Hand Gesture Assignment Write-up

Problem Statement: From video Sequence of 30 images the model has to accurately classify 5 different class of gestures.

Models considered for building the solution.

- 1. 3D Convolution Neural Network (Conv3D).
- 2. 2D Convolution Neural Network + Recurrent Neural Network (ConvGRU).

Experiments Performed and their outcome:

Model Architecture, 3D Convolution Neural Network:

Model 1 (Conv3D):

Activation Function - Relu

Conv3D - 64 - Filter size (3*3*3) - Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) - Stride (2,2,2)

Conv3D - 128 - Filter size (3*3*3) – Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) – Stride (2,2,2)

Conv3D - 256 - Filter size (3*3*3) – Stride (1,1,1) Conv3D - 256 - Filter size (3*3*3) – Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) - Stride (2,2,2)

Fully-Connected -512 Output -SoftMax (5)

Note: *Changes made in the experiment has been highlighted by $\underline{\text{underlying}}$ them.

Experiment Number	Model	Hyperparameters	Observation	Result
1	Model 1 (Conv3D)	Batch Size:100 Image Sequence:15 (skip 1 image in sequence) Image Size: 100*100	Out of memory error and thus the batch size has to be reduced to fit into GPU memory.	Resource Exhausted Error: OOM, During model compilation
		Epochs: 40 ImageNormalization: Scaled all channels by 1/255 (0-1 range)	In the next model Batch size has to be reduced.	
		Optimizer: SGD (Nestrov) LearningRate: 0.001 Momentum: 0.7		

^{*}Batch Normalization Performed after Activation function.

2	Model 1 (Conv3D)	Batch Size: 50	Model has been	Accuracy Score
2	Wodel I (Convad)	Image Sequence:15	overfitting for this	Accuracy Score
		(skip 1 image in	configuration.	categorical_accuracy:
		sequence)	comiguration.	0.8847
		Image Size: 100*100	Validation and test	0.0047
		Epochs: 40	losses have been	val_categorical_accuracy:
		Epochs: 40	decrementing at	0.5600
			slower rate and	
		ImageNormalization:	thus increasing the	
		Scaled all channels	learning rate and	
		by 1/255 (0-1 range)	Epoch numbers	
		Optimizer: SGD		
		(Nestrov)		
		LearningRate: 0.001		
		Momentum: 0.7		
3	Model 1 (Conv3D)	Batch Size: 50	Model overfitting	Accuracy Score
		Image Sequence:15	problem still	
		(skip 1 image in	persists and thus	categorical_accuracy:
		sequence)	reducing the image	0.9598
		Image Size: 100*100	size and increasing	
		Epochs: 50	the batch size to	val_categorical_accuracy:
			check the model	0.6100
		lana a a Na wasa li a ati a a .	accuracy.	
		ImageNormalization: Scaled all channels		
		by 1/255 (0-1 range)		
		by 1/233 (0-1 range)		
		Optimizer: SGD		
		(Nestrov)		
		LearningRate: 0.01		
		Momentum: 0.7		
4	Model 1 (Conv3D)	Batch Size: 70	Model overfitting	Accuracy Score
		Image Sequence:15	problem still	
		(skip 1 image in	persists. And thus,	categorical_accuracy:
		sequence)	introducing	0.9575
		Image Size: 64*64	Droupout layer in	
		Epochs: 50	the next model.	val_categorical_accuracy: 0.7300
		ImageNormalization:		
		Scaled all channels		
		by 1/255 (0-1 range)		
		Ontimicary		
		Optimizer: SGD		
		(Nestrov) LearningRate: 0.01		
		Momentum: 0.7		
		MICHIGING U./		
	1	i.	1	1

5	Model 1 (Conv3D) +	Batch Size: 80	Model is	Accuracy Score
	Droupout Layer	Image Sequence:15	underfitting and	recuracy score
	Dioapout Layer	(skip 1 image in	thus Droupout layer	categorical_accuracy:
		sequence)	is not helping.	0.6048
		Image Size: 64*64	Reducing image	0.0048
		Epochs: 50	sequence to 10 and	val sategorical assuracy:
		Epociis. 50	removing the	val_categorical_accuracy: 0.5800
		lmagaNaymalization.	dropout layer.	0.3800
		ImageNormalization: Scaled all channels	uropout layer.	
		by 1/255 (0-1 range)		
		Optimizer: SGD		
		(Nestrov)		
		LearningRate: 0.01		
		_		
6	Model 1 (Conv3D)	Momentum: 0.9 Batch Size: 80	Model is	Accuracy Score
0	iviouei 1 (Collysu)			Accuracy Score
		Image Sequence:10	underfitting.	sategorical accuracy:
		(skip 2 image in	Dottor stratomy is:	categorical_accuracy: 0.7391
		sequence)	Better strategy is:	0.7391
		Image Size: 64*64	1. Changing	
		Epochs: 50	image	val_categorical_accuracy:
			normalization,	0.5900
			by subtracting	
		ImageNormalization:	each pixel	
		Scaled all channels	value per	
		by 1/255 (0-1 range)	channel by	
			mean value of	
		Optimizer: SGD	channel in test	
		(Nestrov)	image folder.	
		LearningRate: 0.01	2. Changing	
		Momentum: 0.9	Activation	
			function to	
			"Elu".	
			3. Batch	
			normalizing	
			before	
			activation	
			function.	
		1	L	<u> </u>

Model Architecture 3D Convolution Neural Network:

Model 2 (Conv3D):

Activation Function – Elu

Conv3D - 64 - Filter size (3*3*3) - Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) - Stride (2,2,2)

Conv3D - 128 - Filter size (3*3*3) – Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) – Stride (2,2,2)

Conv3D - 256 - Filter size (3*3*3) – Stride (1,1,1) Conv3D - 256 - Filter size (3*3*3) – Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) - Stride (2,2,2)

Fully-Connected -512 Output -SoftMax (5)

^{*}Batch Normalization Performed Before Activation function.

Experiment	Model	Hyperparameters	Observation	Result
Number				
7	Model 2 (Conv3D)	Batch Size: 70	Model is	Accuracy Score
		Image Sequence:15	performing	
		(skip 1 image in	better compared	categorical_accuracy:
		<u>sequence)</u>	to previous	0.8649
		Image Size: 80*80	models. But it can	
		Epochs: 30	be improved	val_categorical_accuracy:
				0.7200
		ImageNormalization:	This is the best	Total parameters:
		Subtracting pixel value	Model that was	15,994,373
		per channel by mean	obtained using	
		value per channel.	conv3D	
			architecture with	
		Optimizer: SGD	an accuracy score	
		(Nestrov)	of 72%.	
		LearningRate: 0.01		
		Momentum: 0.7		

Model Architecture 3D Convolution Neural Network:

Model 3 (Conv3D):

Activation Function - Elu

Conv3D - 64 - Filter size (3*3*3) – Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) – Stride (2,2,2)

Conv3D - 128 - Filter size (3*3*3) – Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) – Stride (2,2,2)

Conv3D - 256 - Filter size (3*3*3) - Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) - Stride (2,2,2)

Conv3D - 256 - Filter size (3*3*3) – Stride (1,1,1) MaxPooling3D - Filter size (2*2*2) - Stride (2,2,2)

Fully-Connected -512 Output -SoftMax (5)

^{*}Batch Normalization Performed Before Activation function.

Experiment Number	Model	Hyperparameters	Observation	Result
8	Model 3 (Conv3D)	Batch Size:70 Image Sequence:16 (Img num: 0,1 + skip 1 image in sequence) Image Size: 80*80 Epochs: 35 ImageNormalization: Subtracting pixel value per channel by mean value per channel. Optimizer: SGD (Nestrov) LearningRate: 0.01 Momentum: 0.7	Overall Model performance has reduced but the performance of the model remains intact.	Accuracy Score categorical_accuracy: 0.6644 val_categorical_accuracy: 0.7100

<u>Model Architecture 2D Convolution Neural Network + Recurrent Neural Network:</u>

Model 4 (ConvGRU):

Activation Function – Relu

Base CNN Model: VGG19 with non-trainable image-net weights.

<u>GRU - 64</u> GRU - 16

Fully-Connected -64 Output -SoftMax (5)

Experiment Number	Model	Hyperparameters	Observation	Result
9	Model 4 (ConvGRU)	Batch Size:40 Image Sequence:15 (skip 1 image in sequence) Image Size: 120*120 Epochs: 40 ImageNormalization: Scaled all channels by 1/255 (0-1 range) Optimizer: SGD (Nestrov) LearningRate: 0.01 Momentum: 0.7	Model is underfitting. but still it can be improved by changing the image normalization strategy.	Accuracy Score categorical_accuracy: 0.5446 val_categorical_accuracy: 0.5300

<u>Model Architecture 2D Convolution Neural Network + Recurrent Neural Network:</u>

Model 5 (ConvGRU):

Activation Fun – Relu

Base CNN Model: VGG19 with non-trainable image-net weights.

<u>GRU - 32</u> GRU - 16

Fully-Connected -64 Output -SoftMax (5)

Experiment Number	Model	Hyperparameters	Observation	Result
10	Model 5 (ConvGRU)	Batch Size:40 Image Sequence:16 (Img num: 0,1 + skip 1 image in sequence) Image Size: 120*120 Epochs: 35 ImageNormalization: Subtracting pixel value per channel by mean value per channel. Optimizer: SGD (Nestrov) LearningRate: 0.01 Momentum: 0.7	Model performance has been improved and this is best Model that was obtained using ConvGRU architecture with an accuracy score of 82%.	Accuracy Score categorical_accuracy: 0.9447 val_categorical_accuracy: 0.8200 Total Params: 20,332,581

Conclusion:

Model 2 (Conv3D, experiment 8) and Model 5 (ConvGRU, experiment 11) have produced very good accuracy score of around 72% and 82% respectively. But ConvGRU based models have lot more parameters when compared to conv3D based models. And, since the overall objective is to obtain a better performing model with lesser number of parameters conv3D model can be used for final model evaluation.

Use Model 2 (Conv3D, experiment 8) architecture with corresponding .h5 file for test evaluation.

Further accuracy of the model can be improved using data augmentation technique.