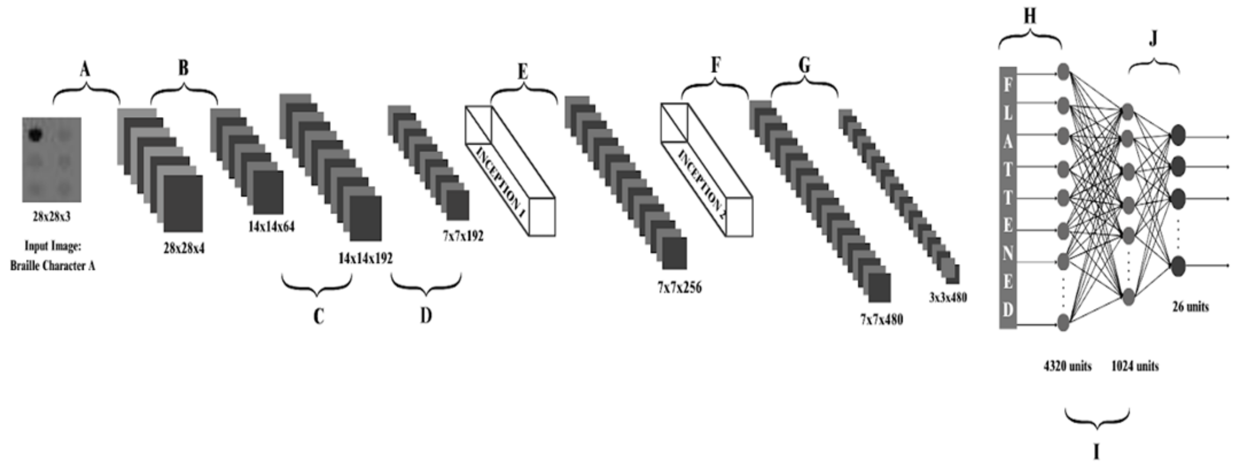


Fig 1 BrailleNet Architecture

This section will cover the descriptions of the Data used and the overall architecture of the customized model BrailleNet.

3.1 Data Collection and Preprocessing

The dataset considered for train and test the proposed model is same as used in Deep Learning Strategy for Braille Character Recognition [22] The dataset contains a total of 1560 images. Each image is an exemplary representation of the braille character from A to Z. The dataset consists of all 26 braille characters, and each character has undergone three different augmentations: width, height shift, Rotation, and brightness. For each augmentation type, 20 images are available. Hence in total, there are 60 images for each character. Hence there are 26 different classes, and the distribution of classes within the dataset is balanced (60 each), with an equal proportion of each character class, thereby mitigating potential class imbalance challenges. The images are in 3D array format, with each pixel in the RGB color space. In this format, each pixel is characterized by three formats, corresponding to intensities of red, blue and green channels. The dataset encapsulates real-world variations in Braille representation, accounting for different brightness or alignment variations. By encompassing this variation, our recognition system is primed to excel in real-world scenarios where Braille characters may exhibit diverse characteristics. The Braille Character Dataset forms the cornerstone of our Braille character recognition system. It provides the necessary groundwork for our machine learning algorithms to grasp the complex intricacies, relationships, and unique features that define Braille characters. Table 1 shows three architectures commonly used in Braille character recognition namely VGG16, ResNet18, and Inception. Out of these, Inception architecture was computationally efficient and had the highest validation accuracy. All the architectures were modified to tailor to the Braille Dataset. From table 3.1 it's clear that Inception has the maximum accuracy.

Table 1 Accuracies of VGG16, ResNet18 and Inception

CNN Model	Training Accuracy (%)	Validation Accuracy(%)
VGG 16	*98.7*	*90.7*
ResNet18	*94.47*	*90.38*

Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2025-378

Inception	***97.03***	***94.23***

The Inception architecture involves stacking multiple Inception modules on top of each other. Each module is a unit that processes and extracts features from the input data[21]. One striking feature of the Inception module is that it performs parallel convolution operations with filters of different sizes (1x1, 3x3, 5x5). 1x1 convolutions capture information at point-wise level. 3x3 convolutions capture local features and 5x5 convolutions capture global features. At each parallel operation, 1x1 convolutions are used for dimensionality reduction. This reduces the depth by decreasing the number of channels and hence making the process computationally efficient, at the same time capturing essential features. The max Pooling Layer will capture the most important features from each of these parallel operations. The features from these operations are stacked along the depth dimensions i.e., they will be stacked on top of each other. It provides a more comprehensive understanding of the input since it incorporates features from various scales and levels of abstraction. The comparison of proposed architecture is listed in the table 2. It compares various aspects of BrailleNet, a custom CNN architecture designed for Braille character recognition, with the original GoogleNet (Inception-v1) architecture. BrailleNet features an input size of (28, 28, 3) compared to the larger input size typically used in GoogleNet. Additionally, BrailleNet includes specific design choices such as a smaller initial convolution layer, limited use of pooling layers, and a single dense layer tailored for the recognition of Braille characters.

Table 2 Comparison between BrailleNet and Original GoogleNet

Aspect	**BrailleNet**	**GoogleNet (Inception-v1)**
Input Size	(28, 28, 3)	Larger (usually 224x224x3)
Initial Conv Layer	(3x3, 64 filters)	(7x7, 64 filters)
Inception Modules	Two modules	Multiple modules with complex branches
Pooling Layers	Limited use	Multiple pooling layers
Fully Connected	1 dense layer (1024 units)	Several dense layers
Dropout Layer	Included (0.4 rate)	Not mentioned
Output Layer	26 units (Braille chars)	1000 units (ImageNet classes)

4. Results and Discussion

Experimental Setup

To expedite the training process and harness the power of parallel processing, we used Google Colab. This cloud-based Python notebook environment provided access to high-performance GPUs. This decision was crucial to enhance the computational efficiency of our deep learning model, which is inherently resource-intensive. The proposed deep learning model developed using Tensorflow and Keras framework. systematically refined our hyperparameters by employing a methodical approach involving trial and error built using the Sequential API in Keras.

The experiment was conducted with various different combinations of hyperparameters to optimize BrailleNet. The hyperparameter considers activation function, learning rate and optimizer. The training, validation and testing accuracy are captured for varied epochs with adam and Nadam optimizers as shown in table 3 & 4. It is observed that Adam measured 99.4% and 97.7% as training and validation accuracy at 100 epochs. The results for adam and Nandam optimizer with varied epochs are shown in figure 2 and 3.

Table 3 Accuracy and Loss for Adam Optimizer at Different epochs

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
20	97.5	93.5	0.0765	0.272
40	98.9	95.8	0.03	0.18
60	99.2	95.1	0.022	0.1567
100	99.4	97.7	0.06	0.12

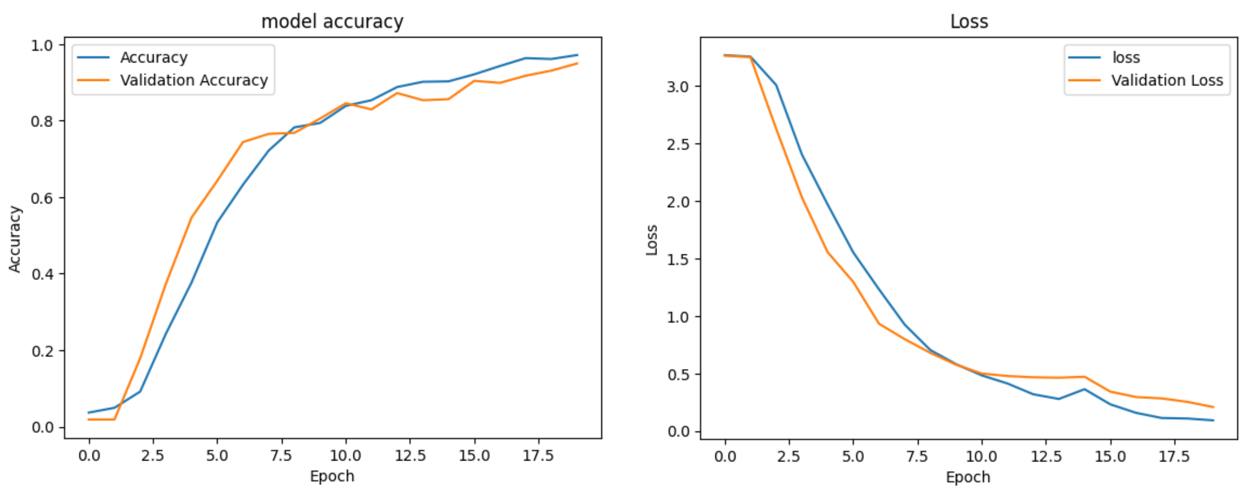
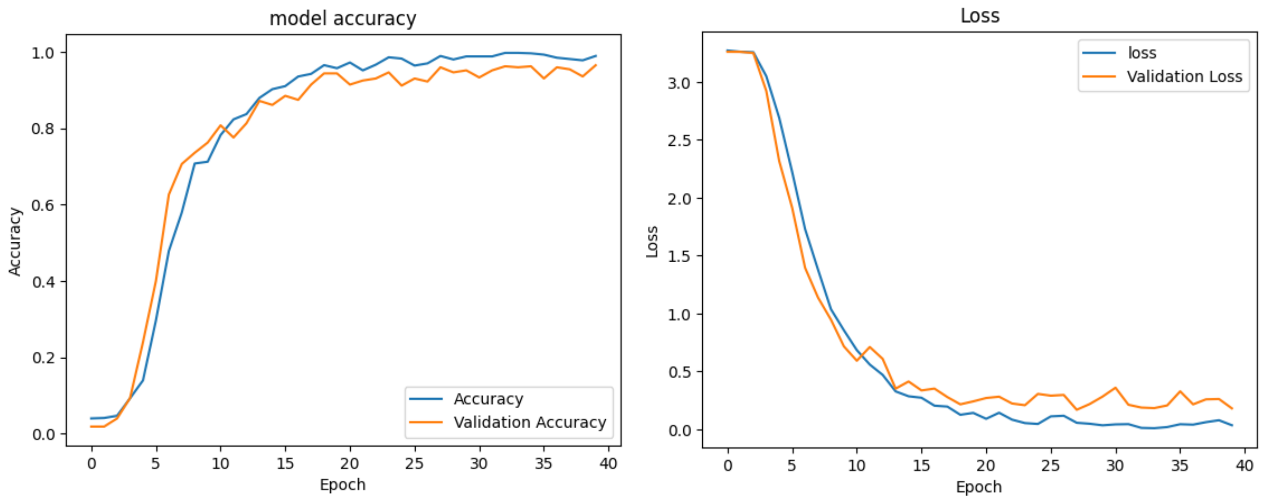


Fig 2a Accuracy and Loss graphs for BrailleNet using Adam Optimizer at epoch 20



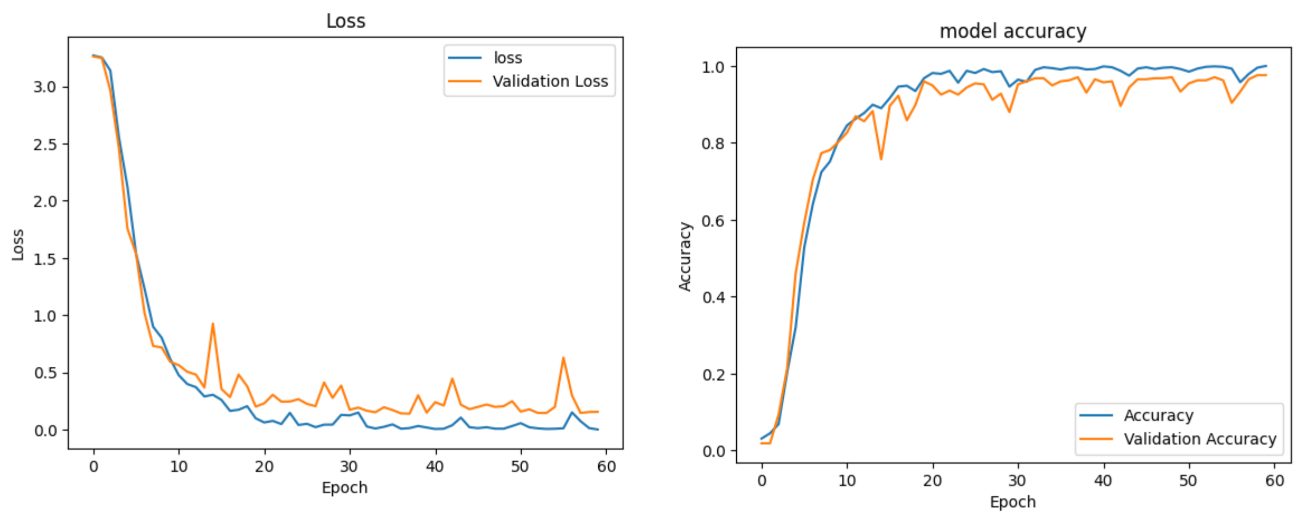


Fig 2b Accuracy and Loss graphs for BrailleNet using Adam Optimizer at epoch 40

Fig 2c Accuracy and Loss graphs for BrailleNet using Adam Optimizer at epoch 60

Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-378**

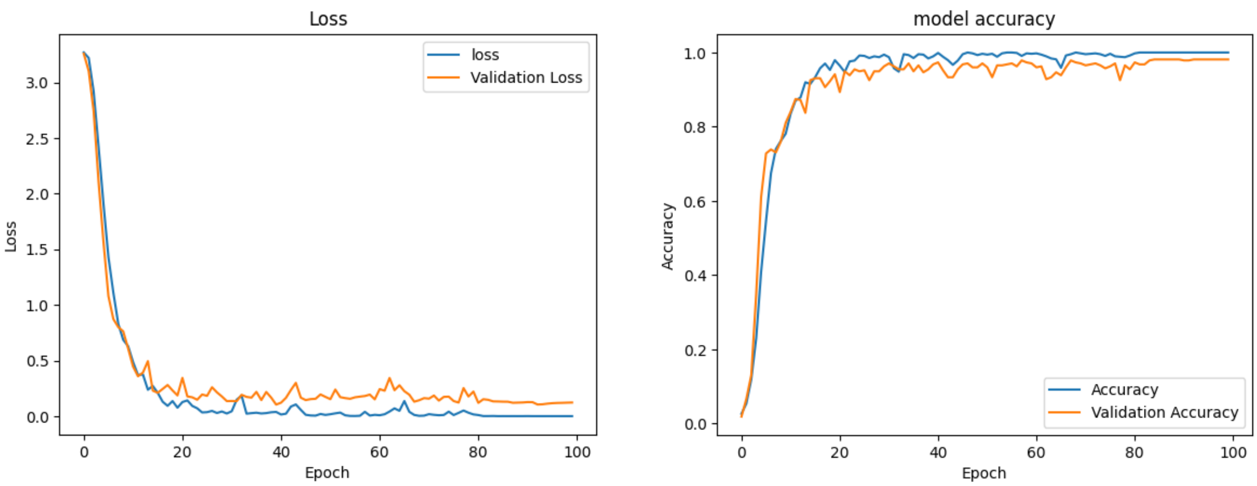
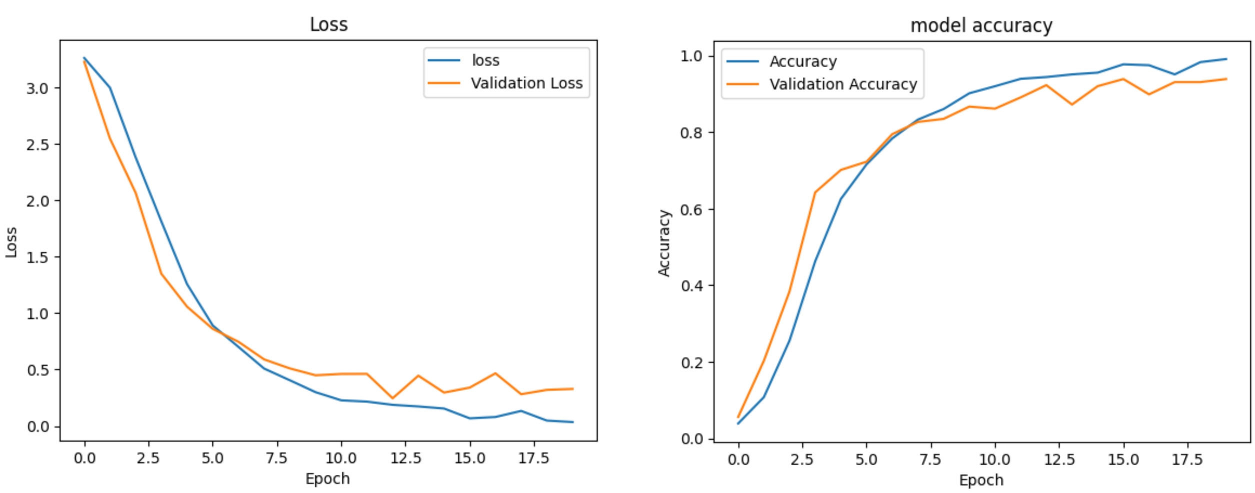


Fig 2d Accuracy and Loss graphs for BrailleNet using Adam Optimizer at epoch 100

Optimizer: Nadam

Table 4 Accuracy and Loss for NAdam Optimizer at Different epochs

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
20	97.43	94.23	0.1114	0.2630
40	99.19	97.43	0.0365	0.1064
60	99.28	98.71	0.0460	0.0533
100	99.52	96.47	0.0403	0.1303



ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-378**

Fig 3a Accuracy and Loss graphs for BrailleNet using Nadam Optimizer at epoch 20

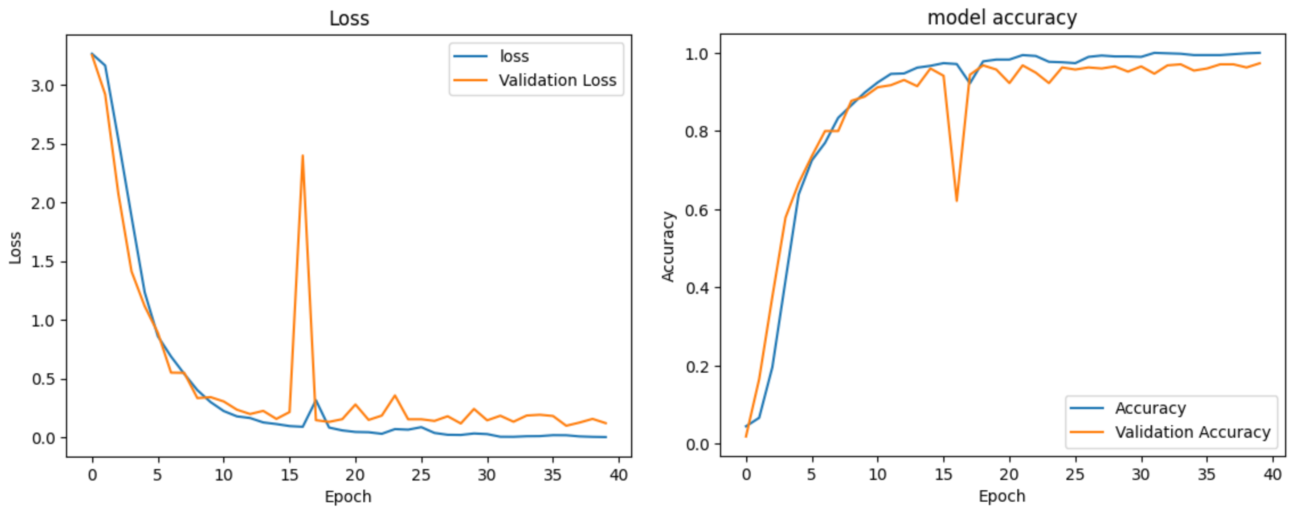


Fig 3b Accuracy and Loss graphs for BrailleNet using Nadam Optimizer at epoch 40

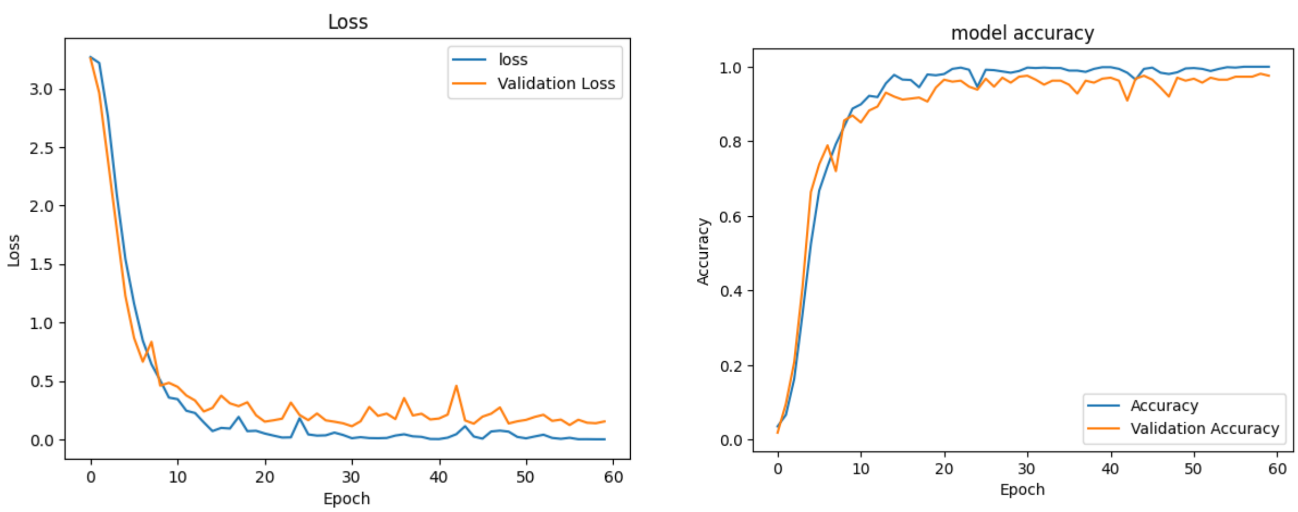


Fig 3c Accuracy and Loss graphs for BrailleNet using Nadam Optimizer at epoch 60

Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2025-378

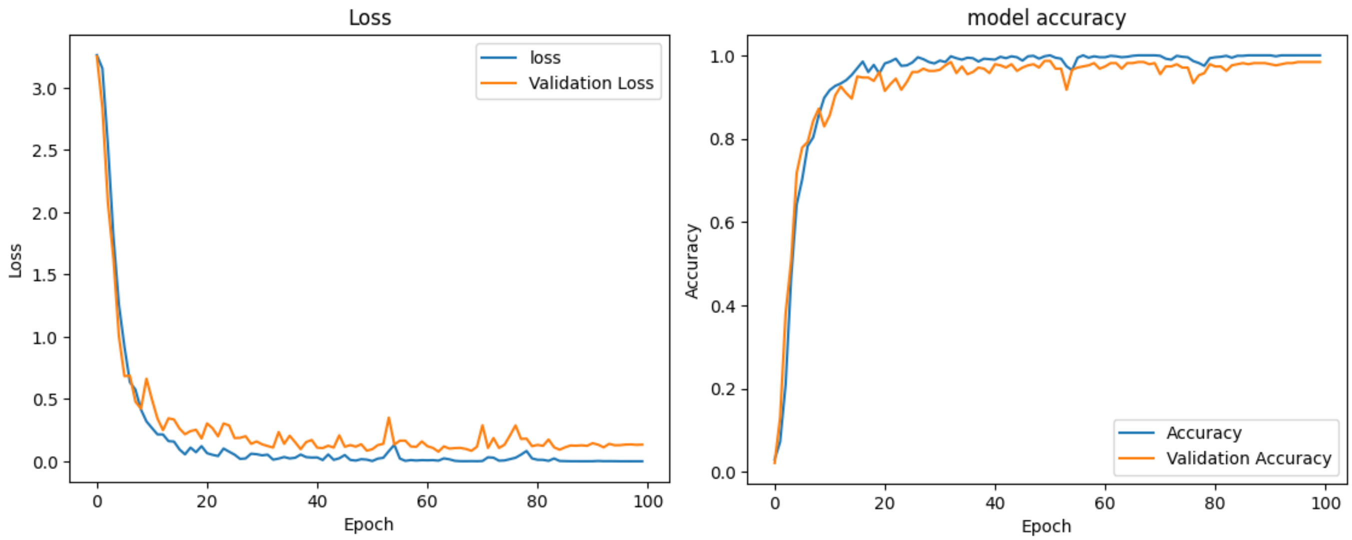


Fig 4.8 Accuracy and Loss graphs for BrailleNet using Nadam Optimizer at epoch 100

Since it seems like Nadam can cause the model to overfit. Therefore we chose Adam as the ***Fig 4***
Fig. 3d Accuracy and Loss graphs for BrailleNet using Nadam Optimizer at epoch 100

Activation Function in the Dense Layer:

To prevent overfitting and improve the efficiency of the training process, early stopping techniques are employed. Two early stopping callbacks were utilized during training each with a patience of 20. The early stopping callbacks monitored the validation performance of the model, and training was halted if no improvement was observed for a predefined number of epochs - here 20, determined by the "patience" parameter.

Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-378**

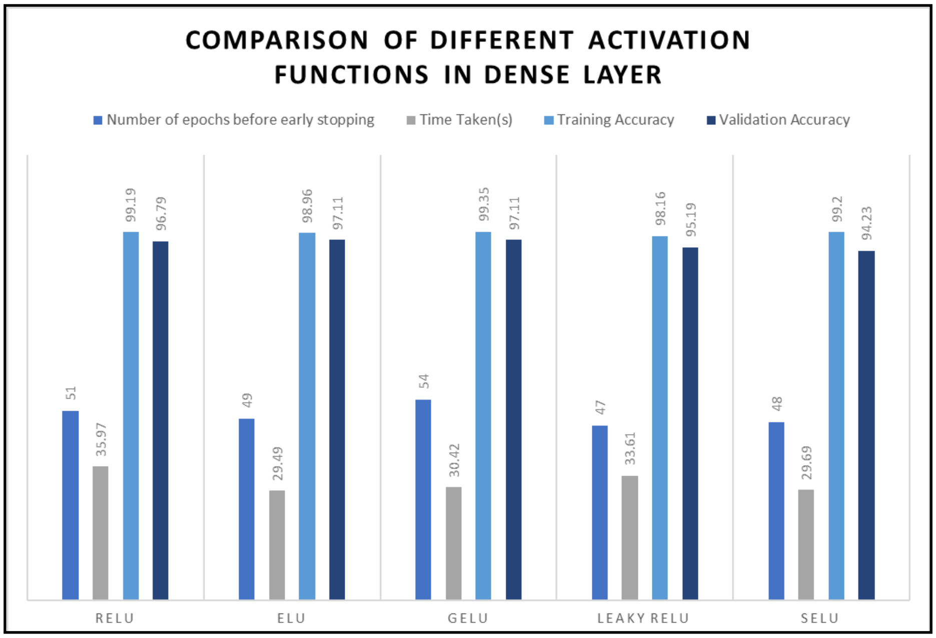


Fig 4 Comparison of different Activation functions in Dense Layer

Taking into account all factors, such as the number of epochs, training time, training accuracy, and validation accuracy, from Figure 4.9, it is evident that ELU yields the best performance.

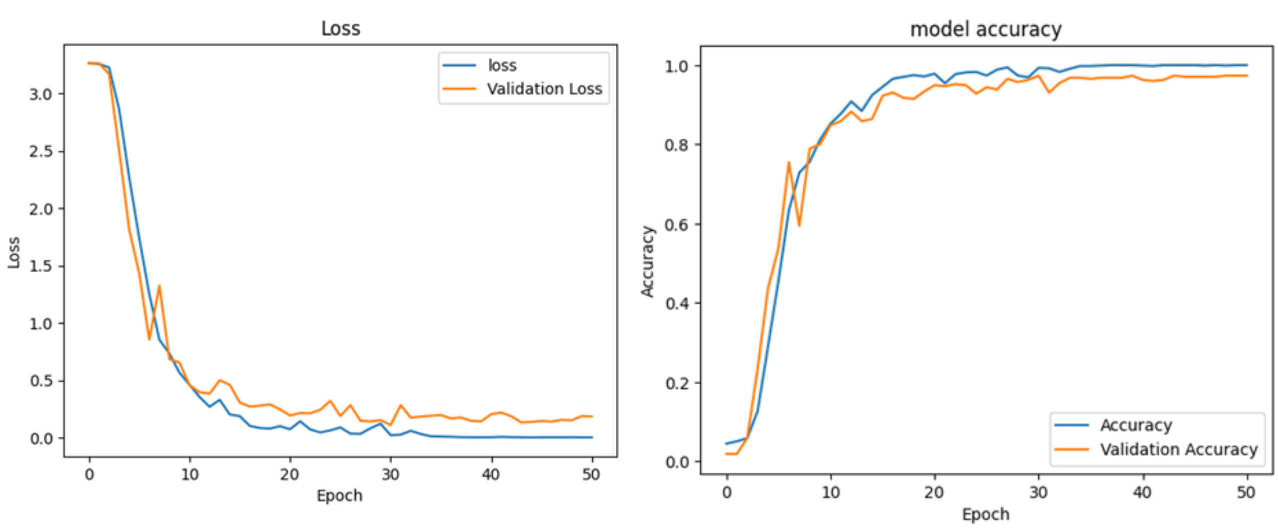


Fig 5a Accuracy and Loss graphs for BrailleNet using ReLu Activation function at epoch 51

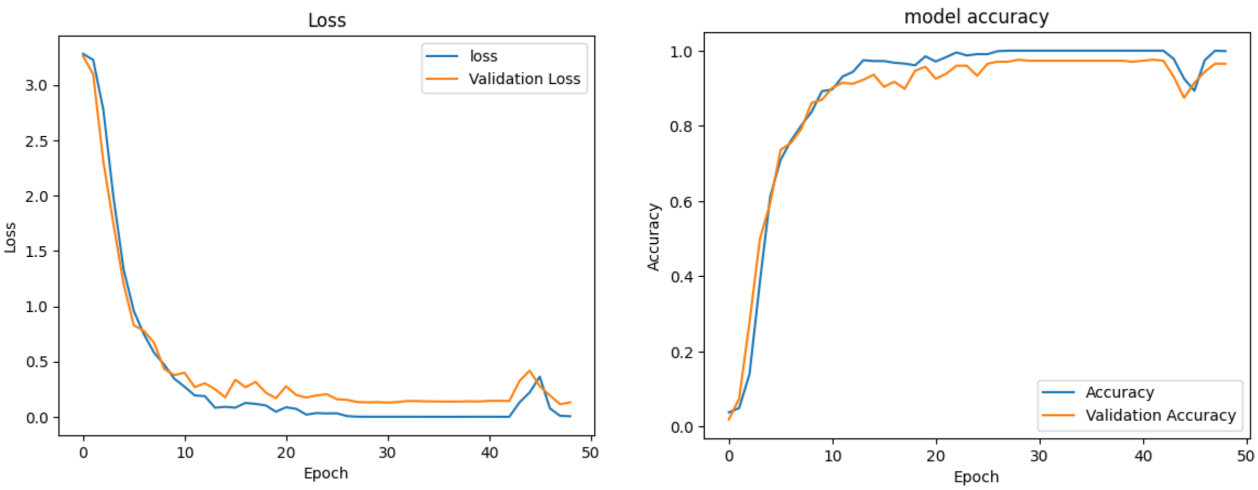


Fig 5b Accuracy and Loss graphs for BrailleNet using Elu Activation function at epoch 49

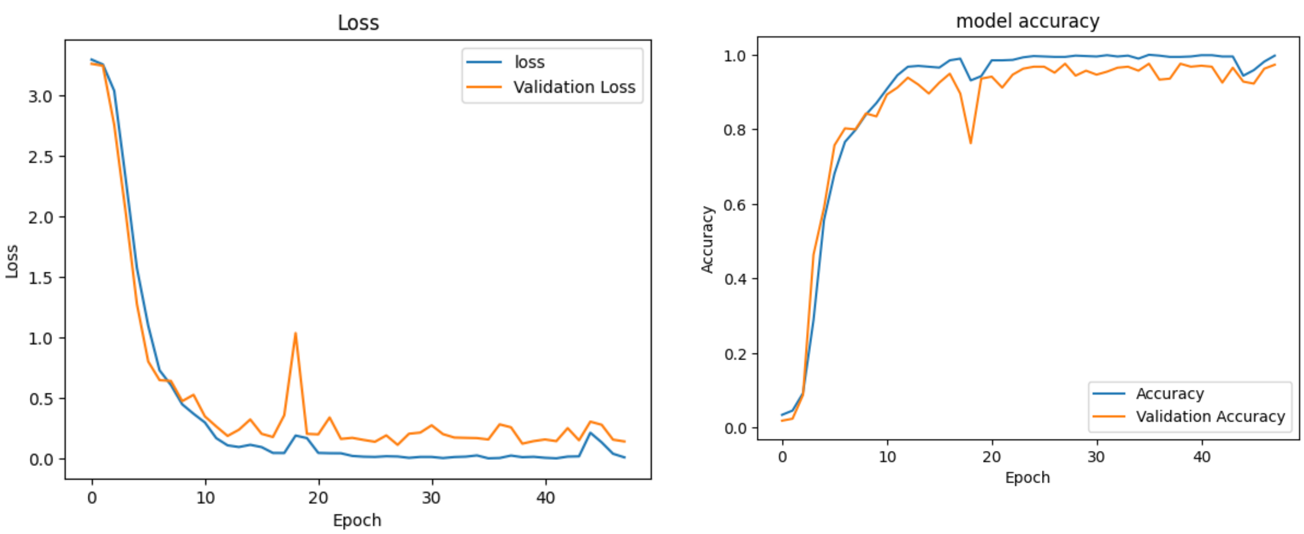


Fig 5c Accuracy and Loss graphs for BrailleNet using Gelu Activation function at epoch 54

Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-378**

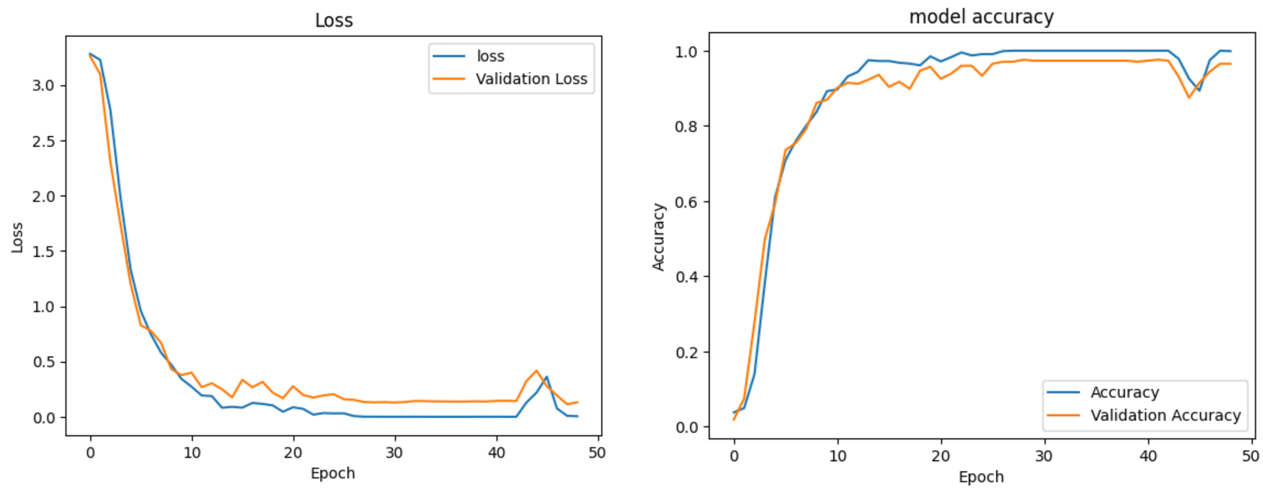


Fig 5d Accuracy and Loss graphs for BrailleNet using Leaky Relu Activation function at epoch 47

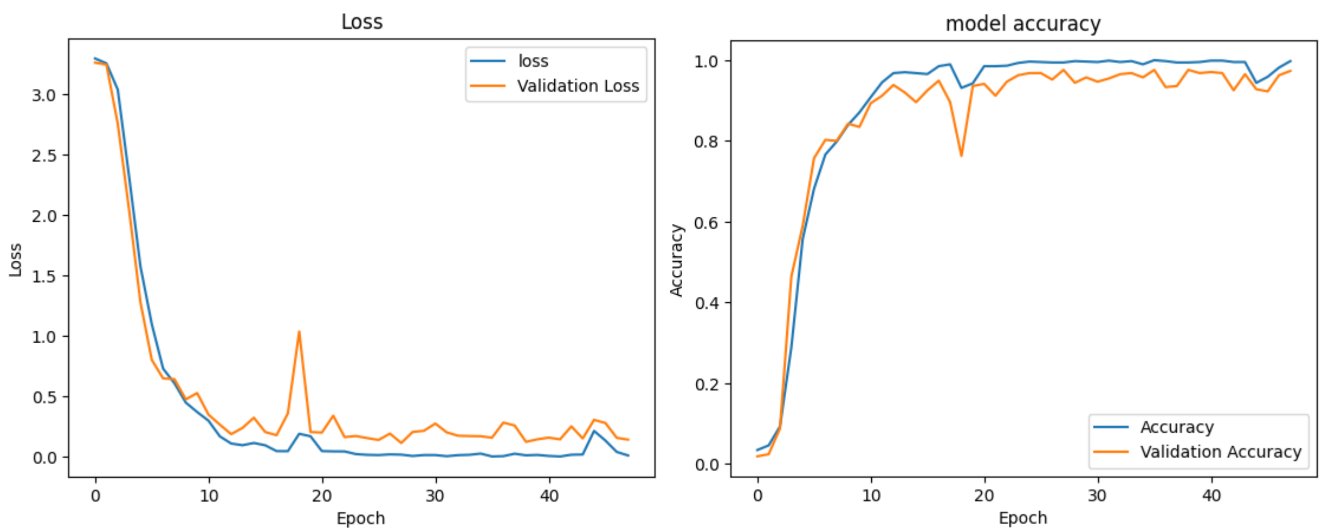


Fig 5e Accuracy and Loss graphs for BrailleNet using Selu Activation function at epoch 48

Learning Rate:

Different learning rate with elu activation function:

Table 4.3 Comparison of different Learning Rates

Learning	Epochs	Time	Training	Validation	Training	Validation

Musik in bayern
ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-378**

rate			Accuray	Accuracy	Loss	Loss
0.001	48	27.69s	99.12	97.11	0.0334	0.0700
0.0001	100	57.134s	98.63	93.59	0.0594	0.1899
0.01	23	14.10s	3.92	3.52	3.2858	3.3079

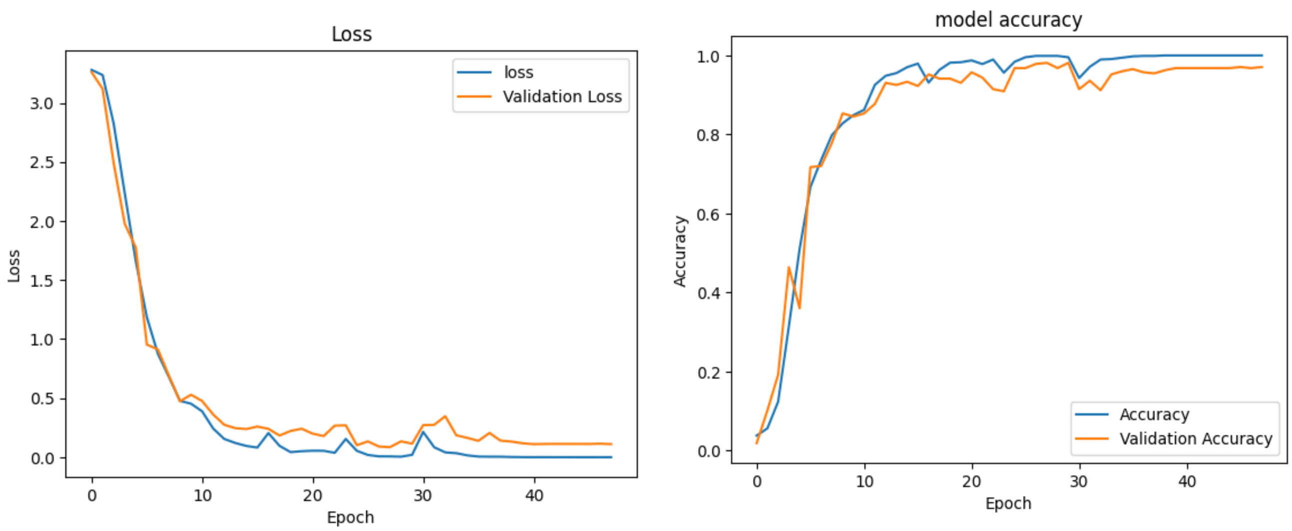
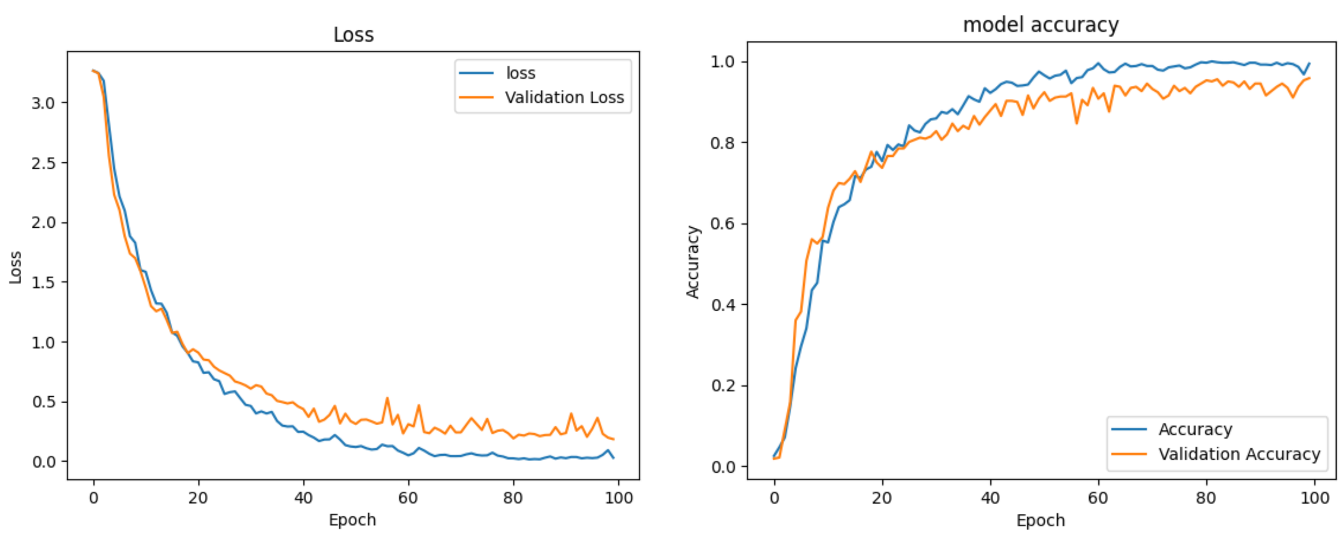


Fig 6a Accuracy and Loss graphs for BrailleNet using Elu Activation function with learning rate - 0.001



ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2025-378**

Fig 6b Accuracy and Loss graphs for BrailleNet using Elu Activation function with learning rate - 0.0001

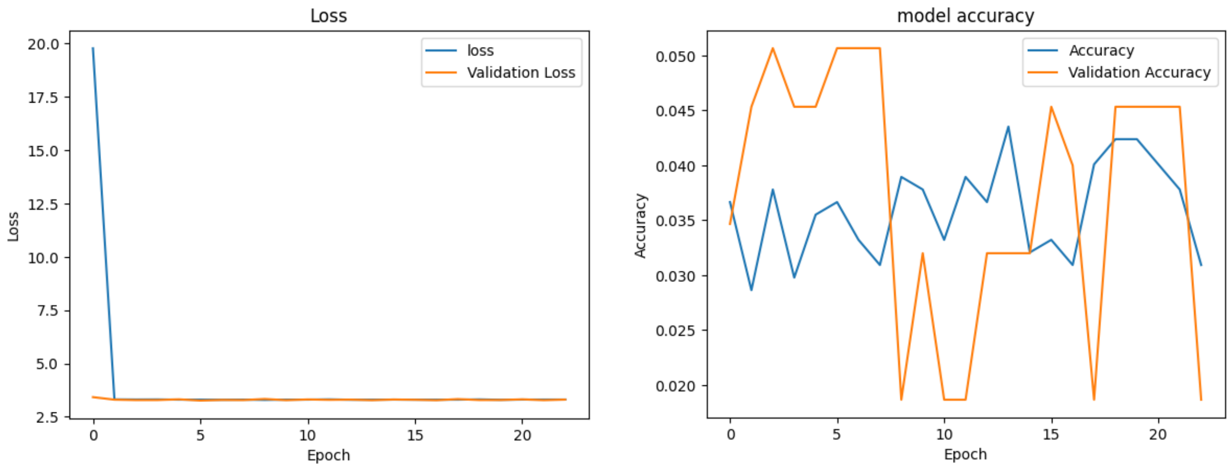


Fig 6c Accuracy and Loss graphs for BrailleNet using Elu Activation function with learning rate - 0.01

From these results it's clear that BrailleNet works best for Adam Optimizer at Learning rate 0.001 and the activation function in the Dense Layer should be Elu.

5. Conclusion

In essence, our Braille learning system is a catalyst for societal change. By breaking down educational barriers, empowering visually impaired children and promoting inclusivity, we strive to create a future where every child, regardless of their visual abilities will have equal access to quality education and the opportunity to unlock their full potential. To achieve this goal, we customized a CNN architecture for accurate Braille character recognition. The CNN architecture we used was Inception. The original Inception architecture was customized and this model was called BrailleNet. BrailleNet architecture has two Inception blocks. For BrailleNet, we got a validation accuracy of 94%. We did some trial and error to determine the best BrailleNet hyperparameters. We can conclude the best optimizer here is Adam. Two early stopping were applied to prevent the model from overfitting. We also found that the best learning rate is 0.001 and the Elu as the activation function in the Dense Layer gives maximum efficiency. Hence, here we designed a model that can accurately recognize Braille characters that can be used to build interactive learning tools for Blind kids and promote inclusivity in society. The future scope would be to extend the system's capabilities to recognize and teach Braille characters beyond the English alphabet, encompassing various languages and scripts. This would broaden the system's accessibility to a more diverse group of visually impaired children globally.

6. References:

[1] M. Đurišić and M. Bunijevac, "Parental involvement as a important factor for successful education," *Center for Educational Policy Studies Journal*, vol. 7, no. 3, pp. 137–153, 2017. doi:10.26529/cepsj.291
[2] X. Ma, J. Shen, H. Y. Krenn, S. Hu, and J. Yuan, "A meta-analysis of the relationship between learning outcomes and parental involvement during early childhood education and early elementary

ISSN: 0937-583x Volume 90, Issue 2 (Feb -2025)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2025-*378*
education," *Educational Psychology Review*, vol. 28, no. 4, pp. 771–801, 2015. doi:10.1007/s10648-015-9351-1

[3] K. A. Toussaint and J. H. Tiger, "Teaching early braille literacy skills within a stimulus equivalence paradigm to children with degenerative visual impairments," *Journal of Applied Behavior Analysis*, vol. 43, no. 2, pp. 181–194, 2010. doi:10.1901/jaba.2010.43-181

[4] S. Susanto and D. S. Nanda, *Teaching and learning english for visually impaired students: an ethnographic case study*, 2020. doi:10.35542/osf.io/jy26f

[5] L. Johnson, "The Braille Literacy Crisis for Children," *Journal of Visual Impairment & Blindness*, vol. 90, no. 3, pp. 276–278, 1996. doi:10.1177/0145482x9609000323

[6] J. A. Taquía Gutiérreza, "Stimulation of numerical skills in children with visual impairments using image recognition," *Procedia Computer Science*, vol. 198, pp. 179–184, 2022. doi:10.1016/j.procs.2021.12.226

[7] M. Shoaib, S. Khan, D. Fitzpatrick, and I. Pitt, "A mobile e-learning application for enhancement of basic mathematical skills in visually impaired children," *Universal Access in the Information Society*, 2023. doi:10.1007/s10209-023-00990-3

[8] B. Lučić, S. Ostrogonac, N. Vujnović Sedlar, and M. Sečujski, "Educational applications for blind and partially sighted pupils based on speech technologies for Serbian," *The Scientific World Journal*, vol. 2015, pp. 1–14, 2015. doi:10.1155/2015/839252

[9] L. R. Milne, C. L. Bennett, and R. E. Ladner, "VBGhost," *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility*, 2013. doi:10.1145/2513383.2513396

[10] L. Nahar, A. Jaafar, E. Ahamed, and A. B. Kaish, "Design of a Braille learning application for visually impaired students in Bangladesh," *Assistive Technology*, vol. 27, no. 3, pp. 172–182, 2015. doi:10.1080/10400435.2015.1011758

[11] T. Kausar *et al.*, "Deep learning strategy for braille character recognition," *IEEE Access*, vol. 9, pp. 169357–169371, 2021. doi:10.1109/access.2021.3138240

[12] C. Li and W. Yan, "Braille recognition using Deep Learning," *2021 4th International Conference on Control and Computer Vision*, 2021. doi:10.1145/3484274.3484280

[13] K. Smelyakov, A. Chupryna, D. Yeremenko, A. Sakhon, and V. Polezhai, "Braille character recognition based on Neural Networks," *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, 2018. doi:10.1109/dsmp.2018.8478615

[14] B.-M. Hsu, "Braille recognition for reducing asymmetric communication between the blind and Non-Blind," *Symmetry*, vol. 12, no. 7, p. 1069, 2020. doi:10.3390/sym12071069

[15] R. Vaidya, D. Trivedi, S. Satra, and Prof. M. Pimpale, "Handwritten character recognition using deep-learning," *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2018. doi:10.1109/icicct.2018.8473291

[16] M. D Souza , P. S, V. S M, N. Vardhan S, and V. G, "BRAILLE CHARACTER RECOGNITION USING DEEP LEARNING STRATEGY IMAGE PROCESSING AND COMPUTER VISION," *International Research Journal of Modernization in Engineering Technology and Science* , vol. 05, no. 05, May 2023.

[17] J. A. Shah, A. Raorane, A. Ramani, H. Rami, and N. Shekokar, "Eyeris: A virtual eye to aid the visually impaired," *2020 3rd International Conference on Communication System, Computing and IT Applications (CSCITA)*, 2020. doi:10.1109/cscita47329.2020.9137777

[18] Ramiati, S. Aulia, Lifwarda, and S. N. Nindya Satriani, "Recognition of image pattern to identification of Braille characters to be audio signals for Blind Communication Tools," *IOP Conference Series: Materials Science and Engineering*, vol. 846, no. 1, p. 012008, 2020. doi:10.1088/1757-899x/846/1/012008

[19] R. M. Pinheiro Pereira, C. E. F. Matos, G. Braz Junior, J. D. S. de Almeida, and A. C. de Paiva, "A deep approach for handwritten musical symbols recognition," *Proceedings of the 22nd Brazilian Symposium on Multimedia and the Web*, 2016. doi:10.1145/2976796.2988171

[20] D. Sarwinda, R. H. Paradisa, A. Bustamam, and P. Anggia, "Deep learning in image classification using residual network (ResNet) variants for detection of colorectal cancer," *Procedia Computer Science*, vol. 179, pp. 423–431, 2021. doi:10.1016/j.procs.2021.01.025

[21] X. Jin *et al.*, "Deep image aesthetics classification using inception modules and fine-tuning connected layer," *2016 8th International Conference on Wireless Communications & Signal Processing (WCSP)*, 2016. doi:10.1109/wcsp.2016.7752571

[22] *T. Kausar, S. Manzoor, A. Kausar, Y. Lu, M. Wasif and M. A. Ashraf, "Deep Learning Strategy for Braille Character Recognition," in IEEE Access, vol. 9, pp. 169357-169371, 2021.*