<u>Table</u>

Experiment Number	Model	Result	Decision (Explnation given after the table due to its length)
1	**Conv3D 4 Layers + GlobalAveragePooling3D + Dense (2 Layers)**	**1. Training Accuracy: 85.46% 2. Validation Accuracy: 95.00%**	**This model is a strong candidate due to its high validation accuracy and moderate parameter count, making it suitable for deployment in resource-constrained environments.**
2	**TimeDistributed Conv2D 3 Layers + GRU + Dense (2 Layers)**	**1. Training Accuracy: 98.75% 2. Validation Accuracy: 80.00%**	**This model is less favorable due to the high risk of overfitting and its large number of parameters, which may not be optimal for real-time applications.**
3	**TimeDistributed MobileNetV2 + GlobalAveragePooling2D + GRU + Dense (2 Layers)**	**1. Training Accuracy: 98.75% 2. Validation Accuracy: 95.00%**	**This model is highly recommended due to its excellent balance between training and validation accuracy and a moderate number of parameters, making it suitable for deployment in real-world applications.**

1. **Conv3D 4 Layers + GlobalAveragePooling3D + Dense (2 Layers):**

Explanation:

Model Construction:
The Conv3D model architecture was chosen to capture spatial and temporal features of video data. By using 3D convolutions, the model processes both the spatial dimensions (height and width) and the temporal dimension (frames) simultaneously, which is crucial for gesture recognition from video sequences.

Performance:
The training accuracy showed a steady improvement over epochs, reaching 85.46%. The validation accuracy significantly improved towards the later stages, reaching 95.00%, indicating the model's ability to generalize well after sufficient training. The graphs show a narrowing gap between training and validation accuracy.

Generator Experiments:
Adjusted the batch size and experimented with different image preprocessing steps such as resizing and normalization to ensure consistent input dimensions.

Advantages:
The model improved its generalization ability significantly by epoch 35, making it a robust choice for handling complex tasks with moderate parameters. The use of GlobalAveragePooling3D and dense layers helps in reducing overfitting while capturing essential features.

2. TimeDistributed Conv2D 3 Layers + GRU + Dense (2 Layers):

Explanation:

Model Construction:

This model combines TimeDistributed Conv2D layers with GRU (Gated Recurrent Unit) layers to leverage the strengths of both convolutional neural networks for spatial feature extraction and recurrent neural networks for temporal sequence modeling. The TimeDistributed wrapper applies the same Conv2D layer to each frame individually before feeding the sequence to the GRU layer.

Performance:

The model demonstrated high training accuracy early on, reaching 98.75%, but had lower validation accuracy at 80.00%, suggesting potential overfitting. The training accuracy curve showed a steep rise, whereas the validation accuracy curve plateaued, indicating the need for regularization.

Generator Experiments:

Handled varying input shapes and experimented with different frame counts and normalization techniques.

Disadvantages:

Despite the high training accuracy, the model's high number of parameters (19,937,221) likely contributed to overfitting. Regularization techniques such as dropout were used, but the large parameter count still posed a challenge for generalization. The loss graphs showed a similar trend, with the validation loss not decreasing as much as the training loss.

3. TimeDistributed MobileNetV2 + GlobalAveragePooling2D + GRU + Dense (2 Layers):

Explanation:

Model Construction:
This model utilizes a pre-trained MobileNetV2 model within a TimeDistributed wrapper, followed by GlobalAveragePooling2D and GRU layers. MobileNetV2 is a lightweight convolutional neural network that is highly efficient and effective for feature extraction. The TimeDistributed wrapper applies the MobileNetV2 to each frame individually. The GRU layer then processes the extracted features over the temporal dimension.

Performance:
This model achieved high training and validation accuracy, both at 98.75% and 95.00% respectively, indicating effective learning and generalization. The training and validation accuracy curves were closely aligned, demonstrating consistent performance across both datasets. The loss curves showed a consistent decrease without significant divergence.

Generator Experiments:
Experimented with different image resizing techniques, ensuring that smaller images were padded to match the input dimensions required by MobileNetV2.

Advantages:
Utilizing the pre-trained MobileNetV2 model for feature extraction significantly enhanced performance. This approach balanced parameter count and performance, making it effective for practical applications with high generalization capabilities. The use of pre-trained networks allows for leveraging previously learned features, which speeds up convergence and improves accuracy with fewer training data.