Video Action Recognition on UCF50 Dataset Using Long-term Recurrent Convolutional Networks

Dylan Costello
Assignment 4
Johns Hopkins University
Baltimore, MD, USA

***Abstract*—This paper presents an implementation and evaluation of video action recognition on the UCF50 dataset using Long-term Recurrent Convolutional Networks (LRCN). Our approach combines spatial feature extraction through a ResNet-50 backbone pretrained on ImageNet with temporal dynamics modeling via a two-layer LSTM network. We address the critical issue of data leakage in video datasets by implementing group-aware data splitting, ensuring videos from the same recording session remain in the same split. Through comprehensive experiments, we achieve a test accuracy of 72.08% with strong performance metrics including a macro F1-score of 0.709 and Cohen's Kappa of 0.715. Our analysis reveals significant variations in class-wise performance, with perfect recognition for well-defined actions like Drumming and Billiards, while complex actions involving similar motion patterns show higher confusion rates. The implementation demonstrates the effectiveness of transfer learning and temporal modeling for video understanding tasks while maintaining rigorous experimental standards for reproducibility.**

***Index Terms*—video action recognition, LRCN, UCF50, deep learning, transfer learning, temporal modeling**

I. Introduction

Human action recognition in videos represents a fundamental challenge in computer vision with applications spanning surveillance, human-computer interaction, sports analytics, and content-based video retrieval. The task requires understanding both spatial appearance and temporal dynamics, distinguishing it from static image classification problems.

The UCF50 dataset [1] provides a challenging benchmark with 50 action categories collected from YouTube videos. Unlike staged datasets with actors, UCF50 contains realistic videos with significant variations in camera motion, viewpoint, object scale, and background clutter. Critically, videos are organized into 25 groups per category, where videos within a group may share the same person, similar backgrounds, or originate from the same longer recording.

This work implements a Long-term Recurrent Convolutional Network (LRCN) [2] architecture that decouples spatial and temporal processing. We leverage transfer learning through a ResNet-50 backbone pretrained on ImageNet-1K for spatial feature extraction, combined with LSTM layers for temporal modeling. Our key contributions include:

- Implementation of group-aware data splitting to prevent data leakage, a critical but often overlooked aspect in video dataset evaluation
- Comprehensive evaluation using multiple metrics beyond accuracy, including class-wise analysis and confusion patterns
- Reproducible implementation with fixed random seeds and deterministic operations
- Analysis of performance variations across action categories, providing insights into the challenges of video understanding

II. Related Work

A. Video Action Recognition Approaches

Early approaches to action recognition relied on handcrafted features such as HOG, HOF, and MBH combined with bag-of-words representations [5]. The advent of deep learning revolutionized the field through two primary paradigms: two-stream networks [3] processing RGB frames and optical flow separately, and 3D convolutional approaches [4] directly learning spatiotemporal features.

B. LRCN Architecture

Long-term Recurrent Convolutional Networks [2] propose an end-to-end trainable architecture combining CNNs and LSTMs. The CNN extracts frame-level features while the LSTM models temporal dependencies. This approach offers flexibility in handling variable-length sequences and computational efficiency compared to 3D convolutions.

C. Transfer Learning in Video Understanding

Transfer learning from image datasets has proven highly effective for video tasks. Carreira and Zisserman [6] demonstrated that ImageNet pretraining significantly improves video model performance. Our work leverages ResNet-50 [7] pretrained on ImageNet-1K, which provides robust feature extraction for diverse visual content.

D. Data Leakage in Video Datasets

A critical but underaddressed issue in video action recognition is data leakage. When videos from the same group (sharing actors or scenes) appear in both training and test sets, models may memorize person-specific or scene-specific features rather than learning generalizable action patterns. We implement group-aware splitting to ensure all videos from a group remain in the same split, providing more realistic performance estimates.

III. Methodology

A. Dataset Preparation

The UCF50 dataset contains 6,681 videos across 50 action categories. Videos follow the naming convention `v_ActionName_g##_c##.avi`, where `g##` indicates the group number and `c##` the clip number within that group.

1) Frame Extraction: We extract 16 frames per video using uniform random sampling to capture temporal information while maintaining computational efficiency. For each video, we:

1) Divide the video into 16 equal temporal segments
2) Randomly sample one frame from each segment
3) Resize frames to 224×224 pixels for ResNet compatibility

This approach balances temporal coverage with stochasticity, preventing overfitting to specific frame positions.

2) Group-Aware Data Splitting: To prevent data leakage, we implement group-aware splitting:

Extract group IDs from video filenames
Create group-to-videos mapping
Split groups (not videos) into train/val/test
Assign videos based on their group membership

This ensures videos from the same recording session never appear in different splits. Our splits contain:

- Training: 4,675 videos from 875 groups
- Validation: 1,003 videos from 187 groups
- Test: 985 videos from 188 groups

B. Model Architecture

Our LRCN implementation consists of three main components:

1) Spatial Feature Extraction: We employ ResNet-50 pretrained on ImageNet-1K as the CNN backbone. The final classification layer is removed, yielding 2,048-dimensional feature vectors per frame. To accelerate training and prevent overfitting, we freeze early convolutional layers while fine-tuning the last 10 layers.

2) Temporal Modeling: Frame features are processed by a two-layer LSTM with 256 hidden units per layer. The LSTM captures temporal dependencies across the 16-frame sequences. We apply dropout (p=0.5) between LSTM layers for regularization.

3) Classification: The final LSTM hidden state is passed through a dropout layer and fully connected layer with 50 outputs corresponding to action categories.

C. Training Configuration

1) Optimization: We train the model using:

- Adam optimizer with learning rate 1×10^{-4}
- Cross-entropy loss
- ReduceLROnPlateau scheduler (factor=0.5, patience=5)
- Batch size of 16
- 30 training epochs

2) Data Augmentation: Training samples undergo:

- Random resized crop (scale: 0.8-1.0)
- Random horizontal flip (p=0.5)
- Color jittering (brightness=0.2, contrast=0.2, saturation=0.2, hue=0.1)

Validation and test samples use center cropping without augmentation.

3) Reproducibility: We ensure reproducibility through:

- Fixed random seeds (42) for all libraries
- Deterministic CUDA operations
- Controlled DataLoader worker initialization

IV. Experimental Results

A. Overall Performance

Table I presents the comprehensive evaluation metrics on the test set.

TABLE I: Overall Performance Metrics on UCF50 Test Set

Metric	**Score**
Accuracy	72.08%
Macro F1-Score	0.709
Weighted F1-Score	0.713
Macro Precision	0.742
Macro Recall	0.723
Cohen's Kappa	0.715
Matthews Correlation	0.716
ROC-AUC (One-vs-Rest)	0.977
ROC-AUC (One-vs-One)	0.977

The model achieves 72.08% accuracy, exceeding the 65% requirement while demonstrating balanced performance across multiple metrics. The high ROC-AUC scores indicate strong discrimination capability, while Cohen's Kappa of 0.715 shows substantial agreement beyond chance.

B. Class-wise Performance Analysis

Figure II illustrates the distribution of F1-scores across the best and worst performing classes, revealing significant performance variations.

TABLE II: Top and Bottom Performing Action Categories

Top 5 Classes	**F1-Score**	**Support**
Drumming	1.000	19
Billiards	1.000	19
Mixing	1.000	23
PlayingGuitar	1.000	24
BenchPress	0.981	26
Bottom 5 Classes	**F1-Score**	**Support**
Nunchucks	0.222	24
HulaHoop	0.174	19
JumpRope	0.169	20
Basketball	0.077	21
PizzaTossing	0.063	16

Perfect recognition (F1=1.0) is achieved for actions with distinctive visual and temporal patterns (Drumming, Billiards, Mixing, PlayingGuitar). Conversely, actions involving rapid periodic motion or similar visual appearance show poor performance.

C. Confusion Analysis

Analysis of the confusion matrix reveals systematic error patterns:

TABLE III: Most Confused Action Pairs

True Class	Predicted As	Rate	Count
Basketball	VolleyballSpiking	47.6%	10
JumpRope	PullUps	40.0%	8
YoYo	JumpRope	35.3%	6
GolfSwing	SoccerJuggling	34.8%	8
Lunges	CleanAndJerk	29.2%	7

The highest confusion occurs between actions with similar motion patterns (Basketball/VolleyballSpiking) or overlapping visual features (JumpRope/PullUps). This suggests the model struggles to distinguish fine-grained differences in human poses and object interactions.

D. Training Dynamics

The model converges after approximately 23 epochs, achieving 77.06% validation accuracy at the best checkpoint. The 5% gap between validation and test accuracy suggests mild overfitting despite regularization techniques.

E. Confidence Analysis

The model exhibits well-calibrated confidence:

- Average confidence for correct predictions: 0.906
- Average confidence for incorrect predictions: 0.589

This 31.7% confidence gap indicates the model appropriately assigns lower confidence to uncertain predictions.

V. Discussion

A. Impact of Group-Aware Splitting

Implementing group-aware splitting likely reduced our test accuracy compared to naive random splitting, but provides a more realistic evaluation. Without this approach, the model could achieve artificially high accuracy by memorizing person-specific or scene-specific features rather than learning generalizable action patterns.

B. Transfer Learning Effectiveness

The ImageNet-pretrained ResNet-50 backbone proves highly effective for video action recognition despite being trained on static images. The learned features transfer well to video frames, particularly for actions involving common objects (PlayingGuitar, Drumming) present in ImageNet.

C. Temporal Modeling Limitations

While LSTM effectively captures temporal dependencies for many actions, it struggles with:

- Periodic actions with variable frequency (JumpRope, HulaHoop)
- Actions differentiated by subtle temporal nuances
- Long-range dependencies exceeding the 16-frame window

D. Class Imbalance Effects

Some performance variation correlates with support size, though not deterministically. Classes with fewer test samples show higher variance in F1-scores, suggesting the need for stratified sampling or class-weighted loss functions.

VI. Limitations and Future Work

A. Current Limitations

- Fixed 16-frame sampling may miss critical action moments
- No optical flow information limits motion understanding
- Limited temporal context for long-duration actions
- Class imbalance not explicitly addressed during training

B. Proposed Improvements

1) **Two-stream architecture**: Incorporate optical flow for explicit motion modeling
2) **Attention mechanisms**: Replace or augment LSTM with temporal attention
3) **3D CNNs**: Explore C3D or I3D for joint spatiotemporal learning
4) **Adaptive sampling**: Learn optimal frame sampling strategies
5) **Multi-scale temporal modeling**: Process videos at multiple temporal resolutions

VII. Conclusion

This work successfully implements video action recognition on UCF50 using LRCN architecture, achieving 72.08% test accuracy while maintaining rigorous experimental standards. Our group-aware data splitting prevents leakage and provides realistic performance estimates. The analysis reveals both strengths and limitations of combining spatial CNN features with temporal LSTM modeling.

Key findings include perfect recognition for visually distinctive actions, systematic confusion between similar motion patterns, and the critical importance of preventing data leakage in video datasets. The implementation serves as a solid baseline for video understanding tasks while highlighting areas for future improvement.

The code and trained models are available for reproducibility, with all random operations controlled through fixed seeds. This work demonstrates that effective video action recognition can be achieved through careful combination of transfer learning, temporal modeling, and rigorous experimental methodology.

Acknowledgments

We thank the creators of the UCF50 dataset for providing a challenging benchmark for video action recognition research.

References

[1] K. K. Reddy and M. Shah, "Recognizing 50 human action categories of web videos," *Machine Vision and Applications*, vol. 24, no. 5, pp. 971–981, 2013.

[2] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2625–2634.

[3] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[4] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4489–4497.

[5] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3551–3558.

[6] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.