Problem Statement:
We need to develop a cool feature in the smart-TV that can recognise five different gestures performed by the user which will help users control the TV without using a remote.

The following table consists of the experiments done to build a model to predict the gestures from the given data set.

Exp#	Model	Hyperparameters	Result	Decision + Explanation

1	**Conv3D**
	- 3D Convolutional Model
	- 3 layers of Conv3D and 3 layers of MaxPooling3D
	- 2 Dense layers
	- 2 Dropout layers
	- 2 Batch Normalization layers
	- 25 epochs
		`total_frames = 30`
		`num_frames = 15`
		`gestures = 5`
		`image_height = 100`
		`image_width = 100`
			Training Accuracy: 0.41
			Validation Accuracy: 0.59
				- From the graph we can see that the model is performing better on the validation data than on the training data.
				- This indicates that the model is too simple and has lesser that required training parameters
				- The model is underfitting with the current parameters
				- The model has a very low training accuracy and better validation accuracy

2	**Conv3D**
	
	Reducing the size of the image from 100 to 50
		`total_frames = 30`
		`num_frames = 15`
		`gestures = 5`
		`image_height = 50`
		`image_width = 50`
			Training Accuracy: 0.48%
			Validation Accuracy: 0.67%
				- Scaling to smaller images not much of difference in the performance of the model.
				- The model is underfitting with the current parameters
				- The model has a very low training accuracy and better validation accuracy

3	**Conv3D**
	
	Reducing the size of the image from 50 to 25
		`total_frames = 30`
		`num_frames = 15`
		`gestures = 5`
		`image_height = 25`
		`image_width = 25`
			Training Accuracy: 0.42%
			Validation Accuracy: 0.49%
				- Scaling from 50 to 25 reduced the training and validation accuracy
				- The model is underfitting with the current parameters
				- The model has a very low training accuracy and validation accuracy

4	**Conv3D** **Decreasing the batch size from 8 to 4**	`total_frames = 30` `num_frames = 15`	**Training Accuracy: 0.71%** **Validation Accuracy: 0.79%**	**- Reducing the batch size has reduced overfitting** **- Model has resulted in better training and validation accuracy**

		gestures = 5 image_height = 50 image_width = 50 batch_size = 4		**- Accuracy has improved from the previous models** **- The model is underfitting with the current parameters**
5.	**Conv3D** **Changing the optmizer from `SGD` to `Adam`**	total_frames = 30 num_frames = 15 gestures = 5 image_height = 50 image_width = 50 batch_size = 4	**Training Accuracy: 0.39%** **Validation Accuracy: 0.48%**	**- The model is underfitting with the current parameters** **- Changing to Adam doesn't yield great result**
6	**Conv3D** **Number of frames reduced to 10**	total_frames = 30 num_frames = 10 gestures = 5 image_height = 50 image_width = 50 batch_size = 4	**Training Accuracy: 0.42%** **Validation Accuracy: 0.61%**	**- The training and validation accuracy has increased after reducing the number of frames from 15 to 10** **- The model is still underfitting with the current parameters**
7	**Conv3D** **Reducing the dropout rate from 0.5 to 0.25**	total_frames = 30 num_frames = 10 gestures = 5 image_height = 50 image_width = 50 batch_size = 4	**Training Accuracy: 0.78%** **Validation Accuracy: 0.86%**	**- Model has resulted in better training and validation accuracy** **- Model's performance has improved from the previous models** **- Validation accuracy is better than the training accuracy**

8	**Conv3D**
	
	Increasing the number of epochs from 25 to 35
		`total_frames = 30`
		`num_frames = 10`
		`gestures = 5`
		`image_height = 50`
		`image_width = 50`
		`batch_size = 4`
			Training Accuracy: 0.82%
			Validation Accuracy: 0.82%
				- Good training and validation accuracy
				- Model's performance has improved from the previous models
				- Both training and validation accuracy are close to each other

9.	**Conv2D + RNN (GRU)**
	
	- 3 layers of Conv2D and 3 layers of MaxPooling2D
	- 2 GRU layers
	- 2 Dense layers
	- 2 Dropout layers
	- 3 Batch Normalization layers
		`total_frames = 30`
		`num_frames = 10`
		`gestures = 5`
		`image_height = 50`
		`image_width = 50`
		`batch_size = 4`
			Training Accuracy: 1.0%
			Validation Accuracy: 0.79%
				- The model is underfitting with the current parameters
				- Model has nearly 100% training accuracy and 80% validation accuracy

10	**Conv2D + RNN (GRU)**
	
	Increase dropout rate from 0.25 to 0.5
		`total_frames = 30`
		`num_frames = 10`
		`gestures = 5`
		`image_height = 50`
		`image_width = 50`
		`batch_size = 4`
			Training Accuracy: 0.89%
			Validation Accuracy: 0.71%
				- The model is overfitting with the current parameters

11	**CNN 2D + RNN architecture with LSTM layers**	`total_frames = 30`
		`num_frames = 10`
		`gestures = 5`
		`image_height = 50`
		`image_width = 50`
			Training Accuracy: 0.91%
			Validation Accuracy: 0.77%
				- The model is slightly overfitting with the current parameters

		batch_size = 4		
12	**CNN 2D + RNN architecture with LSTM layers** **Replace flatten layer with Global average pooling**	total_frames = 30 num_frames = 10 gestures = 5 image_height = 50 image_width = 50 batch_size = 4	**Training Accuracy: 0.83%** **Validation Accuracy: 0.71%**	**- The model is better in terms of training and validation accuracy** **- The model is not overfitting or underfitting** **- The model has a good training and validation accuracy**
Best Model	**Transfer Learning with MobileNetV2** **- MobileNetV2 with 2 Dense layers** **- 2 Dropout layers** **- 3 Batch Normalization layers** **- 25 epochs**	total_frames = 30 num_frames = 10 gestures = 5 image_height = 50 image_width = 50 batch_size = 4	**Training Accuracy: 0.96%** **Validation Accuracy: 0.96%**	**- Best model of all the models** **- Total number of trainable parameters -** 3486085