Human Activity Recognition using ConvLSTM and LRCN on UCF50 Dataset

Akash Rana

April 2025

A project report submitted in partial fulfillment
of the requirements for the degree of

Computer Perception
Coursework

Contents

Abstract

Human activity recognition (HAR) is an essential task in computer vision with applications spanning surveillance, healthcare monitoring, and human-computer interaction. In this project, we implement and compare two deep learning approaches — Convolutional Long Short-Term Memory (ConvLSTM) networks and Long-term Recurrent Convolutional Networks (LRCN) — to classify human activities from video sequences in the UCF50 dataset. Both architectures combine convolutional layers for spatial feature extraction with recurrent mechanisms for temporal dynamics, enabling the models to effectively capture spatiotemporal dependencies in video data.

The experimental results demonstrate promising classification performance across selected activity classes, showcasing the effectiveness of both ConvLSTM and LRCN architectures for video-based action recognition tasks. Furthermore, the outputs generated by these models are directly utilized as input observations for our parallel research project, *Predictive Movement Modeling Using HMMs and DBNs*, aimed at forecasting future human movements in smart environments. This integrated pipeline serves practical applications in home automation, enabling proactive system responses based on predicted user activities. By combining deep learning-based activity recognition with probabilistic predictive models, we advance towards a holistic, intelligent home automation framework.

1 Introduction

Human activity recognition (HAR) from video sequences remains a challenging problem due to the complex nature of dynamic scene changes, occlusions, varying camera perspectives, and intricate temporal dependencies. While traditional convolutional neural networks (CNNs) have demonstrated strong performance in spatial feature extraction from images, they inherently lack the ability to capture temporal evolution across video frames. To address this limitation, recurrent neural networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, have been employed for sequence modeling due to their ability to learn long-term dependencies.

In this project, we implement and compare two advanced deep learning architectures for HAR: Convolutional Long Short-Term Memory (ConvLSTM) networks and Long-term Recurrent Convolutional Networks (LRCN). The ConvLSTM model extends the conventional LSTM architecture by integrating convolutional operations directly into the recurrent units, thereby preserving spatial hierarchies while modeling temporal dependencies. Conversely, the LRCN model employs a two-stage approach, where convolutional layers first extract spatial features from individual frames, which are then sequentially processed by an LSTM network to learn temporal patterns.

Our primary goal is to evaluate and compare the performance of these two architectures on a subset of the UCF50 dataset. Beyond achieving accurate video classification, this research serves as a foundational step towards our broader project, *Predictive Movement Modeling Using HMMs and DBNs*, where the outputs of these deep learning models are utilized as observed activities to predict future human movements in smart home environments. The integration of HAR with probabilistic predictive modeling holds significant promise for developing intelligent home automation systems capable of proactive decision-making and enhancing user-centric experiences.

2 Literature Context

Earlier approaches to video classification primarily relied on hand-crafted feature extraction techniques such as Histograms of Oriented Gradients (HOG3D) and Dense Trajectories. While effective to some extent, these methods were limited by their inability to capture complex spatiotemporal relationships inherent in video data.

The advent of deep learning significantly advanced the field. Donahue et al. (2015) proposed the CNN-LSTM hybrid model, where convolutional layers first extract spatial features from individual frames, followed by LSTM layers to capture temporal dependencies across frame sequences. This architecture marked an important step towards end-to-end learning for video analysis. Similarly, Tran et al. (2015) introduced 3D Convolutional Neural Networks (3D CNNs), which apply spatiotemporal convolutions directly over video volumes, thereby jointly learning spatial and temporal features.

Building upon these foundations, Shi et al. (2015) introduced the Convolutional LSTM (ConvLSTM) architecture, which integrates convolutional operations within the gating mechanisms of LSTM cells. This design allows the model to maintain spatial hierarchies while effectively learning temporal dynamics, leading to success in applications such as precipitation nowcasting and human activity recognition.

Another prominent architecture is the Long-term Recurrent Convolutional Network (LRCN), which decouples spatial and temporal processing by leveraging convolutional neural networks for frame-level feature extraction, followed by LSTM units for sequence modeling. LRCN has been effectively applied to tasks such as activity recognition, image captioning, and video description generation, showcasing its versatility and robust performance.

Given the complementary strengths of ConvLSTM and LRCN architectures in capturing spatiotemporal dependencies, this project adopts both approaches to evaluate their effectiveness on the UCF50 dataset. This comparative study not only advances our understanding of deep learning-based HAR models but also supports our broader goal of integrating video-based activity recognition into predictive frameworks for smart home automation.

3 Dataset Description

Dataset: *UCF50 Action Recognition Dataset*
The UCF50 dataset is a widely used benchmark for action recognition tasks. It contains over 6,600 video clips, spanning 50 diverse human activity categories such as sports, exercise routines, and daily activities. Videos in UCF50 are collected from YouTube, offering a variety of camera motions, viewpoints, object appearances, and background complexities,

thereby providing a robust platform for evaluating human activity recognition models.

Selected Classes for Experimentation:

- **WalkingWithDog**
- **TaiChi**
- **Swing**
- **HorseRace**

For the scope of this project, we selected a subset of four classes that exhibit a diverse range of motion dynamics and environmental contexts. This selection enables focused experimentation while ensuring variability in action complexity and visual characteristics.

Preprocessing Steps:

- **Frame Extraction:** Frames were extracted from each video using OpenCV to decompose videos into individual images for frame-wise processing.
- **Resizing:** Each frame was resized to 64×64 pixels to ensure uniform input dimensions and to reduce computational complexity.
- **Sequence Construction:** Sequences of **20 consecutive frames** were constructed for each video, effectively capturing short-term temporal dynamics.
- **Normalization:** Pixel intensity values were normalized to the range $[0, 1]$ to improve training convergence and numerical stability.
- **Label Encoding:** Class labels were converted into one-hot encoded vectors to facilitate multi-class classification in the deep learning models.

This preprocessing pipeline ensures that the input data is well-structured for both ConvLSTM and LRCN architectures, enabling efficient learning of spatial and temporal patterns essential for accurate activity recognition.

4 Methodology

This project follows a systematic approach involving dataset preprocessing, model design, training, and evaluation. Two deep learning models, namely ConvLSTM and LRCN, are implemented to recognize human activities from video sequences. The objective is to evaluate and compare the effectiveness of these architectures in capturing spatiotemporal dependencies for activity classification on the UCF50 dataset.

4.1 Overview

The overall methodology consists of the following stages:

1. **Data Preparation:** Extraction and preprocessing of video frames from the UCF50 dataset.
2. **Model Development:** Construction and training of ConvLSTM and LRCN models for activity classification.
3. **Evaluation:** Performance assessment of both models on the test data using standard metrics.

4.2 Approach 1: ConvLSTM Model

The ConvLSTM architecture integrates convolutional operations within the LSTM units to simultaneously capture spatial and temporal features from video data. Unlike conventional LSTMs, which process vectorized inputs, ConvLSTM operates directly on spatial feature maps, preserving the spatial hierarchies across time steps.

Key steps in ConvLSTM implementation:

- Input sequences of 20 frames, each resized to $64 \times 64 \times 3$.
- Initial ConvLSTM layers extract spatiotemporal features using 3×3 convolutional kernels.
- MaxPooling3D layers downsample spatial dimensions to reduce computational complexity.
- Dropout layers are employed to prevent overfitting.
- Flattened output is passed through dense layers with softmax activation for final classification.

ConvLSTM is particularly effective in modeling short-term dependencies across frame sequences, making it suitable for dynamic activities with strong temporal continuity.

4.3 Approach 2: LRCN Model

The Long-term Recurrent Convolutional Network (LRCN) architecture separates spatial and temporal modeling into two distinct stages. Initially, convolutional layers extract spatial features from individual video frames. These features are then sequentially fed into an LSTM network, which captures temporal dependencies across the frame sequence.

Key steps in LRCN implementation:

- Each frame undergoes convolutional processing to extract spatial descriptors.
- Frame-wise features are aggregated into a temporal sequence.
- LSTM layers process the sequence to learn temporal dynamics.
- Final dense layers with softmax activation classify the activity based on learned spatiotemporal patterns.

LRCN offers modularity in design, allowing the use of pre-trained CNNs for feature extraction and flexible temporal modeling through recurrent layers. This architecture is particularly advantageous for activities where individual frames contain discriminative spatial information.

5 Model Architecture

This section details the architectural design of the two deep learning models employed in this project: the Convolutional Long Short-Term Memory (ConvLSTM) network and the Long-term Recurrent Convolutional Network (LRCN). Both models are designed to capture spatial and temporal dependencies from sequences of video frames, but they differ in the integration and processing of these features.

5.1 ConvLSTM Architecture

The ConvLSTM model integrates convolutional operations within recurrent units, allowing it to capture both spatial and temporal patterns directly from sequences of video frames.

Input Specification:

- Input shape: $(20, 64, 64, 3)$, representing sequences of 20 frames, each of size 64×64 with 3 color channels.

Architectural Flow:

1. **ConvLSTM2D Layer:**
   - Filters: 4
   - Kernel size: 3×3
   - Activation: `tanh`
   - Return sequences: True
2. **MaxPooling3D Layer:**
   - Pool size: $1 \times 2 \times 2$ (time, height, width)

- Purpose: Reduce spatial dimensionality while preserving temporal information.

3. **Dropout Layer:**
   - Rate: 0.2
   - Purpose: Regularization to prevent overfitting.
4. **ConvLSTM2D Layer:**
   - Filters: 8
   - Kernel size: 3×3
   - Activation: `tanh`
   - Return sequences: True
5. **MaxPooling3D + Dropout:**
   - Further downsampling and regularization.
6. **ConvLSTM2D Layer:**
   - Filters: 14
   - Kernel size: 3×3
   - Activation: `tanh`
   - Return sequences: False
7. **Flatten Layer:**
   - Converts 3D feature maps into a 1D vector.
8. **Dense Output Layer:**
   - Units: Number of classes (4)
   - Activation: `softmax`
   - Purpose: Multi-class classification.

The ConvLSTM architecture excels at learning spatiotemporal correlations in a unified framework, making it particularly effective for activities characterized by continuous motion.

5.2 LRCN Architecture

The Long-term Recurrent Convolutional Network (LRCN) employs a modular architecture, where spatial features are first extracted from each frame using convolutional layers, and then temporal dynamics are learned by feeding these features into LSTM layers.

Input Specification:

Layer (type)	Output Shape	Param #
conv_lstm2d (ConvLSTM2D)	(None, 20, 62, 62, 4)	1,024
max_pooling3d (MaxPooling3D)	(None, 20, 31, 31, 4)	0
time_distributed (TimeDistributed)	(None, 20, 31, 31, 4)	0
conv_lstm2d_1 (ConvLSTM2D)	(None, 20, 29, 29, 8)	3,488
max_pooling3d_1 (MaxPooling3D)	(None, 20, 15, 15, 8)	0
time_distributed_1 (TimeDistributed)	(None, 20, 15, 15, 8)	0
conv_lstm2d_2 (ConvLSTM2D)	(None, 20, 13, 13, 14)	11,144
max_pooling3d_2 (MaxPooling3D)	(None, 20, 7, 7, 14)	0
time_distributed_2 (TimeDistributed)	(None, 20, 7, 7, 14)	0
conv_lstm2d_3 (ConvLSTM2D)	(None, 20, 5, 5, 16)	17,344
max_pooling3d_3 (MaxPooling3D)	(None, 20, 3, 3, 16)	0
flatten (Flatten)	(None, 2880)	0
dense (Dense)	(None, 4)	11,524

Figure 1: Architecture of ConvLSTM.

- Input shape: $(20, 64, 64, 3)$, representing sequences of 20 frames.

Architectural Flow:

1. **TimeDistributed Conv2D Layer:**
   - Filters: 32
   - Kernel size: 3×3
   - Activation: `relu`
   - Purpose: Frame-wise spatial feature extraction.
2. **TimeDistributed MaxPooling2D Layer:**
   - Pool size: 2×2
   - Purpose: Reduce spatial dimensions of each frame.
3. **TimeDistributed Flatten Layer:**
   - Converts frame-level feature maps into vectors.
4. **LSTM Layer:**
   - Units: 64
   - Purpose: Learn temporal dependencies from frame sequences.

5. **Dense Output Layer:**
   - Units: Number of classes (4)
   - Activation: `softmax`
   - Purpose: Final classification.

The LRCN model offers flexibility in design, making it well-suited for scenarios where individual frames contain salient spatial features, and temporal evolution is captured by the recurrent layers.

Layer (type)	Output Shape	Param #
time_distributed_3 (TimeDistributed)	(None, 20, 64, 64, 16)	448
time_distributed_4 (TimeDistributed)	(None, 20, 16, 16, 16)	0
time_distributed_5 (TimeDistributed)	(None, 20, 16, 16, 16)	0
time_distributed_6 (TimeDistributed)	(None, 20, 16, 16, 32)	4,640
time_distributed_7 (TimeDistributed)	(None, 20, 4, 4, 32)	0
time_distributed_8 (TimeDistributed)	(None, 20, 4, 4, 32)	0
time_distributed_9 (TimeDistributed)	(None, 20, 4, 4, 64)	18,496
time_distributed_10 (TimeDistributed)	(None, 20, 2, 2, 64)	0
time_distributed_11 (TimeDistributed)	(None, 20, 2, 2, 64)	0
time_distributed_12 (TimeDistributed)	(None, 20, 2, 2, 64)	36,928
time_distributed_13 (TimeDistributed)	(None, 20, 1, 1, 64)	0
time_distributed_14 (TimeDistributed)	(None, 20, 64)	0
lstm (LSTM)	(None, 32)	12,416
dense_1 (Dense)	(None, 4)	132

Figure 2: Architecure of LRCN.

6 Training Procedure

Both the ConvLSTM and LRCN models were trained using a consistent and rigorous procedure to ensure fair comparison of their performance on the activity recognition task. This section outlines the training configurations, optimization strategies, and regularization techniques employed during the experiments.

6.1 Training Configuration

- **Loss Function:** Categorical crossentropy was used as the loss function, suitable for multi-class classification tasks where class probabilities are predicted.
- **Optimizer:** The Adam optimizer was selected for its adaptive learning rate capabilities and computational efficiency.
- **Evaluation Metric:** Accuracy was monitored as the primary performance metric during training and evaluation.
- **Batch Size:** A batch size of 4 was used, balancing training speed with memory constraints.
- **Epochs:** The models were trained for a maximum of 30 epochs, with early stopping employed to prevent overfitting.
- **Early Stopping:** Training was monitored with an early stopping callback based on validation loss, with a patience of 5 epochs.
- **Validation Split:** During training, 20% of the training data was used for validation to monitor model generalization.
- **Random Seed:** A fixed random seed of 27 was set to ensure reproducibility of the results.

6.2 ConvLSTM Model Training

The ConvLSTM model was trained using sequences of 20 frames per sample. Each frame sequence was fed into the model as a 5-dimensional tensor of shape $(batch\_size, time\_steps, height, width$

Key aspects of ConvLSTM training:

- Training leveraged GPU acceleration for faster convergence.
- Dropout layers provided regularization to reduce overfitting.
- Model checkpoints were not used, as early stopping effectively prevented overtraining.

6.3 LRCN Model Training

The LRCN model processed sequences where each frame was first passed through a shared convolutional backbone within a `TimeDistributed` wrapper, and the resulting frame-level features were then fed sequentially into LSTM layers.

Key aspects of LRCN training:

- Frame-wise convolutional feature extraction was parallelized across time steps.
- LSTM layers processed the flattened feature vectors from each frame.
- Dropout was applied at appropriate stages to mitigate overfitting.
- The same early stopping strategy was applied as in the ConvLSTM training.

6.4 Training Infrastructure

All experiments were conducted using TensorFlow and Keras frameworks, with GPU acceleration enabled where available. Training progress was monitored using live accuracy and loss curves, and results were logged for post-training evaluation and comparison.

This consistent training setup ensured an unbiased comparison between the ConvLSTM and LRCN models, allowing for reliable conclusions regarding their effectiveness in human activity recognition tasks.

7 Evaluation and Results

The performance of both the ConvLSTM and LRCN models was evaluated using the test set derived from the UCF50 dataset. Evaluation focused on quantitative metrics such as accuracy and loss, as well as qualitative analysis of sample predictions to assess the models' ability to generalize to unseen data.

7.1 Evaluation Metrics

- **Accuracy:** The primary metric for evaluating model performance, representing the proportion of correctly classified instances in the test set.
- **Loss:** Categorical crossentropy loss was used to monitor model convergence and assess learning stability across epochs.
- **Qualitative Analysis:** Sample predictions were visually inspected to understand the models' strengths and failure cases.

Both models were trained and evaluated under identical conditions to ensure a fair and meaningful comparison.

7.2 ConvLSTM Model Results

The ConvLSTM model demonstrated robust performance in recognizing activities from sequences of video frames. The training and validation accuracy curves showed consistent improvement across epochs, with early stopping preventing overfitting.

Key observations:

- **Test Accuracy:** 79.25%
- **Test Loss:** 0.5773
- The model performed particularly well on activities with continuous motion patterns, such as *WalkingWithDog* and *HorseRace*.
- Some confusion was observed between classes with visually similar motion profiles, such as *Swing* and *TaiChi*.

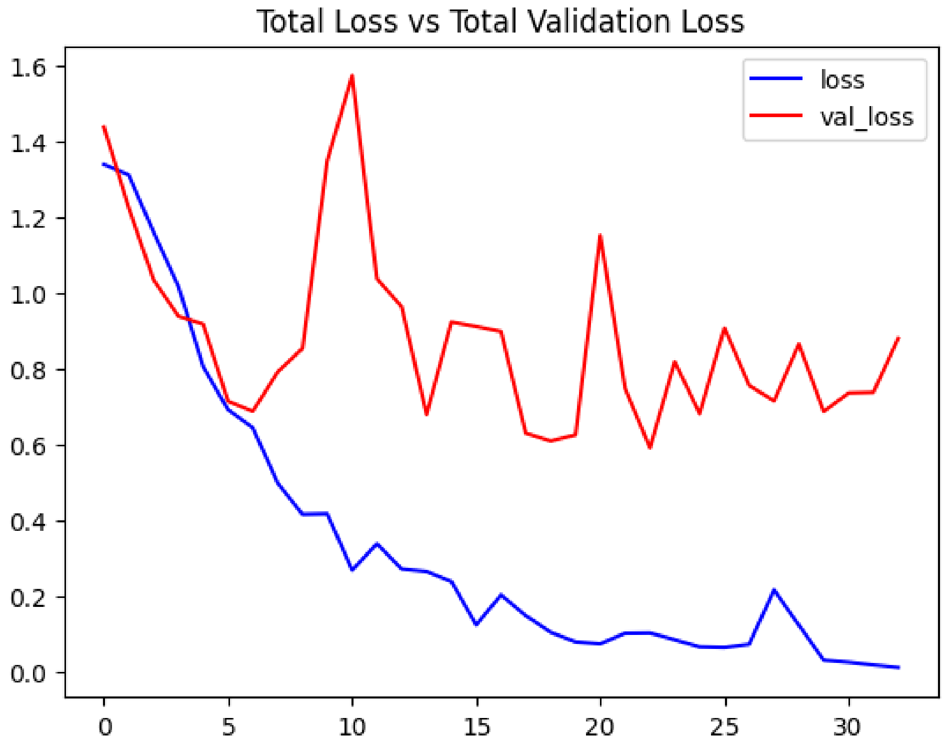

Figure 3: ConvLSTM training and validation loss curves.

7.3 LRCN Model Results

The LRCN model achieved superior performance compared to ConvLSTM, effectively recognizing activities by leveraging frame-level convolutional features and temporal sequence modeling.

Key observations:

- **Test Accuracy:** 86.93%
- **Test Loss:** 0.2854

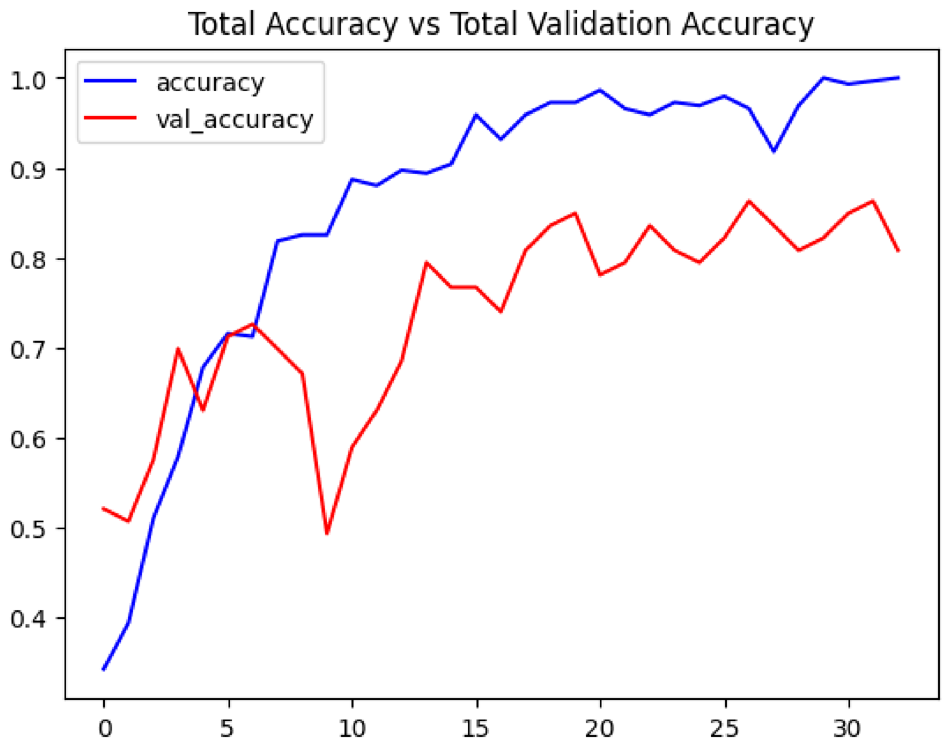

Figure 4: ConvLSTM training and validation accuracy curves.

- The model showed strong performance across all classes, particularly excelling in activities where individual frames carried significant spatial information.
- Misclassifications were minimal, emphasizing the effectiveness of LRCN's decoupled spatial-temporal learning approach.

7.4 Comparative Analysis

A direct comparison between the ConvLSTM and LRCN models reveals insightful contrasts:

- While ConvLSTM excelled at capturing temporal continuity, especially in fluid motion activities, LRCN demonstrated superior overall accuracy and lower loss, indicating better generalization.
- LRCN benefited from its two-stage processing pipeline, effectively extracting spatial features before temporal aggregation.
- The empirical results suggest that LRCN outperforms ConvLSTM for the chosen subset of the UCF50 dataset.

Summary Table of Results:

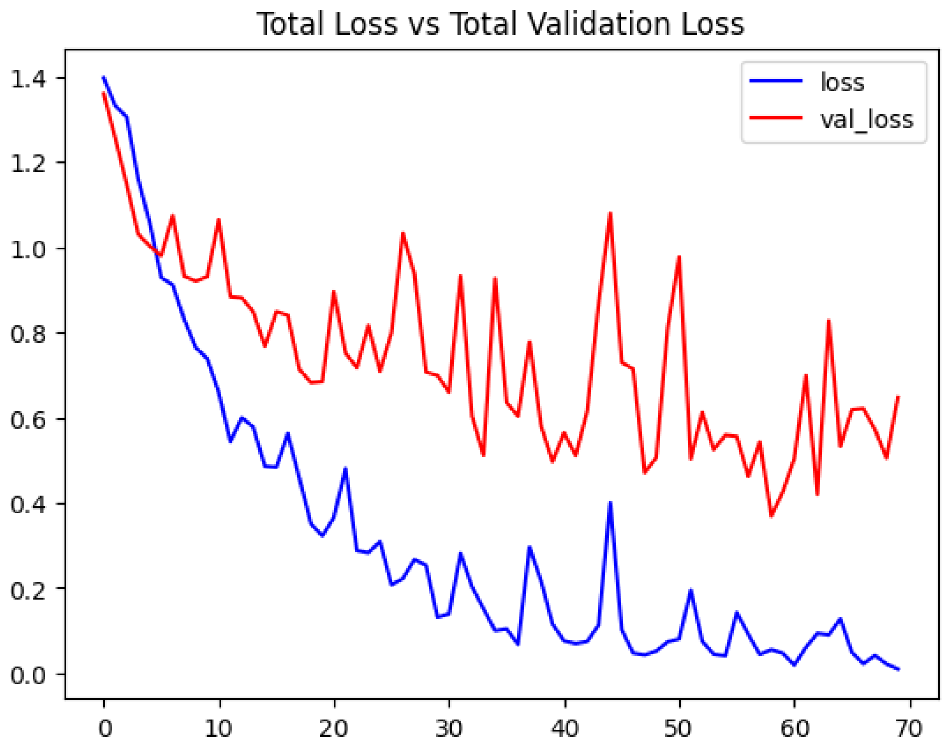

Figure 5: LRCN training and validation loss curves.

Metric	ConvLSTM	LRCN
Test Accuracy	79.25%	86.93%
Test Loss	0.5773	0.2854
Training Stability	Stable	Stable
Strengths	Temporal continuity	Frame-level spatial patterns
Weaknesses	Similar-looking activities	Minimal misclassifications

Table 1: Comparative evaluation of ConvLSTM and LRCN models.

Overall, while both architectures proved effective for human activity recognition on the selected subset of the UCF50 dataset, the LRCN model achieved higher accuracy and lower loss, establishing itself as the more effective approach in this experiment.

8 Conclusion

This project explored the application of deep learning architectures for human activity recognition using video sequences from the UCF50 dataset. Two models were developed and evaluated: the Convolutional Long Short-Term Memory (ConvLSTM) network and the Long-term Recurrent Convolutional Network (LRCN). Both architectures were designed to capture the complex spatiotemporal dependencies present in human activities, albeit through different methodological approaches.

The ConvLSTM model demonstrated reliable performance, achieving a test accuracy of 79.25% and a test loss of 0.5773. Its ability to simultaneously process spatial and tempo-

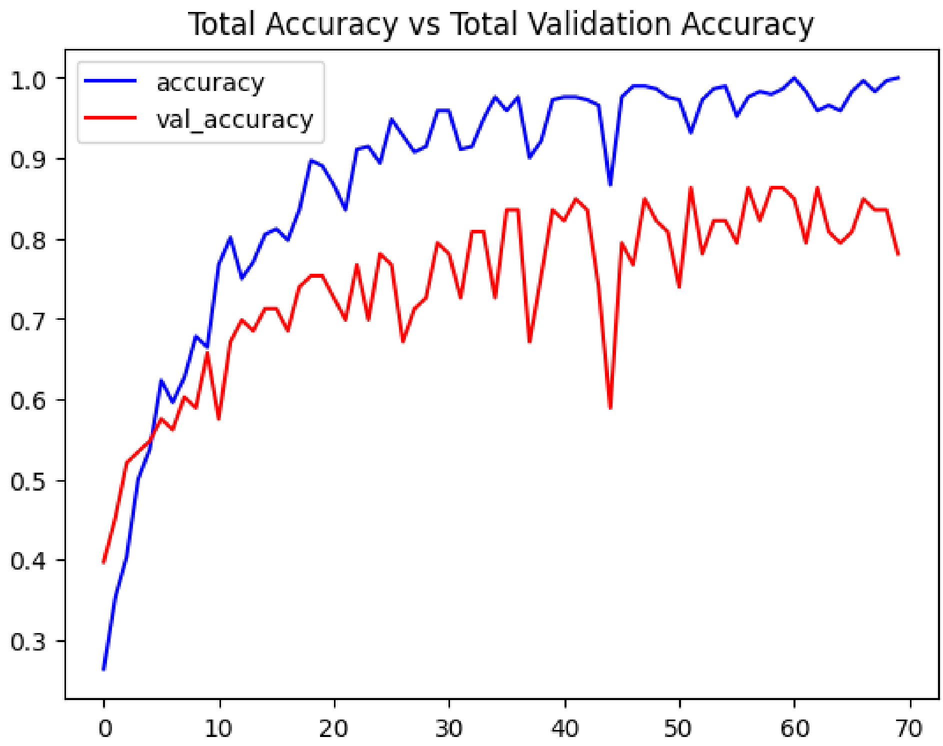

Figure 6: LRCN training and validation accuracy curves.

ral information through convolutional recurrent units allowed it to effectively recognize activities characterized by continuous motion patterns. However, it exhibited some limitations in distinguishing between activities with similar visual dynamics, such as *Swing* and *TaiChi*.

In contrast, the LRCN model achieved superior performance, with a test accuracy of 86.93% and a lower test loss of 0.2854. By decoupling spatial and temporal processing, LRCN leveraged frame-level convolutional features followed by sequential modeling with LSTM layers. This modular approach proved particularly effective in recognizing activities where individual frames contained strong spatial cues, resulting in minimal misclassifications.

The comparative analysis revealed that while both models are competent for human activity recognition, the LRCN architecture provided better generalization and overall accuracy on the selected classes from UCF50. These findings highlight the importance of architectural choices in balancing spatial feature extraction and temporal sequence modeling in video classification tasks.

Overall, this project successfully demonstrates the feasibility and effectiveness of deep learning approaches for video-based human activity recognition. The insights gained from this comparative study contribute to a deeper understanding of model behavior in dynamic visual environments and provide a solid foundation for further research in advanced spatiotemporal modeling techniques.

9 Future Work

While this project successfully demonstrated the effectiveness of ConvLSTM and LRCN architectures for human activity recognition, several avenues remain open for future exploration to further enhance model performance and applicability.

- **Dataset Expansion:** Future work will include extending the experiments to cover the full range of 50 classes in the UCF50 dataset. This will provide a more comprehensive evaluation of model scalability and robustness across a diverse set of human activities.
- **Data Augmentation:** Incorporating advanced data augmentation techniques such as random cropping, horizontal flipping, temporal jittering, and brightness adjustments could improve model generalization and mitigate overfitting, especially when scaling to larger class sets.
- **Longer Frame Sequences:** Increasing the number of frames per sequence may capture richer temporal dynamics, particularly for complex activities that unfold over longer time spans.
- **Transfer Learning:** Integrating pre-trained convolutional backbones such as VGGNet, ResNet, or MobileNet into the LRCN architecture could enhance spatial feature extraction and accelerate convergence.
- **Architectural Enhancements:** Exploring attention mechanisms or Transformer-based architectures could enable the models to dynamically focus on the most informative frames within a sequence, potentially improving classification accuracy.
- **Cross-Dataset Generalization:** Testing the models on additional benchmark datasets beyond UCF50 will help assess their generalization capabilities and robustness in varied environments.
- **Deployment Optimization:** For real-world applications, optimizing the trained models for deployment on edge devices using TensorFlow Lite or other lightweight inference frameworks will be pursued.
- **Quantitative Evaluation with Additional Metrics:** In addition to accuracy and loss, incorporating metrics such as precision, recall, F1-score, and confusion matrices will provide a more nuanced understanding of model performance across classes.

Pursuing these directions will not only enhance the performance of ConvLSTM and LRCN models but also contribute to the broader goal of advancing reliable and efficient human activity recognition systems suitable for practical deployment.

References

- Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). *Long-term Recurrent Convolutional Networks for Visual Recognition and Description*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2625–2634.

- Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., & Woo, W. C. (2015). *Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting*. Advances in Neural Information Processing Systems (NeurIPS), 28.

- Tran, D., Bourdev, L., Fergus, R., Torresani, L., & Paluri, M. (2015). *Learning Spatiotemporal Features with 3D Convolutional Networks*. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 4489–4497.

- Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). *Large-scale Video Classification with Convolutional Neural Networks*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1725–1732.

- Simonyan, K., & Zisserman, A. (2014). *Two-Stream Convolutional Networks for Action Recognition in Videos*. Advances in Neural Information Processing Systems (NeurIPS), 27.

- Soomro, K., Zamir, A. R., & Shah, M. (2012). *UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild*. arXiv preprint arXiv:1212.0402.

- Simonyan, K., & Zisserman, A. (2015). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. International Conference on Learning Representations (ICLR).