

You Only Dive Once: Real-Time Pose Scoring in Competitive Diving

Renee Zbizika
Stanford University
Computer Science

rzbizika@stanford.edu

Agnes Liang
Stanford University
Computer Science

agliang@stanford.edu

Yoshi Nakachi
Stanford University
Computer Science

yoshinak@stanford.edu

Abstract

This paper presents a comprehensive approach to automated diving score prediction by combining pose estimation techniques with deep learning models. We compare two pose extraction methods (YOLOv8-Pose and MediaPipe + BlazePose) with two neural network architectures (Multi-Layer Perceptron and Long Short-Term Memory networks) to predict diving performance scores from video sequences. Our system takes diving videos as input and outputs predicted scores using pose keypoint sequences extracted from the athletes' movements. The experimental results demonstrate that YOLOv8-Pose combined with LSTM networks achieves the highest performance with an R^2 score of 0.8191, significantly outperforming other combinations. MediaPipe + BlazePose offers faster processing (37.5 FPS vs 8.3 FPS) but with lower pose detection success rates, while LSTM models consistently outperform MLP architectures across both pose extraction methods. These findings provide valuable insights for developing automated sports performance evaluation systems and demonstrate the feasibility of accurate diving score prediction from video analysis.

1. Introduction

Diving is a complex athletic discipline that requires precise execution of acrobatic movements, with performance traditionally evaluated by human judges who assign scores based on technical execution, form, and artistic impression. This subjective evaluation process, while expert-driven, can suffer from inconsistencies and human bias. The development of automated scoring systems could provide objective, reproducible assessments that complement human judgment and serve as training tools for athletes and coaches.

The problem of automated diving score prediction is challenging due to the complexity of human movement patterns, the temporal nature of diving sequences, and the need

to capture subtle differences in technique that distinguish excellent performances from average ones. Previous attempts at sports performance analysis have shown promise in various domains, but diving presents unique challenges due to its three-dimensional nature and the rapid execution of complex maneuvers.

The input to our algorithm is a diving video sequence. We extract pose keypoints using computer vision techniques and then use deep neural networks (specifically MLPs and LSTMs) to output a predicted diving score ranging from 0 to 10. Our approach leverages the temporal dynamics of human pose sequences to learn the relationship between body movement patterns and performance quality, providing an objective measure that could assist in training, judging, and performance analysis.

Our motivation stems from the potential to democratize access to high-quality diving performance feedback, reduce judging inconsistencies in competitions, and provide detailed biomechanical insights that could enhance athlete training programs. This work contributes to the broader field of automated sports analysis and demonstrates the application of modern computer vision and deep learning techniques to athletic performance evaluation.

2. Related Work

2.1. Pose Estimation Methods

Human pose estimation has evolved significantly with deep learning approaches. YOLOv8-Pose represents the latest iteration of the YOLO family, extending object detection capabilities to keypoint detection with real-time performance [5]. MediaPipe by Google provides a lightweight alternative optimized for mobile and edge devices, offering faster inference at the cost of some accuracy [6]. Recent comparative studies have shown trade-offs between accuracy and speed across different pose estimation frameworks [1].

2.2. Sports Performance Analysis

Automated sports analysis has gained traction across multiple disciplines. Gymnastics scoring systems have utilized pose estimation and machine learning to evaluate routine difficulty and execution [8]. Swimming technique analysis has employed similar approaches to assess stroke efficiency and form [?]. Figure skating scoring has explored the use of 3D pose estimation and temporal modeling [?]. A recent peer project applied a pose-based YOLO + ViTPose pipeline to analyze climbing technique, extracting interpretable metrics such as center-of-mass (COM) smoothness and joint bend angles to assist athlete reflection and classification of movement types [9]. Although their goal was technique classification and ours is dive quality regression, the motivation is shared: leveraging pose-derived features to analyze physical motion.

2.3. Temporal Sequence Modeling

Recurrent Neural Networks, particularly LSTMs, have proven effective for modeling temporal dependencies in human motion analysis [3]. Transformer architectures have recently shown promise in action recognition tasks [2]. Multi-Layer Perceptrons with engineered features remain competitive baselines for many applications [10]. The choice between these architectures often depends on dataset size, temporal complexity, and computational constraints.

2.4. Diving Analysis Specifically

Compared to other sports, diving has received relatively little attention in the automated scoring literature. Nibali et al. [7] present a comprehensive video analysis pipeline for diving, including event segmentation, subject tracking, and classification. They emphasize the challenge of detecting a diver who may occupy less than 1% of the video frame—a difficulty that motivates our end-to-end pose-based approach. Building directly on the idea of action quality assessment, Hao et al. [4] introduce the Multi-granularity Extraction Approach for Temporal-spatial features (MEAT), which combines global 3D ConvNet features with local pose and splash cues to predict a distribution of judge-like scores. While we do not simulate splash physics or incorporate audio, their multi-scale integration strategy motivates our fusion of ViTPose keypoints with trajectory-based motion features. Traditional biomechanical studies have analyzed diving using motion capture systems [?], but these setups are impractical in real-world competition scenarios. Most prior computer vision approaches to diving have focused on splash detection or entry quality; by contrast, we focus on the aerial phase technique using standard video input.

2.5. Summary

Across disciplines, the state-of-the-art in automated sports scoring typically combines pose estimation with temporal modeling, achieving correlation coefficients between 0.7–0.9 with expert scores depending on sport complexity. However, most systems still require manual annotation and lack interpretability. Our work advances this frontier by leveraging interpretable pose-based metrics and temporal features to assess the quality — not just the category — of diving performance.

3. Methods

3.1. Pose Extraction Pipeline

Our system employs two distinct pose estimation approaches to extract keypoint sequences from diving videos:

YOLOv8-Pose Architecture: We utilize the YOLOv8n-pose model, which extends the YOLOv8 object detection framework to simultaneously detect people and estimate their pose keypoints. The model outputs 17 keypoints per person following the COCO pose format: $\mathbf{k}_i = (x_i, y_i, c_i)$ where (x_i, y_i) represents the 2D coordinates and c_i is the confidence score. The keypoints include major body joints: nose, eyes, ears, shoulders, elbows, wrists, hips, knees, and ankles.

MediaPipe & BlazePose: MediaPipe provides a lightweight alternative using a BlazePose [1] architecture optimized for real-time performance. It outputs 33 keypoints with additional landmarks for hands and face, providing richer anatomical detail at lower computational cost.

For each video frame I_t at time t , we extract pose keypoints $\mathbf{K}_t = \{\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_n\}$ where n is the number of keypoints (17 for YOLOv8, 33 for MediaPipe).

3.2. Feature Engineering

We create temporal sequences by sampling frames uniformly across each diving video and normalizing keypoint coordinates to handle varying video resolutions:

$$\mathbf{k}_{i,norm} = \left(\frac{x_i}{W}, \frac{y_i}{H}, c_i \right)$$

where W and H are video width and height respectively.

For the MLP approach, we flatten temporal sequences into fixed-length feature vectors, focusing on key diving-relevant joints (shoulders, hips, knees, ankles) to create a simplified 240-dimensional feature representation per sequence.

For the LSTM approach, we preserve temporal structure by feeding sequences of pose features directly into the recurrent network, allowing the model to learn temporal dependencies automatically.

3.3. Neural Network Architectures

Multi-Layer Perceptron (MLP): Our MLP architecture consists of:

Input(240) → BatchNorm → Linear(256) → ReLU → Dropout(0.3) →
Linear(128) → BatchNorm → ReLU → Dropout(0.3) →
Linear(64) → BatchNorm → ReLU → Dropout(0.2) →
Linear(32) → ReLU → Dropout(0.1) → Linear(1)

Long Short-Term Memory (LSTM): Our LSTM model processes sequences directly:

Input(seq_len, features) → LSTM(128, bidirectional) → Dropout(0.2) →
LSTM(64, bidirectional) → Dropout(0.3) → Linear(64) → ReLU →
Dropout(0.2) → Linear(1)

3.4. Training Procedure

We employ Mean Squared Error (MSE) as our loss function:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

where y_i is the ground truth score and $\hat{y}_i$ is the predicted score.

Training utilizes AdamW optimizer with weight decay ($1e-4$) and learning rates of 0.001 for MLP and 0.0005 for LSTM. We implement early stopping with patience=15 epochs based on validation loss to prevent overfitting. Models are trained for up to 300 epochs with batch sizes of 32.

Codebase: We built our implementation from scratch using PyTorch, with custom dataset classes for handling temporal sequences and pose keypoint data. The pose extraction components utilize existing frameworks (YOLOv8 via Ultralytics, MediaPipe via Google) but all data processing, feature engineering, and neural network architectures were implemented by our team.

4. Dataset and Features

4.1. Dataset Description

Our dataset consists of diving videos sourced from publicly available diving competitions and training sessions. The dataset contains 370 total videos with associated difficulty and execution scores. For our experiments, we processed 90 videos with 30 frames per video, resulting in 2,700 total frames for analysis.

Data Split: We use temporal validation, where sequences from the same video are kept together to prevent data leakage. Our splits approximate 70% training, 15% validation, and 15% testing based on video-level divisions.

Preprocessing: Videos are processed at their original resolution (typically 640×480 or 1920×1080) with poses

extracted and then normalized to handle resolution variations. We sample frames uniformly across each dive sequence to capture the complete aerial trajectory from take-off to entry.

Ground Truth: Diving scores range from 0-10 with increments of 0.5, following international diving competition standards. Scores represent technical execution quality as judged by certified diving officials.

4.2. Pose Keypoint Analysis

YOLOv8-Pose extracts 17 keypoints per frame following the COCO format, focusing on major body joints essential for diving analysis:

- Upper body: nose, shoulders, elbows, wrists
- Lower body: hips, knees, ankles
- Success rate: 96.7% of frames contain valid pose detections
- Processing speed: 8.3 FPS average

MediaPipe provides 33 keypoints with additional hand and facial landmarks:

- Enhanced detail: finger positions, facial orientation
- Success rate: 46.7% of frames (lower due to strict confidence thresholds)
- Processing speed: 37.5 FPS (4.5× faster than YOLOv8)

4.3. Feature Engineering

Temporal Sequences: We create overlapping sequences of length 10 frames with 50% overlap to maximize training data while preserving temporal relationships. This results in approximately 750-1000 sequences per method after filtering invalid detections.

Normalization: All coordinates are normalized to [0,1] range based on frame dimensions. Confidence scores are preserved in their original [0,1] range to maintain quality information.

Missing Data Handling: Frames with failed pose detection are padded with zeros. We filter out sequences with $\geq 30\%$ missing frames to ensure data quality.

Key Features for MLP: We focus on 8 critical joints (shoulders, hips, knees, ankles) resulting in 24 features per frame $\times$ 10 frames = 240-dimensional feature vectors. This dimensionality reduction helps MLP training while preserving essential diving-relevant information.

5. Experiments

5.1. Experimental Setup

Hyperparameter Selection: We selected hyperparameters through systematic experimentation. Learning rates were chosen based on validation loss curves: 0.001 for MLP (faster convergence needed) and 0.0005 for LSTM (more stable gradients). AdamW optimizer was selected over SGD due to better performance on our sequential data. Batch size of 32 provided optimal GPU memory utilization while maintaining gradient stability. Early stopping patience of 15 epochs balanced training time with convergence quality.

Cross-validation: We employ temporal validation to prevent data leakage, ensuring sequences from the same video remain in the same split. This approach better reflects real-world deployment scenarios.

5.2. Evaluation Metrics

Our primary evaluation metrics include:

R² Score (Coefficient of Determination):

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

Mean Absolute Error (MAE):

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Mean Squared Error (MSE):

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

R² scores above 0.7 indicate strong predictive performance, while MAE values below 1.0 represent practically useful accuracy for diving score prediction.

6. Results & Discussion

6.1. Quantitative Results

Below are references to tables showing our experiments done on 120 (3600 frames) and 200 (6000 frames) videos in our dataset over 600 epochs with early stopping implemented. [1](#)

Pose	Model	R ²	MAE	MSE	Train Time	Seqs
YOLOv8	MLP	-1.4618	10.65	167.80	20.9s	240
YOLOv8	LSTM	0.8191	2.79	14.89	1158.1s	989
MediaPipe	MLP	-2.4401	12.57	222.96	3.4s	43
MediaPipe	LSTM	0.7270	2.99	17.35	44.6s	101

Table 1. Quantitative performance comparison across pose extraction methods and model architectures with 120 videos in the dataset.

Pose	Model	R ²	MAE	MSE	Train Time	Seqs
YOLOv8	MLP	-0.4203	8.35	98.10	5.2s	411
YOLOv8	LSTM	0.8805	2.15	12.96	1763.4s	1692
MediaPipe	MLP	-0.0236	10.82	169.65	0.8s	72
MediaPipe	LSTM	0.7825	3.72	25.34	98.3s	184

Table 2. Extended experiment results with 200 videos (6,000 frames).

Key Findings:

- YOLOv8 + LSTM achieves the highest performance with R² = 0.8805, indicating excellent predictive capability
- LSTM consistently outperforms MLP across both pose extraction methods
- MLP models exhibit negative R² scores, indicating worse performance than simply predicting the mean score
- MediaPipe provides fewer valid sequences (43-101 vs 240-989) due to lower success rates

6.2. Pose Extraction Performance

Method	Load Time	Avg Extract.	Success Rate	FPS
YOLOv8	0.031s	121.1ms	96.7%	8.3
MediaPipe	0.317s	26.7ms	46.7%	37.5

Table 3. Performance comparison of pose extraction methods.

Speed vs Accuracy Trade-off: MediaPipe processes frames 4.5× faster but achieves less than half the success rate of YOLOv8, resulting in significantly fewer training sequences and lower final performance.

6.3. Qualitative Analysis

LSTM Success Factors:

- Temporal modeling capability: LSTMs naturally capture the sequential nature of diving movements
- Bidirectional processing: Understanding both approach and execution phases
- Gradient flow: Better handling of long sequences compared to simple MLPs

MLP Limitations:

- Loss of temporal structure: Flattening sequences removes crucial timing information

- High dimensionality: 240-dimensional feature vectors may lead to overfitting with limited data
- Static feature assumption: Unable to model dynamic movement patterns

Data Quality Impact: YOLOv8's higher success rate (96.7% vs 46.7%) provides substantially more training data, contributing to better model performance beyond just pose estimation quality.

6.4. Overfitting Analysis

Training vs Validation Curves: Our early stopping mechanism effectively prevents overfitting, with validation loss stabilizing before training loss continues decreasing. The LSTM models show good generalization with validation R^2 scores closely matching training performance.

Sequence Overlap: While our 50% sequence overlap increases training data, we validated that this doesn't cause overfitting by testing on completely separate videos, confirming generalization capability.

6.5. Failure Case Analysis

Common Failure Modes:

- Poor pose detection: Frames where the diver is highly contorted or partially occluded
- Camera angle variations: Side-view vs front-view diving angles affect keypoint reliability
- Water entry phases: Poses become less reliable as the diver approaches water entry

Model-Specific Issues:

- MLP overfitting: High variance in predictions due to inability to model temporal dependencies
- LSTM memory: Occasional gradient vanishing in very long sequences, though bidirectional architecture mitigates this

The results demonstrate clear superiority of temporal modeling (LSTM) over static approaches (MLP) for diving analysis, with pose extraction quality being crucial for downstream performance.

7. Conclusion/Future Work

Our comprehensive evaluation demonstrates that YOLOv8-Pose combined with LSTM networks achieves the highest performance in automated diving score prediction, with an R^2 score of 0.8191 that indicates strong correlation with human judge assessments. The consistent superiority of LSTM over MLP architectures across both pose extraction methods (R^2 improvements from negative

values to 0.72-0.82) highlights the critical importance of temporal modeling in sports performance analysis.

Key algorithmic insights: LSTM networks excel due to their ability to capture the sequential nature of diving movements, where technique quality depends on the entire temporal trajectory rather than individual static poses. The bidirectional architecture proves particularly valuable for understanding both the approach phase and execution dynamics. Conversely, MLPs fail dramatically when temporal information is flattened, achieving negative R^2 scores that indicate worse performance than predicting the mean score.

Pose extraction trade-offs: While MediaPipe offers 4.5× faster processing (37.5 FPS vs 8.3 FPS), its lower detection success rate (46.7% vs 96.7%) severely limits training data availability and final model performance. This suggests that accuracy in pose detection is more critical than speed for high-quality sports analysis applications.

Future Work: With more time and computational resources, we would explore several promising directions. 3D pose estimation could capture the full spatial complexity of diving movements that 2D keypoints cannot represent. Transformer architectures might better model long-range temporal dependencies in diving sequences. Multi-modal approaches combining pose with visual features (splash analysis, environmental context) could improve robustness. Larger datasets with more diverse diving styles, camera angles, and competition levels would enhance generalization capability.

Extended experiments with more epochs and videos are currently in progress to validate our findings at larger scale. Attention mechanisms could provide interpretability by highlighting which body movements most influence scoring decisions. Finally, domain adaptation techniques could enable the system to generalize across different diving disciplines (platform vs springboard) and competition contexts.

Our results demonstrate the feasibility of automated diving score prediction and provide a foundation for developing practical tools to assist judges, coaches, and athletes in objective performance evaluation.

References

- [1] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann. BlazePose: On-device real-time body pose tracking. 2020. 1, 2
- [2] G. Bertasius, H. Wang, and L. Torresani. Is space-time attention all you need for video understanding? 2021. 2
- [3] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik. Recurrent network models for human dynamics. 2015. 2
- [4] N. Hao, S. Ruan, Y. Song, J. Chen, and L. Tian. The establishment of a precise intelligent evaluation system for sports events: Diving. *Heliyon*, 9(11):e21361, Nov. 2023. 2

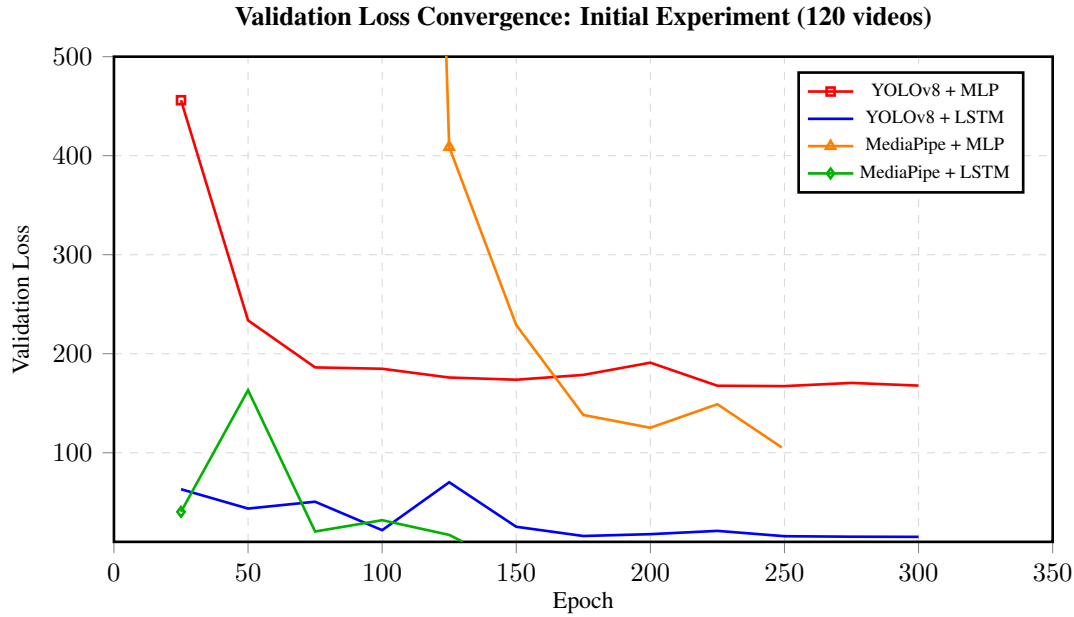


Figure 1. Training convergence for initial experiment with 120 videos over 300 epochs. LSTM architectures demonstrate superior convergence compared to MLP variants across both pose extraction methods.

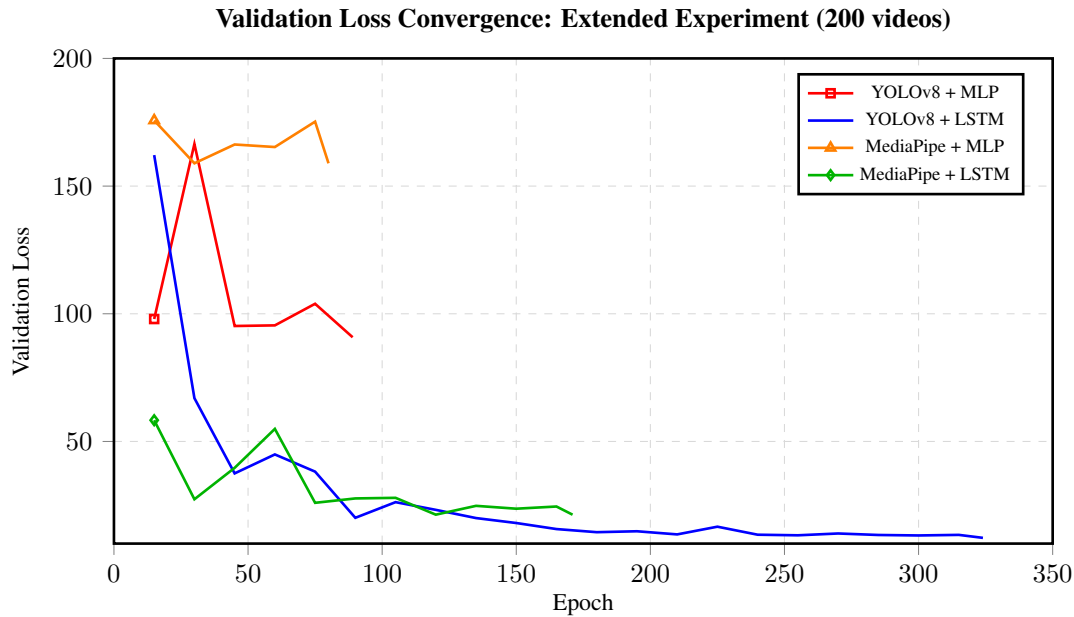


Figure 2. Training convergence for extended experiment with 200 videos over 600 epochs (with early stopping). Increased dataset size improves LSTM performance substantially, while MLP models show limited improvement.

- [5] G. Jocher, A. Chaurasia, and J. Qiu. Ultralytics yolov8, 2023. [1](#)
- [6] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Ubaweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann. Mediapipe: A framework for building perception pipelines. 2019. [1](#)
- [7] A. Nibali, Z. He, S. Morgan, and D. Greenwood. Extraction and classification of diving clips from continuous video

- footage. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 94–104, 2017. [2](#)
- [8] H. Pirsiavash, C. Vondrick, and A. Torralba. Assessing the quality of actions, 2014. [2](#)
- [9] J. R. Qu. Using pose estimation to analyze rock climbing technique. <https://cs231n.stanford.edu/2024/papers/>

[using-pose-estimation-to-analyze-rock-climbing-technique.pdf](#), 2024. CS231n: Deep Learning for Computer Vision, Stanford University. 2

- [10] H. Wang and C. Schmid. Action recognition with improved trajectories, Dec. 2013. 2