**Experiment Writeup for Gesture Recognition Using Conv3D and ConvLSTM**

This document presents a detailed account of the experiments conducted to develop an accurate model for gesture recognition using 3D Convolutional Neural Networks (Conv3D) and ConvLSTM models. Various adjustments to data preprocessing, model architecture, and hyperparameters were made to improve performance and generalize the model better.

The metric used to evaluate model performance is **accuracy**, and each experiment's logic is explained below.

**Experiment No. 1**

**Model Used**: Conv3D  
**Result**: Error during training due to generator issues  
**Decision & Explanation**: The first experiment failed due to errors in the data generator. The main issue was related to how the images were being cropped or resized. The data was not being processed correctly, resulting in incompatible input dimensions for the Conv3D model. The plan is to properly crop the images to the correct size and simplify the experiment by attempting to overfit on a small subset of data. This strategy would help ensure that the model can at least learn the patterns on a limited dataset before scaling it up.

**Experiment No. 2**

**Model Used**: Conv3D  
**Result**: Model not trainable due to a large number of parameters  
**Decision & Explanation**: The Conv3D model architecture was too large, with a high number of parameters, making it untrainable within a reasonable time frame and computational resources. To address this, the image size was reduced, and the number of layers in the model was decreased. Reducing the input size and complexity of the network was crucial to make the model computationally feasible.

**Experiment No. 3**

**Model Used**: Conv3D  
**Accuracy**: 0.21  
**Decision & Explanation**: After reducing the image size and number of layers, the model was trainable but performed poorly with an accuracy of 0.21. The decision was made to increase the amount of trainable data to provide the model with more diverse examples. Additionally, the filter size was reduced in the Conv3D layers to extract more localized features and avoid overfitting on large receptive fields.

**Experiment No. 4**

**Model Used**: Conv3D  
**Accuracy**: 0.32  
**Decision & Explanation**: By increasing the dataset and tuning the model, there was an improvement in accuracy, reaching 0.32. However, the image cropping method was still too aggressive, potentially losing important information from the edges of the frame. The decision was made to reduce the extent of cropping to preserve more of the original image content. The logic was that hand gestures often span a significant portion of the frame, and over-cropping was likely discarding useful information.

**Experiment No. 5**

**Model Used**: Conv3D  
**Accuracy**: 0.38  
**Decision & Explanation**: The accuracy improved to 0.38 after adjusting the cropping, but the model was still not achieving acceptable results. It became clear that the Conv3D architecture, while useful for extracting spatiotemporal features, was not capturing sufficient complexity in the motion patterns required for gesture recognition. Conv3D might struggle with longer temporal dependencies and sequential motions, suggesting the need to explore architectures better suited for handling temporal sequences.

**Experiment No. 6**

**Model Used**: Conv3D  
**Accuracy**: 0.45  
**Decision & Explanation**: With further fine-tuning of hyperparameters and increasing the training data, the Conv3D model accuracy reached 0.45. Despite the improvement, this was still below the desired threshold. At this stage, it was decided to explore **ConvLSTM** models, which combine the spatial feature extraction of convolutional layers with the temporal sequence handling of LSTM layers. The hypothesis was that ConvLSTM would outperform Conv3D by better capturing the temporal aspect of the gesture sequences.

**Experiment No. 7**

**Model Used**: ConvLSTM  
**Accuracy**: 0.55  
**Decision & Explanation**: Initial training with ConvLSTM yielded an accuracy of 0.55, demonstrating an improvement over Conv3D. The temporal features captured by LSTM layers helped the model better recognize gestures that involve motion and sequential hand positions. The ConvLSTM model also required tuning, but it proved to be more effective at handling the video data.

**Experiment No. 8**

**Model Used**: ConvLSTM  
**Accuracy**: 0.65  
**Decision & Explanation**: After tuning hyperparameters such as learning rate, batch size, and LSTM units, the ConvLSTM model reached an accuracy of 0.65. This significant improvement over the Conv3D model validated the hypothesis that combining convolutional layers for spatial features with LSTM layers for temporal dependencies yields better performance in gesture recognition tasks.

**Experiment No. 9**

**Model Used**: ConvLSTM  
**Accuracy**: 0.75  
**Decision & Explanation**: By experimenting with additional data augmentation techniques (rotation, flipping, and scaling) and regularization strategies (dropout and batch normalization), the accuracy of the ConvLSTM model improved further to 0.75. The decision to use data augmentation was based on the understanding that the model needed more variety in the training data to generalize well to unseen examples.

**Final Model**

**Model Used**: ConvLSTM  
**Training Accuracy**: 0.95  
**Validation Accuracy**: 0.88  
**Decision & Explanation**: After several iterations, the final model achieved a training accuracy of 0.95 and a validation accuracy of 0.88. The use of ConvLSTM allowed the model to effectively capture both spatial and temporal features from the video sequences, making it superior to the Conv3D model for this specific gesture recognition task. This result confirms that ConvLSTM is better suited for tasks involving sequential data like hand gesture recognition, where capturing motion patterns over time is crucial for accurate predictions.

**Conclusions**

* **Conv3D** alone was insufficient in capturing the necessary temporal dependencies in the gesture sequences, as reflected in the modest accuracy improvements.
* **ConvLSTM** provided the best balance of spatial and temporal feature extraction, resulting in a final accuracy of 0.88 on the validation set.
* Data augmentation and tuning of hyperparameters played a significant role in improving model performance.

Further improvements could involve:

* Experimenting with deeper ConvLSTM architectures.
* Incorporating additional temporal data preprocessing techniques.
* Leveraging transfer learning by using pretrained models on similar tasks.

The final model is now ready for deployment, with the ability to predict gestures based on video input sequences.