S.N	EXPERIMENT	# TOTAL PARAMETER	# TRAINABLE PARAMETERS	BEST TRAINING ACCURACY	BEST VALIDATION ACCURACY	OBSERVATION
Basic Conv 3D						
1	**Size check** Image size of 160 and 30 frames	1,736,389	1,735,525	-	-	ResourceExhaustedError: OOM when allocating tensor with shape [40,16,30,160, 160]
2	**Size check** Image size of 160 and 16 frames	1,736,389	1,735,525			Size fits in memory. So number of epochs can be increased to check the performance
3	**Reduced Image size** Image size of 100 , 16 frames, batch size of 30 & 20 epochs	687,813	686,949	89	78	The model is performing decently. So, the model can be experimented with different hyperparameters
4	**Increasing batch size:** Image size of 100, 16 frames , batch size of 60 & 20 epochs	687,813	686,949	86	70	Model is slightly overfitting. So, augmentation and dropouts can be tried.
5	**Using Augmentation** Image size of 100, 16 frames of batch size 30 & 20 epochs	687,813	686,949	88	76	Augmentation has done a marginal improvement in accuracy but the training time has increased considerably.

6	**Increasing Dropout –** Image size of 100, 16 frames of batch size 30 & 20 epochs	687,813	686,949	84	77	Overfitting has been reduced upon using dropouts alone.
7	**Increasing batch size:** Image size of 100, batch size of 80 & 20 epochs	687,813	686,949	93	75	The model starts overfitting as batch size increases
8	**Increased dropout with augmentation** Image size of 100, 16 frames of batch size 30 & 20 epochs	687,813	686,949	86	82	Combining both dropout and augmentation has reduced overfitting considerably. Getting good results with lesser parameters. But, training time increases as augmentation is used
9	**Increased number of epochs –** Image size of 100, 16 frames of batch size 30 & 30 epochs	687,813	686,949	89	84	Getting good results with lesser parameters. Slight increase in performance and convergence noticed on increasing epochs.
10	**Increased image size – no augmentation-** Image size of 160, 16 frames of batch size 30 & 30 epochs	1,736,389	1,735,525	88	87	Getting good results. Parameters increased as image size increases but training time is less as augmentation is not used

11	**Increased number of frames** - Image size of 160, 20 frames of batch size 30 & 30 epochs	1,736,389	1,735,525	92	87	Increasing number of frames and found Good Results
12	**Increased image size & augmentation** Image size of 160, 16 frames of batch size 20 & 30 epochs	1,736,389	1,735,525	94	88	To increase accuracy, increase in image size & augmentation are combined which resulted in very good results
13	**Increased kernel size –** 2,2,2 kernels are replaced with 3,3,3 filters -- - Image size of 100, 16 frames of batch size 30 & 30 epochs	892,101	891,237	88	82	Experimented with increased kernel size & got Good results
14	**Increased kernel size** - Image size of 160, 16 frames of batch size 30 & 30 epochs	1,940,677	1,939,813	90	83	Though the model gives good results in a few epochs, its generalisability is not satisfactory
15	**Decreasing learning rate** - Image size of 100, 16 frames of batch size 30 & 30 epochs	892,101	891,237	88	73	Decreased learning rate could not make a general model in 30 epochs
16	**Increased kernel size with augmentation** - Image size of 100, 16 frames of batch size 30 & 30 epochs	892,101	891,237	88	83	Good results but training time was more. **With and without augmentation, kernel size of 3,3,3 performs almost the same**

17	**Reduced kernel size -** - Image size of 100, 16 frames of batch size 30 & 30 epochs	686,901	686,037	96	82	Number of parameters reduces but Model is slightly overfitting. So trying dropout
18	**Reduced kernel size and increased dropout** - Image size of 100, 16 frames of batch size 30 & 30 epochs	686,901	686,037	86	80	Dropout reduces overfitting & Decent performance
19	**Increased image size**- Image size of 120, 16 frames of batch size 30 & 30 epochs	899,893	899,029	83	80	Image size in increased to check for increase in accuracy. Decent performance but no improvement in accuracy
20	**Increased image size**- Image size of 160, 16 frames of batch size 30 & 30 epochs	1,735,477	1,734,613	87	81	Decent performance. Inference is as image size increases the 2,2,2 kernel model doesn't increase accuracy
21	**Increased neurons in dense layer** - Image size of 100, 16 frames of batch size 30 & 30 epochs	1,285,557	1,284,437	88	78	Number of parameters increases and Model slightly overfits
CNN-RNN MODEL						
22	**CNN-RNN Model without transfer learning -** Image size of 160, 20 frames of batch size 20 & 30 epochs	5,026,293	5,025,813	92	76	Model overfits

23	**Transfer learning – Mobilenet with GRU – non trainable parameters**	3,692,869	461,957	97	46	Poor performance
24	**Transfer learning – Mobilenet with GRU – trainable parameters**	3,692,869	3,668,933	98	92	Excellent results

- Note : Highlighted Models have good performance

CONV3D MODEL WITH LEAST PARAMETERS: (preferable model for deployment)

- Total Parameters : 687,813
- Training Accuracy: 89
- Validation Accuracy: 84
- Decision: This model is preferred where low footprint is required

CONV3D BEST PERFORMANCE MODEL:

- Total Parameters : 1,736,389
- Training Accuracy: 94
- Validation Accuracy: 88
- Decision: This model is preferred where low footprint is not mandatory

CNN-RNN TRANSFER LEARNING MODEL:

- Total Parameters : 3,692,869
- Training Accuracy: 98
- Validation Accuracy: 92
- Decision: This model is preferred where high accuracy is the priority without any constraints

As we have to deploy the model in a television according to the problem statement, the conv3D model which gives good results with least parameters can be chosen for deployment.