## **Machine Learning Project**

**Group Members:**

1. Chinni Vamshi Krushna- 2021A7PS2084H
2. Atharva Chikhale- 2021A7PS2752H
3. Rishabh Mittal- 2021A7PS2620H

**Observations**

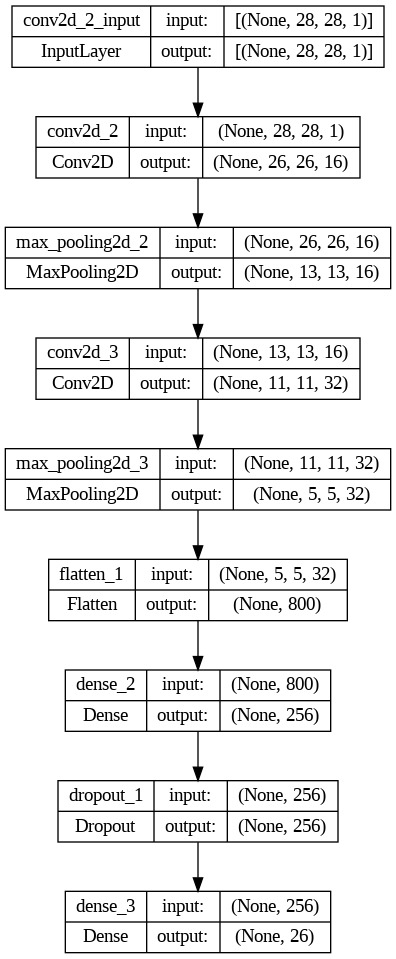
## **Analyze the variations in loss and testing accuracy over epochs to assess model performance.**

Based on the results of training and validation over 10 epochs, here are the observations and analysis that can be included in the report to discuss the model performance, specifically addressing overfitting and how the CNN handles it: Analysis of Model Performance

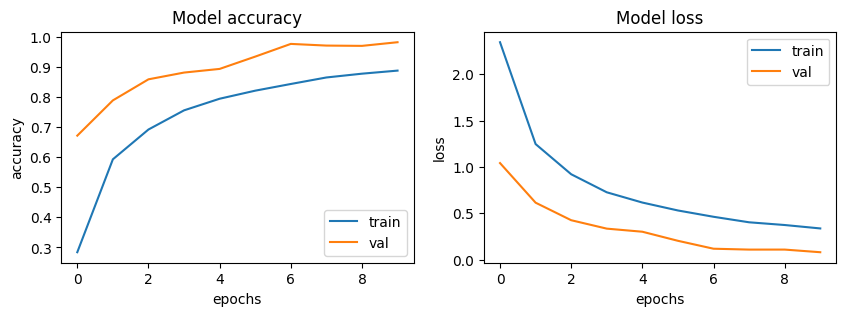
**Over Epochs Trend in Accuracy:** The training accuracy consistently improves from 90.78% in the first epoch to 94.65% by the tenth epoch. This steady increase indicates effective learning, where the model gradually becomes better at recognizing ASL hand gestures from the training data. The validation accuracy starts at an already high 98.63% and finishes at 99.61% in the final epoch, showing that the model performs exceptionally well on unseen data.

**Trend in Loss**: The training loss decreases from 0.2755 to 0.1610 across epochs, showcasing a general improvement in the model's predictions on the training data. The validation loss starts low at 0.0499 and exhibits some fluctuations but generally trends downwards, reaching its lowest at 0.0169 in the eighth epoch before a slight increase in subsequent epochs. The low and decreasing trend in validation loss is indicative of good generalization to new data.

**Modal Visualisations**



## 



**Analysis of Model Performance With Additional Dropout Layers**

**Model Training and Validation Performance:**

Initial Training and Validation: In the first epoch, the training loss was significantly higher at 2.5999 with a training accuracy of 21.64%, while the validation loss was 1.4868 with a validation accuracy of 53.11%. This indicates that the model started with less immediate overfitting compared to typical starts where training accuracy significantly exceeds validation accuracy.

**Progress Over Epochs:** Over 10 epochs, both training and validation loss consistently decreased, showing an improvement from 2.5999 to 0.4847 in training loss and from 1.4868 to 0.0869 in validation loss. Similarly, accuracy improved significantly from 21.64% to 83.47% in training and from 53.11% to 97.39% in validation.

Comparative Analysis to Previous Configuration:

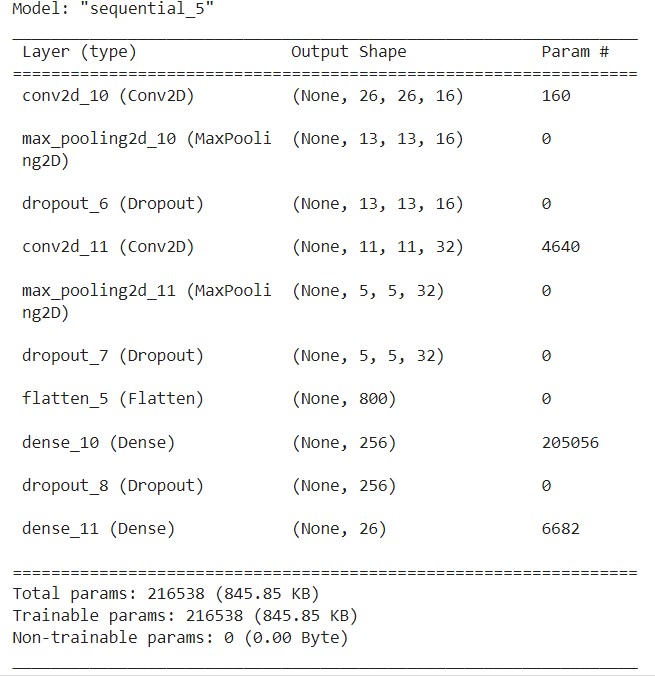
Higher Initial Losses: With additional dropouts, the model initially presented higher losses and lower accuracies, suggesting that the model was less able to fit the training data perfectly from the beginning, which is a characteristic desired to prevent overfitting.

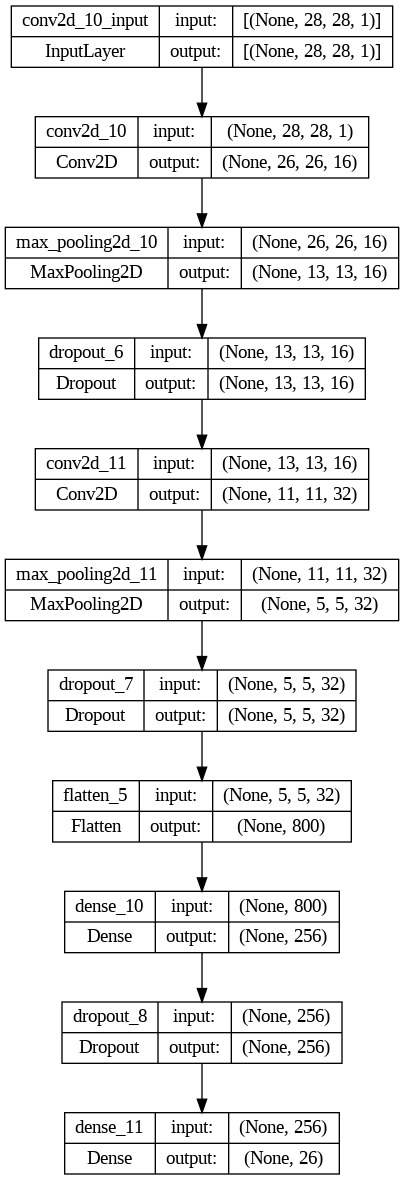
**Steady Convergence:** The convergence rate of the model in terms of loss reduction and accuracy improvement suggests a stable learning process without signs of overfitting, as indicated by the substantial gap reduction between training and validation metrics over epochs.

Evaluation of Overfitting:

**Reduced Overfitting**: The close parity between training and validation accuracy by the end of 10 epochs, particularly the validation accuracy surpassing training accuracy, strongly suggests that the additional dropout layers have effectively mitigated overfitting. Initially, the model was underfitting (validation metrics surpassing training metrics significantly), but as the epochs progressed, the model balanced out, achieving high performance on both training and validation sets.

**Dropout Effectiveness:** The dropout layers helped in regularizing the network by preventing it from relying too heavily on any particular set of weights, thus enhancing its generalization capabilities to new, unseen data.





## **Share structure property and Invariance property**

**Shared Structure Property**

Definition: The shared structure property in CNNs refers to the network's ability to learn filters that are useful across the entire input space. This means that the same filters (weights) are applied to different parts of the image, allowing the network to detect features like edges, shapes, or specific textures regardless of their position in the image.

Implementation in our Project:

**Convolutional Layers:** Our model uses convolutional layers which inherently utilize shared weights to scan through the entire image. This allows the model to detect the same features anywhere in the image, making it efficient and powerful for image recognition tasks, including recognizing ASL signs, where the same hand configuration might appear in different spatial positions.

**Feature Map Generation:** As the convolutional filters slide over the image, they generate feature maps that represent the presence of specific features at different locations in the image. This operation encodes the spatial hierarchy of features, which is crucial for recognizing complex patterns like hand gestures.

**Invariance Property**

Definition: Invariance in the context of CNNs refers to the ability of the model to correctly classify input images despite variations in position, scale, or other transformations. In essence, the model can recognize the same object (or gesture) no matter how it is presented.

Implementation in our Project:

**Pooling Layers:** Our CNN includes max pooling layers following the convolutional layers. Max pooling helps in making the representation more abstract and invariant to minor translations and distortions. This is especially important in gesture recognition, where a gesture might not be perfectly centered or aligned in every image.

**Data Augmentation:** Through the use of ImageDataGenerator, your model processes images that have been randomly zoomed, shifted, or rotated. This exposure to variations during training teaches the model to recognize ASL gestures regardless of how they are oriented or positioned in the input image, thereby enhancing its invariance to such transformations.