Gesture Recognition for Smart TV Control

Problem Statement:

The project aims to develop a deep learning model for recognizing gestures captured through a smart TV's webcam, each corresponding to specific TV control functions. The objective is to achieve high accuracy while maintaining a low memory footprint (<50MB).

1. **Thumbs Up**: Increase the volume
2. **Thumbs Down**: Decrease the volume
3. **Left Swipe**: Jump backward 10 seconds
4. **Right Swipe**: Jump forward 10 seconds
5. **Stop**: Pause the movie

Dataset:

The dataset comprises 'train' and 'val' folders with 663 and 100 video frames, respectively. Two CSV files provide frame details and corresponding class labels. Videos consist of 30 frames, with sizes of 120x160 and 320x320.

Solution Approach:

Two model architectures were explored:

1. Conv3D CNNs
2. 2D CNN + RNN.

The 2D CNN + RNN category included Custom CNN + RNN and Pre-Trained CNN + RNN (utilizing MobileNet and ResNet152V2).

Experimental Methodology:

All models underwent initial testing with 30% of the input data (both training and validation sets) to evaluate their learning capabilities. Only models demonstrating learning potential were selected for further experiments. Once confirmed, the chosen models were retrained from scratch using 100% of the data.

All Experiments are conducted using :

- Image Size: 160 x 160
- Number of Channels: 3
- Number of Epochs: 55
- Batch Size: 10

Experimental Observations with 100% Data:

Custom CNN with GRU as RNN:

- *Significance:* Initial attempt using a custom CNN with a GRU as RNN.
- Training Accuracy: 55%
- Validation Accuracy: 36%
- Learning issues; ineffective model.

Custom CNN with LSTM as RNN:

- *Significance:* Custom CNN with LSTM explored for potential improvements.
- Training Accuracy: 85%
- Validation Accuracy: 56%
- Overfitting observed; indicating suboptimal performance.

Pre-trained MobileNet with GRU as RNN:

- *Significance:* Utilizing a pre-trained MobileNet for improved feature extraction.
- Training Accuracy: 95%
- Validation Accuracy: 72%
- Overfitting persists, but with an improvement in validation accuracy.

Pre-trained ResNet152V2 with GRU as RNN:

- *Significance:* Experimenting with a more complex pre-trained ResNet152V2.
- Training Accuracy: 98%
- Validation Accuracy: 75%
- Overfitting persists, but enhanced accuracy compared to MobileNet.

3D Convolution Model with Kernel Regularizer ('l2'):

- *Significance:* Introducing a 3D Convolution Model with kernel_regularizer='l2'.
- Training Time: ~574ms/step
- Training Accuracy: 94.53%
- Validation Accuracy: 82%
- Improved accuracy and reduced overfitting compared to previous models.

Conclusion:

The 3D Convolution Model with kernel_regularizer='l2' emerged as the most promising solution, offering faster training, improved accuracy, and reduced overfitting. This model effectively classifies gestures from webcam-captured video frames for smart TV control, aligning with the project's objectives. Despite its performance, the final model is impressively compact, with a size of only 32.3 MB, making it highly suitable for deployment in real-world applications with limited memory resources.