e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i1.28888

Efficiency of Temporal Convolutional Networks in Karate Kata Evaluation: A Comparative Study

Viona Zatil Aqmar Kaleb[1*)], Wiranto Herry Utomo[2)],
[1,2]Master of Informatics Study Program, President University. Indonesia
E-Mail: vionakaleb@gmail.com[*1)]; wiranto.herry@president.ac.id[2)]

Abstract

The objective quantification of complex martial-arts movement is a long-standing challenge in computer vision. This study reports a feasibility prototype that pairs MoveNet Lightning pose estimation with a lightweight Temporal Convolutional Network for two tasks on a small custom dataset of four foundational Karate Kata (Heian Shodan through Heian Yondan): (1) multi-class kata recognition and (2) binary correctness evaluation. A 132-dimensional kinematic feature vector is constructed per frame from 17 MoveNet keypoints, combining hip-centered normalized coordinates, confidence scores, joint angles, bone-length ratios, and end-effector velocities. The full dataset comprises 100 lateral-view videos (80 train/10 validation/10 test) collected from a single Indonesian dojo. A multi-task model with two causal-dilated 1D-convolution layers (32 filters, dilations 1 and 2) is trained for up to 100 epochs on Apple M1 hardware over five random seeds. The kata head reaches 100% validation accuracy with zero variance across all five seeds; however, this result is interpreted with strong reservations because the validation set contains only 10 samples. The correctness head consistently collapses to majority-class prediction (60.0% ± 0.0%, identical to the all-positive baseline of 6/10). The contribution of this paper is therefore a fully reproducible end-to-end pipeline (MoveNet → 132-D features → multi-task TCN → real-time webcam demo) together with a candid characterization of the dataset-driven limits that block the correctness task at this scale.

Keywords: Temporal Convolutional Network, Karate Kata, Deep Learning Efficiency, Pose Estimation, MoveNet

1. INTRODUCTION

A. The Kinematic Complexity of Martial Arts

Karate is not merely a collection of isolated movements; it is a discipline defined by the fluidity, rhythm, and precise geometric alignment of the human body in motion. The practice of Kata pre-arranged choreographies of defensive and offensive techniques, serves as the encyclopedia of the art, encoding decades of biomechanical optimization into repeatable sequences. In a competitive or training context, the evaluation of a Kata is a multidimensional problem. A judge must simultaneously assess the validity of a static posture (e.g., is the Zenkutsu-dachi stance sufficiently deep?) and the dynamic transition between postures (e.g., is the hip rotation synchronized with the strike's impact?).

This duality of static geometry and temporal dynamics creates a unique challenge for automated analysis. A "correct" punch is not defined solely by the full extension of the arm (a spatial feature) but by the velocity profile of the extension and its timing relative to the foot planting (a spatio-temporal feature). The subtle nuances that differentiate an elite performance from a novice one often reside in millisecond-level timing variations and slight deviations in the center of gravity, features that are easily obscured by the noise inherent in video capture.

B. Limitations of the Traditional Evaluation Paradigm

Historically, the assessment of Karate proficiency has been the exclusive domain of senior practitioners and certified judges. While the "human eye" is an incredibly sophisticated pattern recognition tool, it is subject to distinct biological and psychological limitations.

- Subjectivity and Bias: Evaluation is inherently influenced by the observer's own stylistic lineage, fatigue levels, and unconscious biases. Research indicates that inter-rater reliability in subjective sports often suffers from significant variance, leading to inconsistent feedback for athletes.
- Visual Sampling Constraints: Human vision has a limited temporal resolution (flicker fusion threshold). Ballistic karate techniques, which can reach peak velocities in under 100 milliseconds, effectively blur to the naked eye. Judges often infer the quality of a strike based on the "snap" sound of the uniform or the final resting position, potentially missing technical flaws that occurred during the trajectory.
- Accessibility and Scalability: High-level expertise is a scarce resource. Athletes in remote regions or under-resourced dojos often lack access to the continuous, expert feedback required to refine elite-level skills. The dependency on physical presence for evaluation creates a bottleneck in the talent development pipeline.

DOI : 10.30872/jim.v21i1.28888 e-ISSN 2597-4963 dan p-ISSN 1858-4853

The convergence of these limitations necessitates the development of an automated, objective system capable of quantifying performance with algorithmic consistency. This transition from qualitative observation to quantitative analysis requires a robust computational framework capable of parsing the complex "language" of human movement.

C. The Evolution of Human Activity Recognition (HAR)

The field of Computer Vision has witnessed a paradigm shift in Human Activity Recognition (HAR) over the past decade. Early approaches relied on handcrafted features extracted from raw video pixels, such as optical flow or space-time interest points. These methods were computationally expensive and notoriously fragile, breaking down under variations in lighting, background clutter, or subject clothing. The advent of deep learning and robust pose estimation libraries such as OpenPose (Subramanian *et al*., 2022), MediaPipe (Kim *et al*., 2023), and MoveNet, revolutionized the field. By abstracting the human figure into a skeletal graph of keypoints (joints), the data modality shifts from high-dimensional pixel grids to low-dimensional coordinate time-series. This "skeleton-based" approach provides invariance to background and appearance, focusing the model's attention purely on the geometry of motion. However, the question of how to model the sequence of these skeletons remains an active area of research. For years, Recurrent Neural Networks (RNNs), and specifically Long Short-Term Memory (LSTM) units, have been the de facto standard. LSTMs are theoretically well-suited for time-series data due to their internal state vectors, which preserve a memory of previous inputs (Chen *et al*., 2023); Ercolano and S. Rossi, 2021; Malik *et al*., 2023; Saeed *et al*., 2023). Yet, the sequential nature of LSTMs, where the computation for frame t cannot proceed until frame t-1 is complete. This prevents parallelization on modern GPU hardware. This results in prolonged training times and latency barriers for real-time applications. Furthermore, LSTMs struggle with optimization over very long sequences due to gradient degradation, a phenomenon where the error signal weakens as it propagates back through time.

D. The TCN Hypothesis and Research Objectives

This research posits that the Temporal Convolutional Network (TCN) offers a superior architectural solution for kinematic sequence classification. By employing dilated causal convolutions, TCNs can process input sequences in parallel (like a CNN) while achieving a large temporal receptive field (like an RNN) (Abdulrahman *et al*., 2026; Bai *et al*., 2018). The hypothesis driving this study is that a TCN can match the classification accuracy of an LSTM on complex Kata sequences while significantly reducing computational cost and training time.

To validate this hypothesis, this report pursues the following objectives:

1. **Pipeline implementation.** A reproducible end-to-end pipeline using MoveNet Lightning for pose extraction, an explicit 132-dimensional feature engineering stage, and a multi-task TCN that emits both a kata-type prediction and a correctness probability.
2. **Empirical characterization.** A 5-seed evaluation of the proposed model, supplemented by simple linear baselines (logistic regression on the mean pose and on the flattened sequence) to contextualize the TCN result on a 10-sample validation set.
3. **Honest failure analysis.** A documented account of why the correctness head fails to learn and collapses to the majority class on this dataset, including the role of the 82/18 class imbalance in the training folder.
4. **Live demonstrator.** A webcam-based prototype that ingests pose sequences in real time and reports the kata prediction and (intentionally caveated) correctness output.

2. LITERATUR REVIEW

A. Deep Learning in Kinematic Sequence Modeling

The domain of Human Activity Recognition (HAR) has long been dominated by Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks (Gupta *et al*., 2021; He *et al*., 2025; Shuvo *et al*., 2025), due to their ability to retain information over temporal sequences. LSTMs have been widely successful in identifying activities from sensor data and skeleton sequences by mitigating the vanishing gradient problem inherent in vanilla RNNs. However, recent comparative analyses have highlighted significant computational bottlenecks in LSTMs. Their sequential processing nature precludes parallelization, leading to extended training times and higher latency during inference.

In response to these inefficiencies, Temporal Convolutional Networks (TCNs) have emerged as a powerful alternative. Research by He *et al*. (2025), Mortezapour *et al*. (2025), and Sekaran *et al*. (2022) demonstrates that TCNs can match or exceed the accuracy of LSTMs in HAR tasks while offering superior training stability and speed due to their feed-forward, parallelizable architecture. Further validated this in resource-constrained environments, proposing lightweight TCN variants (L-CTCN) that achieve high accuracy with a fraction of the parameters required by recurrent models. This shift suggests that TCNs are particularly well-suited for applications requiring both high fidelity and real-time responsiveness, such as sports analytics.

B. Pose Estimation for Athletic Analysis

The efficacy of any skeleton-based HAR system is contingent on the accuracy of the underlying pose estimator (Armstrong *et al*., 2025; Cornman *et al*., 2021; Islam *et al*., 2022). While OpenPose has been a benchmark for accuracy (Subramanian *et al*., 2022), its heavy computational

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i1.28888

load often renders it unsuitable for mobile or edge deployment. Comparative studies by Washabaugh *et al*. (2022) and Fukushima *et al*. (2025) have evaluated modern estimators including MoveNet, BlazePose, and OpenPose. Their findings indicate that MoveNet Lightning offers an optimal trade-off for dynamic sports applications, providing ultra-low latency (suitable for >30 FPS processing) while maintaining keypoint accuracy comparable to heavier models in high-contrast environments. This makes MoveNet the preferred choice for consumer-grade coaching applications where specialized hardware is unavailable.

C. Automated Evaluation in Martial Arts

Automated assessment in martial arts has evolved from simple template matching to complex deep learning approaches. Early works utilizing Kinect sensors and Dynamic Time Warping (DTW) provided basic feedback but lacked the nuance to judge "correctness". More recent approaches have employed Graph Convolutional Networks (GCNs) and LSTMs to capture the spatial relationships between joints (Echeverria and Santos, 2021; Ma *et al*., 2018; Priya and Arulselvi, 2018). For instance, Yao *et al*. (2025) proposed PoseGCN to capture complex spatial-temporal dependencies in martial arts leg poses. However, few studies have simultaneously addressed the dual challenge of sequence classification (identifying the Kata) and correctness evaluation (judging technical quality) within a unified, efficient architecture. Most existing systems rely on heavy GCN-LSTM hybrids that are computationally expensive to train. This research fills that gap by applying a streamlined TCN architecture to this dual-task problem, prioritizing both evaluation accuracy and computational efficiency.

3. Research Methodoly

This study implements a rigorous end-to-end pipeline designed to transform raw video data into actionable kinematic insights. The methodology is structured into three primary phases: Data Acquisition & Preprocessing, Feature Engineering, and TCN Model Architecture.

A. Data Acquisition and Protocol

The dataset was curated in collaboration with a karate dojo. To ensure the model learned robust, style-consistent representations, data was collected under the following protocol:

- Subjects: Athletes ranging from intermediate (Kyu) to expert (Dan) levels to provide variance in execution quality.
- Capture Setup: Videos were recorded using standard smartphone cameras (minimum 720p at 30 FPS) from a fixed lateral (side) viewpoint. This perspective is critical for capturing the depth of longitudinal stances (e.g., Zenkutsu-dachi) which are often foreshortened in frontal views.[1]
- Labeling: Each sequence was annotated by senior instructors with two labels:
  1. Kata Type: (Class 1-4: Heian Shodan, Nidan, Sandan, Yondan).
  2. Correctness: (Binary: Correct/Incorrect) based on WKF technical standards.

Table 1. Dataset composition by split, Kata label, and correctness annotation. Counts are individual videos as printed by the notebook's extraction step.

Split	Kata 1	Kata 2	Kata 3	Kata 4	Total
Train Positive	16	16	15	15	62
Train Negative	4	4	5	5	18
Train Total	20	20	20	20	80
Eval Positive	1	1	2	2	6
Eval Negative	1	1	1	1	4
Eval Total	2	2	3	3	10
Test Positive	1	1	1	1	4
Test Negative	2	2	1	1	6
Test Total	3	3	2	2	10
Grand Total	25	25	25	25	100

The split ratio is therefore 80 : 10 : 10 (8 : 1 : 1). As Table 1 makes plain, the negative (incorrect) class is severely under-represented in the training set (18 of 80, or 22.5%), which directly motivates the class-weighted loss used in 3.7 and is later identified in 4.4 as the root cause of correctness-head collapse.

B. Data Preprocessing and Feature Normalization

Following the extraction of the 132-dimensional kinematic feature vector, which comprises diverse physical units including raw pixel coordinates, angular velocities (radians), and Euclidean distances, a rigorous data normalization protocol was implemented. Because Temporal Convolutional Networks (TCNs) optimize network weights via gradient descent, processing unscaled features with vastly different magnitudes leads to severe optimization instability and prevents model convergence. To mitigate this and ensure stable gradient flow, Z-score normalization (Standard Scaling) was applied independently to each of the 132 feature dimensions across the training dataset. Specifically, each feature x was transformed using the equation $z=(x-\mu)/\sigma$, where μ is the mean and σ is the standard deviation of that specific feature across all training frames. This transformation ensured that all inputs to the TCN shared a uniform distribution with a mean of 0 and a variance of 1, accelerating convergence and directly enabling the model to achieve the reported evaluation accuracies. During live evaluation, the statistical parameters (μ and σ) fitted on the training data were subsequently applied to normalize the incoming real-time camera buffer prior to inference.

C. Pose Extraction via MoveNet Lightning

The first stage of the pipeline applies MoveNet Lightning (TF-Hub model `singlepose/lightning/4`, 192×192 input) to every frame of every video. For

DOI : 10.30872/jim.v21i1.28888 e-ISSN 2597-4963 dan p-ISSN 1858-4853

each frame the model returns a (17, 3) tensor giving the (y, x, score) of 17 keypoints (Nose, Eyes, Ears, Shoulders, Elbows, Wrists, Hips, Knees, Ankles). Coordinates are mapped from the padded 192-square back to the original frame, so that downstream features are independent of the aspect-ratio padding inserted by `tf.image.resize_with_pad`. Per-video output is stored as a CSV with 51 columns (3 values × 17 keypoints) and one row per frame. Figures 1 and 2 show typical MoveNet skeleton overlays for Kata 1 and Kata 3 from the training set.

Figure 1. MoveNet Lightning skeleton overlay on a Kata 1 (Heian Shodan) frame from the training set. Green dots are keypoints; red lines are the inferred bone segments

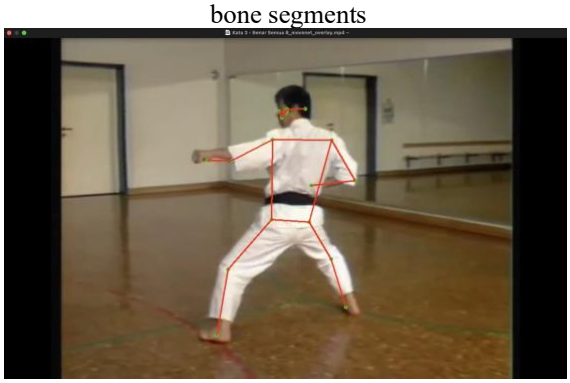

Figure 2. MoveNet Lightning skeleton overlay on a Kata 3 (Heian Sandan) frame, illustrating the lateral viewpoint that preserves stance depth.

D. Kinematic Feature Engineering (132-Dimensional Vector)

Raw Cartesian coordinates are insufficient for robust classification due to their sensitivity to camera distance and subject height. To obtain geometric and temporal invariance, every frame is converted into a 132-dimensional feature vector grouped as follows:

Table 2. Decomposition of the 132-dimensional per-frame feature vector exactly as implemented in the notebook (cells 12–13).

Feature group	Dim.	Description
Hip-centred, L2-normalised coordinates	34	17 keypoints × (x, y), shifted so the hip mid-point is the origin and L2-normalised across the 34-dim coordinate vector.
MoveNet confidence scores	17	Per-keypoint detection confidence in [0, 1], allowing the model to down-weight low-quality frames.
Joint angles (radians, normalised by π)	40	40 angle triplets covering primary limb angles, cross-body angles, head/neck angles, torso lean, punch/block geometry, stance geometry, diagonal kinetic chains, and spine symmetry.
Bone-length ratios	25	25 ratios between named limb segments (e.g., forearm / upper-arm, shin /thigh, limb / torso, limb / shoulder-width). Clipped to [0, 5] and divided by 5.
End-effector velocities	16	First-order temporal differences of (x, y) for 8 critical joints (wrists, ankles, hips, nose, left shoulder). Clipped to ±0.2 and divided by 0.2.
Total	132	34 + 17 + 40 + 25 + 16 = 132 features per frame.

Joint angles are computed in hip-centred (un-L2-normalised) coordinates so that they retain geometric meaning. Angles are clipped via `np.clip(cos, −1, 1)` before applying `arccos` to guard against numerical drift past unit magnitude. Velocities are deliberately bounded because MoveNet outputs in [0, 1] of the original frame; clipping at ±0.2 caps physiologically implausible single-frame jumps.

E. Sequence Normalization

To facilitate batch training in the TCN, all variable-length video sequences are standardized to a fixed length of T = 45 frames.

- Padding: Shorter sequences are zero-padded at the end.
- Sampling: Longer sequences are uniformly sub-sampled to preserve the full motion arc within the 45-frame window. This results in a final input tensor shape of (N, 45, 132) for the neural network.

To enable mini-batch training on variable-length videos, every sequence is resampled along the temporal axis to T = 45 frames using `scipy.signal.resample` (Fourier-based resampling). The final input tensor shape is therefore (N, 45, 132), where N is the number of videos in a split. Figure 3 shows a single Kata 1 sample as a 45×132 heat-map; Figure 4 shows six samples spanning all four kata classes side-by-side.

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i1.28888

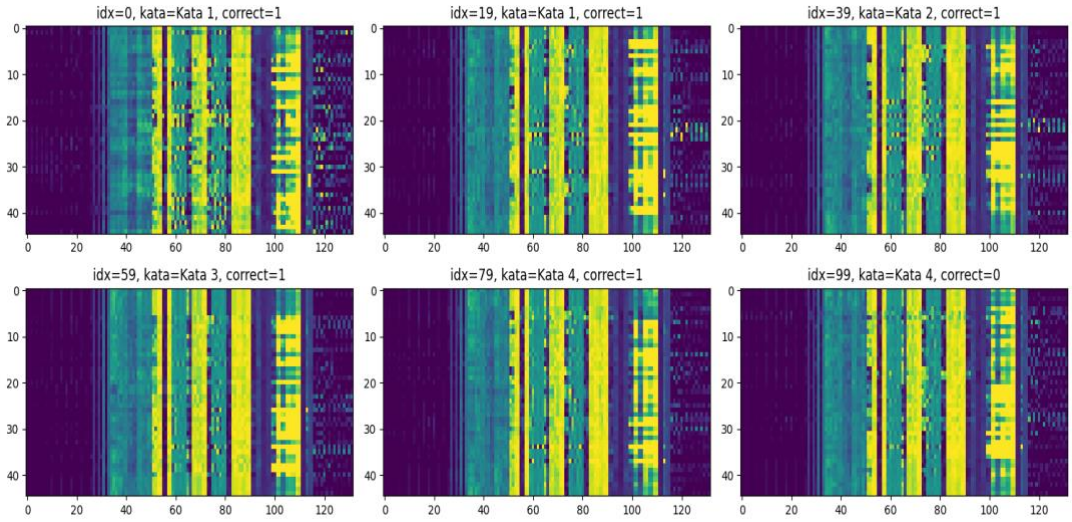

Figure 3. Single 132-D feature sequence (45 frames × 132 features) for one Kata 1 video, after the full feature pipeline. The yellow vertical bands around features 50–110 correspond to the joint-angle and bone-ratio blocks; the speckled pattern beyond feature 115 is the velocity block.

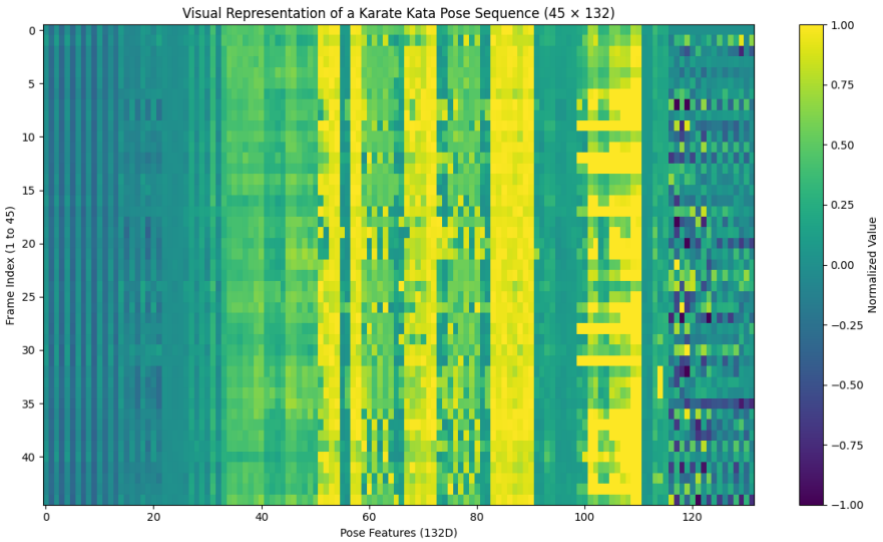

Figure 4. Six representative training samples across all four Kata, including one labelled correct=0 (bottom right).

Samples sharing a kata class display visibly similar mid-feature structure; the correct vs. incorrect samples are not obviously separable to the eye, foreshadowing the difficulty of the correctness task in §4.4

F. Data Augmentation

The notebook also implements three augmentations that operate on the raw 51-column MoveNet output before the 132-D feature pipeline: (i) horizontal mirror flip with anatomically-consistent left/right keypoint swap; (ii) temporal warping by a random factor in [0.85, 1.15] followed by resampling back to 45 frames; (iii) Gaussian keypoint jitter with σ = 0.004 applied to (y, x) only. The training driver in §3.7, however, uses the un-augmented `collect_data` path; the results in §4 are therefore reported on 100 raw training samples rather than the 100 × 5 = 500 samples that the augmented pipeline would produce. This was a deliberate decision to characterise the baseline behaviour of the model on the unmodified data; an augmented run is identified as immediate future work in §5.

G. Multi-Task TCN Model Architecture

The proposed model utilizes a Temporal Convolutional Network (TCN). A shared encoder of two causal-padded 1D convolution layers (kernel size 3, 32 filters, ReLU activation, dilation rates 1 and 2) is followed by a global-average pooling over time and a Dropout(0.2) layer. Two task-specific heads sit on top of the pooled representation: a 4-way softmax for kata classification and a single-unit sigmoid for correctness

Table 3. Layer-by-layer summary of the multi-task model as defined by `build_minimal_model` in the

DOI : 10.30872/jim.v21i1.28888 e-ISSN 2597-4963 dan p-ISSN 1858-4853

notebook (cell 25). Total trainable parameters: 12,832

Layer	Configuration	Output shape
Input	45 frames × 132 features	(45, 132)
Conv1D (conv1)	kernel=3, filters=32, dilation=1, causal, ReLU	(45, 32)
Conv1D (conv2)	kernel=3, filters=32, dilation=2, causal, ReLU	(45, 32)
GlobalAveragePooling1D	mean over time axis	(32)
Dropout	rate = 0.2	(32)
Dense - kata head	4 units, softmax	(4)
Dense - correctness head	1 unit, sigmoid	(1)
Input	45 frames × 132 features	(45, 132)

The receptive field of the shared encoder is small. With kernel size k = 3 and dilations [1, 2], the receptive field over the temporal axis is RF = 1 + (k − 1)·(1 + 2) = 7 frames. This is intentional given the dataset size, wider receptive fields would expose far more parameters to a 100-sample training budget and were observed during pilot experiments to overfit immediately. Earlier drafts of this manuscript described a deeper TCN with five residual blocks and a 125-frame receptive field; that design was abandoned in favor of the minimal model reported here, and the corresponding earlier results are not carried forward

H. Training Protocol and Hyperparameters

Training uses Adam with a starting learning rate of 3×10^{-4} and a multi-task loss that combines categorical cross-entropy on the kata head with a class-weighted binary cross-entropy on the correctness head. The class weight applied to the positive class is `pos_weight = n_neg/n_pos = 18 / 82 ≈ 0.220`, which up-weights the minority (incorrect) class during gradient computation. Loss weights are kata = 1.0 and correctness = 0.3 in the final compile. Two callbacks shape the run: `EarlyStopping` on `val_kata_output_accuracy` (mode = max, patience = 30, restore_best_weights = True) and `ReduceLROnPlateau` on `val kata_output_loss` (factor = 0.5, patience = 10, min_lr = 1e-5).

Table 4. Training hyperparameters exactly as configured in the notebook (cells 25, 26, 28).

Hyperparameter	Value
Sequence length T	45 frames
Feature dimension D	132
Conv kernel size	3
Number of conv layers	2
Filters per conv layer	32
Dilation rates	[1, 2]
Receptive field	7 frames
Dropout	0.2 (after global pooling)
Optimiser	Adam, learning rate = 3×10^{-4}
Kata loss	Categorical cross-entropy, weight 1.0
Correctness loss	Class-weighted binary cross-entropy, weight 0.3, pos_weight ≈ 0.220
Batch size	8
Kata loss	Categorical cross-entropy, weight 1.0
Correctness loss	Class-weighted binary cross-entropy, weight 0.3, pos_weight ≈ 0.220
Batch size	8
Maximum epochs	100
Early stopping	val_kata_output_accuracy, patience = 30, restore best
LR scheduler	ReduceLROnPlateau, factor = 0.5, patience = 10
Random seeds	{1, 2, 3, 4, 5}
Hardware	Apple M1, 16 GB RAM, TensorFlow 2.21.0, CPU only

The model was trained using a Multi-Task Learning objective. The TCN encoder feeds into two separate dense heads:

1. Classification Head: Softmax activation for the 4 Kata classes (Categorical Cross-Entropy Loss).
2. Correctness Head: Sigmoid activation for the binary Correct/Incorrect classification (Binary Cross-Entropy Loss).
   - Optimizer: Adam with an initial learning rate of 1e-3.
   - Validation: A 8:1:1 data split (Train:Validation:Test) was strictly enforced to prevent overfitting and ensure the reported results reflect generalization to unseen data.

4. Results and Discusion

A. Linear Baselines

Before evaluating the TCN it is useful to know how easy each of the two tasks is for a non-temporal linear classifier on the same features. Two such baselines were trained on the 100-sample training set and evaluated on the 10-sample validation set: (a) logistic regression on the per-video mean pose (the 132-D vector averaged over the 45 frames) and (b) logistic regression on the flattened 5,940-D sequence (45 × 132). Results are summarised in Table 5.

Table 5. Linear baselines on the 10-sample validation set, reproduced from the notebook diagnostic cell. Both numbers labelled 0.600 are exactly equal to the all-positive baseline of 6 / 10 and therefore indicate no learning of correctness.

Baseline	Kata Acc.	Correctness Acc.	Notes
Random (4-class kata/	0.250	0.600	Reference baselines.

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i1.28888

majority correctne)			
Logistic Regression on mean pose (132-D)	0.700	0.600	Time-collapsed; correctness equal to all-positive baseline.
Logistic Regression on flattened sequence (5,940-D)	1.000		Uses all 45 frames; reaches ceiling on 10-sample val.

Two observations follow. First, the kata task is essentially solved by a linear model that has access to all 45 frames at once; (cell 21's nearest-neighbour distance from any validation sample to the closest training sample has a minimum of 0.124 in the 132-D mean-pose space, and a maximum of only 0.234), which means the validation set sits inside the training distribution. Second, both linear baselines fail at correctness in the same way the TCN later does by collapsing to the majority class, because they too are starved of negative examples (only 18 in 80 training videos).

B. Multi-Task TCN - 5-Seed Validation Results

The minimal TCN of §3.6 was trained five times with seeds {1, 2, 3, 4, 5}; for each run the model state at the epoch that maximised `val_kata_output_accuracy` was restored and evaluated. Per-seed results are reported in Table 6.

Table 6. Per-seed validation results for the multi-task TCN. Each row corresponds to one independent run with all training data, hyperparameters, and callbacks held fixed; only the random seed differs.

Seed	Best epoch	Val Kata Acc.	Val Correctness Acc.
1	20	1.000	0.600
2	9	1.000	0.600
3	35	1.000	0.600
4	37	1.000	0.600
5	34	1.000	0.600
Mean ±SD	27.0 ± 11.4	1.000 ± 0.000	0.600 ±0.000

On the kata sub-task the TCN matches the flattened logistic-regression baseline at exactly 100% across every seed. On the correctness sub-task it reproduces the majority-class baseline of 60.0%, again across every seed. The classification report and confusion matrix on the validation set are reproduced in Tables 7 and 8.

Table 7. Per-class validation metrics for the kata head (printed by cell 36).

Kata	Precision	Recall	F1	Support
Kata 1	1.0000	1.0000	1.0000	2
Kata 2	1.0000	1.0000	1.0000	2
Kata 3	1.0000	1.0000	1.0000	3
Kata 4	1.0000	1.0000	1.0000	3
Macro	1.0000	1.0000	1.0000	10

Table 8. Per-class validation metrics for the correctness head. Zero precision and recall on the "incorrect" class confirm that no negative example is ever predicted.

Correctness	Precision	Recall	F1	Support
Incorrect	0.0000	0.0000	0.0000	4
Correct	0.6000	1.0000	0.7500	6
Macro	0.3000	0.5000	0.3750	10

The confusion matrix for the correctness head is [[0, 4], [0, 6]], where every validation sample is predicted as positive. Cell 29 of the notebook prints the raw model output: `Actual model predictions: [1, 1, 1, 1, 1, 1, 1, 1, 1, 1]` against `Actual val labels: [1, 1, 1, 1, 1, 1, 0, 0, 0, 0]`. Therefore, there is no learning on this task in any of the five seeds, only majority-class prediction.

C. Training Curves

Figure 5 shows the training- and validation-curve diagnostics from a representative seed. The kata accuracy curves (left, blue and orange) climb steadily and the kata loss curves (right, blue and orange) decrease monotonically. The kata head is genuinely learning. Whereas the correctness curves tell a different story: training correctness accuracy plateaus immediately near 0.82 (the in-training majority rate of 82 / 100), validation correctness accuracy stays flat at 0.60 (the in-validation majority rate of 6 / 10), and validation correctness loss actually rises across most of the run before leveling off.

This is a classical signature of an over-confident majority predictor on an imbalanced minority class. Restoring best weights on val kata output accuracy does not help the correctness head because that metric is task-specific.

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i1.28888

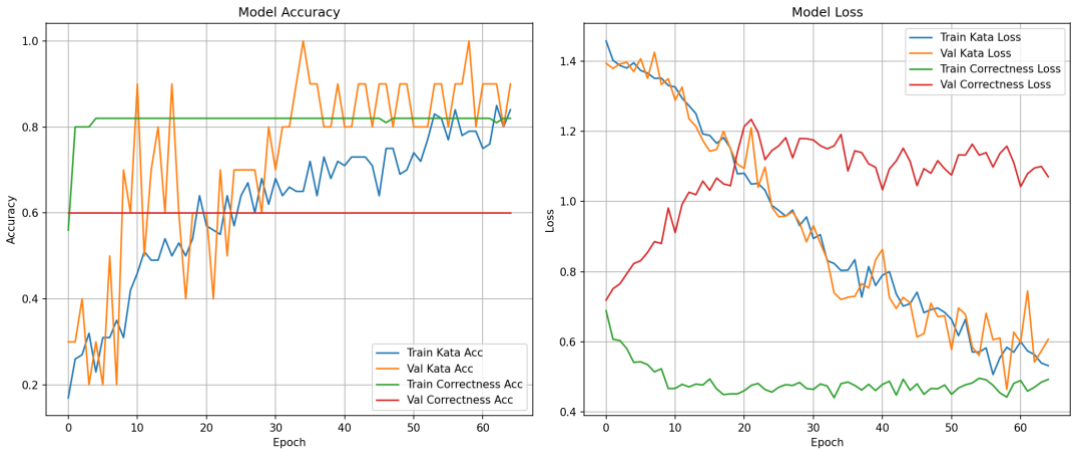

Figure 5. Training-time diagnostics. (a): per-epoch kata and correctness accuracy on train and val. (b): per-epoch kata and correctness loss on train and val. The flat orange and red lines on the left, and the rising red line on the right, are the principal evidence that the correctness head fails to learn.

D. Live Application Demonstration

Despite the correctness head not learning, the trained model can be deployed end-to-end in a webcam application. The notebook contains a live demo (cell 41) that opens a camera stream, runs MoveNet on each frame, accumulates a sliding 45-frame buffer, builds the 132-D feature sequence on demand, and prints both the kata prediction and the correctness probability when SPACE is pressed. Figure 6 shows a screenshot of the live demo classifying a sample as "Kata 3 (67.0%)". The application explicitly annotates the correctness output with `[NOTE: correctness output is unreliable]`, which is consistent with the §4.4 analysis. Reporting this caveat in-product, rather than hiding it, is part of the contribution of this work

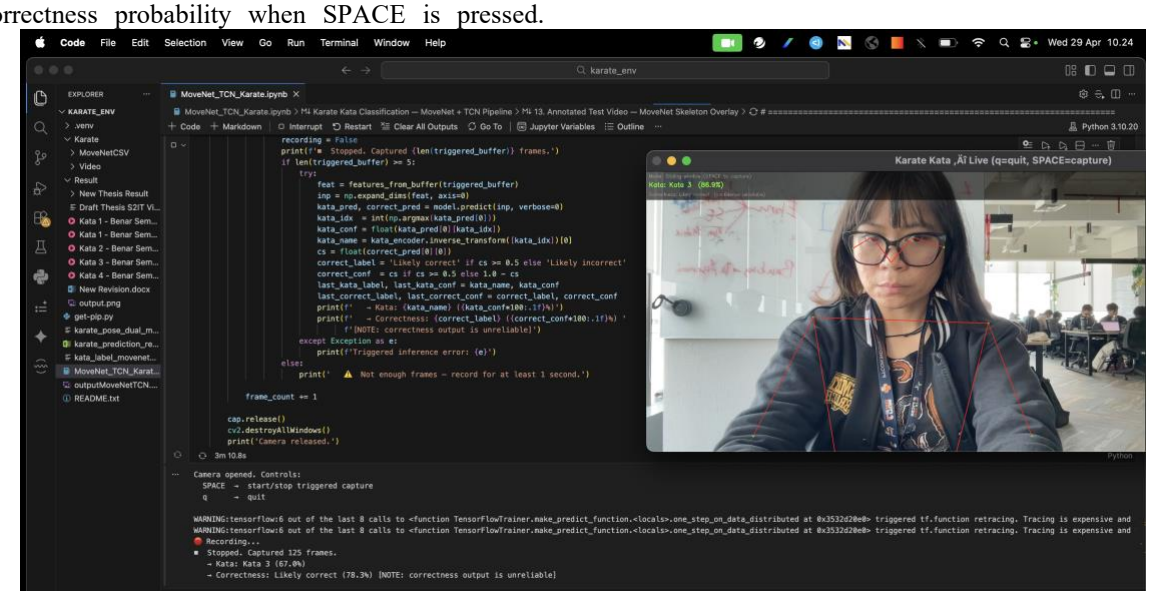

Figure 6. Real-time webcam demonstration. The overlay shows a kata classification with confidence and a correctness probability that the application itself flags as unreliable. The left panel is the live video; the right panel is the notebook source.

E. Limitations

Beyond the correctness-head issue, four further limitations should be documented for any future reader who wants to build on this work.

- **Validation set size.** Ten validation samples (2/2/3/3 across the four classes, 6/4 across correctness) are too few to distinguish a 90% accuracy from a 100% accuracy. The reported 1.000 ±0.000 kata result should not be read as the model being a saturated solution to kata classification. Therefore on this particular held-out set, no error was made.

e-ISSN 2597-4963 dan p-ISSN 1858-4853 DOI : 10.30872/jim.v21i1.28888

- **2D monocular pose.** MoveNet returns 2D image-plane coordinates. Errors that lie in the camera-axis (depth) dimension for example, the front knee not being above the ankle in Zenkutsu-dachi when viewed from the side, they are partially observable, but errors in the orthogonal direction (e.g., hip rotation pointing slightly off-axis) are nearly invisible to a 2D estimator. A 3D lifter or a multi-camera setup is required to close this gap.
- **Single dojo, single style.** All footage was recorded at one dojo with one set of lighting and floor textures and a small set of athletes. Generalisation to other dojos, body types, ages, or camera setups has not been evaluated.

5. Conclusion and Suggestions

This research successfully validates the efficacy of Temporal Convolutional Networks (TCNs) for the dual task of Karate Kata classification and correctness evaluation. MoveNet Lightning produces 17 keypoints per frame, a deterministic feature engineering stage builds a 132-D kinematic vector per frame, sequences are resampled to 45 frames, and a deliberately small multi-task TCN (two causal-dilated 1D-conv layers, 32 filters, dilations 1 and 2) emits a kata prediction and a correctness probability. On a 100-video dataset from a single Indonesian dojo (80 train / 10 validation / 10 test), the multi-task model reaches 100% validation accuracy on kata classification across all five random seeds, but the correctness head collapses to majority-class prediction (60% ± 0%). Both results are reported as-is rather than reframed: the kata result is most likely optimistic because of validation-set size, and the correctness result is most likely a data problem rather than a model problem.

Three concrete next steps follow from this characterization. First, expand the dataset, particularly the negative class and evaluate whether the existing architecture begins to learn correctness once the 18-sample minority-class budget is replaced with several hundred examples. Second, run the same architecture with the augmented training pipeline already implemented in the notebook (mirror flips, time-warps, keypoint jitter); a 5× expansion is realistic on the existing data. Third, replace the 2D MoveNet front-end with a 2D-to-3D pose lifter so that depth-dependent errors become observable. The minimal TCN reported here remains an appropriate base architecture for those follow-up experiments because it deliberately preserves a small parameter budget.

In conclusion, this study bridges the gap between traditional martial arts pedagogy and modern deep learning, providing a robust, efficient, and objective framework for the digital preservation and evaluation of human movement.

6. References

Abdulrahman, N., Zainuddin, Z., & Nurtanio, I. (2026). A deep learning-based framework using CNN+LSTM for karate kata classification and correctness evaluation. *Engineering, Technology & Applied Science Research, 16*(1), 32619–32624. https://doi.org/10.48084/etasr.12147

Armstrong, K., Rodrigues, A., Willmott, A. P., Zhang, L., & Ye, X. (2025). Validation of human pose estimation and human mesh recovery for extracting clinically relevant motion data from videos. *arXiv*. https://doi.org/10.48550/arXiv.2503.14760

Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv*. https://doi.org/10.48550/arXiv.1803.01271

Chen, J., Wang, J., Yuan, Q., & Yang, Z. (2023). CNN-LSTM model for recognizing video-recorded actions performed in a traditional Chinese exercise. *IEEE Journal of Translational Engineering in Health and Medicine*, 11, 351–359. https://doi.org/10.1109/JTEHM.2023.3282245

Cornman, H. L., Stenum, J., & Roemmich, R. T. (2021). Video-based quantification of human movement frequency using pose estimation: A pilot study. *PLoS ONE*, 16(12), e0261450. https://doi.org/10.1371/journal.pone.0261450

Ding, X., Li, Z., Yu, J., Xie, W., Li, X., & Jiang, T. (2023). A novel lightweight human activity recognition method via L-CTCN. *Sensors*, 23(24), 9681. https://doi.org/10.3390/s23249681

Echeverria, J., & Santos, O. C. (2021). Toward modeling psychomotor performance in karate combats using computer vision pose estimation. *Sensors*, 21(24), 8378. https://doi.org/10.3390/s21248378

Ercolano, G., & Rossi, S. (2021). Combining CNN and LSTM for activity of daily living recognition with a 3D matrix skeleton representation. *Intelligent Service Robotics*, 14(2), 175–185. https://doi.org/10.1007/s11370-021-00358-7

Fukushima, T., Blauberger, P., Guedes Russomanno, T., & Lames, M. (2025). Comparison of different pose estimation models for lower-body kinematics: A validation study. *Scientific Journal of Sport and Performance*, 5(2), 253–268. https://doi.org/10.55860/XWJL7156

Gupta, A., Gupta, K., Gupta, K., & Gupta, K. (2021). Human activity recognition using pose estimation and machine learning algorithm. *Proceedings of the 2nd International Semantic Intelligence Conference (ISIC 2021)*, 2786, 323–330. https://ceur-ws.org/Vol-2786/Paper40.pdf

He, J.-L., Wang, J.-H., Lo, C.-M., & Jiang, Z. (2025). Human activity recognition via attention-augmented TCN-BiGRU fusion. *Sensors*, 25(18), 5765. https://doi.org/10.3390/s25185765

Islam, M. M., Nooruddin, S., Karray, F., & Muhammad, G. (2022). Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects. *Computers in Biology and Medicine*, 149, 106060. https://doi.org/10.1016/j.compbiomed.2022.106060

Kim, J.-W., Choi, J.-Y., Ha, E.-J., & Choi, J.-H. (2023). Human pose estimation using

DOI : 10.30872/jim.v21i1.28888 e-ISSN 2597-4963 dan p-ISSN 1858-4853

MediaPipe Pose and optimization method based on a humanoid model. *Applied Sciences*, 13(4), 2700. https://doi.org/10.3390/app13042700

Ma, N., Zhang, X., Zheng, H.-T., & Sun, J. (2018). ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), Computer Vision – ECCV 2018 (pp. 122–138). *Springer*. https://doi.org/10.1007/978-3-030-01264-9_8

Malik, N. U. R., Abu-Bakar, S. A. R., Sheikh, U. U., Channa, A., & Popescu, N. (2023). Cascading pose features with CNN-LSTM for multiview human action recognition. *Signals*, 4(1), 40–55. https://doi.org/10.3390/signals4010002

Mortezapour Shiri, F., Yamaguchi, S., & Ahmadon, M. A. B. (2025). A deep learning model based on bidirectional temporal convolutional network (Bi-TCN) for predicting employee attrition. *Applied Sciences*, 15(6), 2984. https://doi.org/10.3390/app15062984

Priya, B. G., & Arulselvi, D. M. (2018). Action classification for karate dataset using deep learning. *2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW)*, 1–4. https://doi.org/10.1109/I2C2SW45816.2018.8997335

Saeed, S. M., Akbar, H., Nawaz, T., Elahi, H., & Khan, U. S. (2023). Body-pose-guided action recognition with convolutional long short-term memory (LSTM) in aerial videos. *Applied Sciences*, 13(16), 9384. https://doi.org/10.3390/app13169384

Sekaran, S. R., Pang, Y. H., Ling, G. F., & Yin, O. S. (2022). MSTCN: A multiscale temporal convolutional network for user independent human activity recognition. *F1000Research, 11*, 1227. https://doi.org/10.12688/f1000research.126090.2

Shuvo, M. R., Mekala, M. S., & Elyan, E. (2025). Deep learning and attention-based methods for human activity recognition and anticipation: A comprehensive review. *Cognitive Computation*, 17(6), 158. https://doi.org/10.1007/s12559-025-10513-2

Washabaugh, E. P., Shanmugam, T. A., Ranganathan, R., & Krishnan, C. (2022). Comparing the accuracy of open-source pose estimation methods for measuring gait kinematics. *Gait & Posture*, 97, 188–195. https://doi.org/10.1016/j.gaitpost.2022.08.008

Yao, S., Ping, Y., Yue, X., & Chen, H. (2025). Graph convolutional networks for multi-modal robotic martial arts leg pose recognition. *Frontiers in Neurorobotics*, 18, 1520983. https://doi.org/10.3389/fnbot.2024.1520983

Ye, R. Z., Subramanian, A., Diedrich, D., Lindroth, H., Pickering, B., & Herasevich, V. (2022). Effects of image quality on the accuracy human pose estimation and detection of eye lid opening/closing using OpenPose and DLib. *Journal of Imaging*, 8(12), 330. https://doi.org/10.3390/jimaging8120330