**Objective:**

To design an efficient and robust model for real-time gesture recognition using video sequences.

**Trial 1: Baseline 3D CNN Model (Shallow Architecture)**

**Model 1 Architecture:**

* **Layers**: 2 × Conv3D → 1 × Dense
* **Preprocessing Approach 1**: *On-the-fly* frame extraction and transformation during training.

**Observations:**

* Despite GPU support, **training was extremely slow**, as the GPU remained idle most of the time due to preprocessing bottlenecks on the CPU.
* **Outcome**: Abandoned this approach due to poor hardware utilization and inefficiency.

**Preprocessing Approach 2:**

* Frames extracted **prior to training**, converted into 4D tensors for each video.
* **Configuration**:
  + 18 frames per video
  + Resized to **90×90**, no cropping
  + Tensor shape: **(18, 90, 90, 3)** for RGB input

**Training Details:**

* **Optimizer**: Adam
* **Initial Learning Rate**: 0.001
* **LR Scheduler**: ReduceLROnPlateau (factor=0.5, patience=3)
* **Epochs**: 15

**Results:**

* **Training time** improved by ~8x
* **Validation Accuracy**: 74%

**Fine-tuning:**

* Reduced **patience** in LR scheduler from 3 → 2
* Trained up to 50 epochs
* **Peak Validation Accuracy**: **77%**

**Trial 2: Deeper 3D CNN Model (Model 2)**

**Architecture:**

* **Conv3D Blocks**: 4 sequential blocks with filter sizes → 16 → 32 → 64 → 128
* Each block: **Conv3D → ReLU → BatchNorm → MaxPooling3D**
* Followed by: **Global Average Pooling → Dense(256) → Dropout(0.4) → Dense(Softmax)**

**Model Summary:**

* **Total Parameters**: 301,381
* **Trainable**: 300,901
* **Non-trainable**: 480

**Results (with previous preprocessing: 18 frames, 90×90, no crop):**

* **Epochs**: 50
* **Val Accuracy**: 98%
* Further training to **100 epochs** led to:
  + **Val Accuracy**: 99%
  + **Val Loss**: 0.04

**Limitation:**

* **Real-time performance** was unsatisfactory
* Model failed to generalize well to **live camera input**, yielding only ~50% real-time accuracy

**Trial 3: Resolution & Frame Reduction + Smart Cropping**

**Motivation:**

To reduce overfitting and improve real-time inference by shortening temporal length and focusing on relevant spatial features.

**Configuration 1:**

* **Resolution**: 64×64
* **Frames**: 11 per video
* **Cropping**: 20 pixels from all sides
* **Val Accuracy**: 96%
* **Val Loss**: 0.056

**Configuration 2:**

* **Resolution**: increased to **90×90**
* Same frame count and crop
* **Test Accuracy**: 99.5%
* **Val Accuracy**: 99%
* **Val Loss**: 0.03

**Configuration 3:**

* Increased **crop size** to **30 pixels**
* Improved spatial localization of hand gestures
* **Val Accuracy**: **99%**
* **Real-time Prediction**: Significantly improved with higher consistency

**Trial 4: Temporal Modeling with 2D CNN + GRU**

**Motivation:**

To model temporal dynamics explicitly using a **2D CNN encoder** followed by **GRU** layers.

**Results:**

* **Max Validation Accuracy**: 96%
* Slightly underperformed compared to optimized **3D CNN (Model 2)**
* No significant gain in real-time accuracy

**Summary Table:**

| **Trial** | **Model Type** | **Preprocessing** | **Val Accuracy** | **Test Accuracy** | **Real-time Performance** | **Notes** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 3D CNN (Shallow) | On-the-fly → Preloaded | 77% | N/A | Poor | Preprocessing bottleneck initially |
| 2 | 3D CNN (Deep) | 18×90×90 no crop | 99% | 98–99% | ~50% | Overfitting suspected, poor live accuracy |
| 3 | 3D CNN (Optimized) | 11 frames, 90×90, crop | 99% | 99.5% | Good | Cropping greatly improved generalization |
| 4 | 2D CNN + GRU | Sequential frame input | 96% | N/A | Fair | Did not outperform best 3D CNN model |