Spatial-Temporal Graph Kronecker Convolutional Network with Cauchy Activation Function: Trustworthy Skeleton-Based Tennis Stroke Recognition Architecture

Pawel Powroznik[1,*], Maria Skublewska-Paszkowska[1], Marcin Barszcz[1], Krzysztof Dziedzic[1] and Sylwester Korga[1]

[1] *Lublin University of Technology, Nadbystrzycka 38D, 20-618 Lublin, Poland*

Abstract

Vision-based sensors provide versatility in detecting a broad range of movements making the Human Action Recognition models under constant limelight. This study proposes an innovative, trustworthy Spatial-Temporal Graph Kronecker Convolutional Neural Network with a Cauchy activation function for tennis movement identification. Instead of the typical convolution, the Kronecker operation is proposed, which reduces computational complexity as well as improves the accuracy of detection. Moreover, the most effective parameters for the Cauchy activation function are established during the experiments. The model is verified utilising a trustworthy evaluation protocol combining standard metrics, SHAP interpretability, model calibration, model generalisation with cross-dataset testing on THETIS, Tennis-Mocap and 3DTennisDS, model calibration employing Expected Calibration Error, and robustness model. The ablation study proved the Cauchy to be the most adequate activation function for the ST-GCN-based models. The proposed architecture enhances the performance of tennis stroke recognition based on motion capture data, providing a trustworthy architecture.

Keywords

ST-GCN, Kronecker convolution, Cauchy activation function, SHAP values, Motion Capture, THETIS, Tennis-Mocap, 3DTennisDS, Trustworthy AI

1. Introduction

Human Action Recognition (HAR), a part of Computer Vision, has been widely applied for understanding human behaviours, including motion analysis, human-computer interaction, as well as intelligent and medical monitoring [1, 2]. Apart from video data, motion capture (MoCap) systems have gained great importance due to very accurate data. Although recording these kinds of precise movements is difficult and their post-processing is highly time-consuming, publicly available datasets are still being developed. They consist of various data differing in their recording methods, marker characteristics, and frequency. The vision-based sensors provide versatility in detecting a broad range of movements making the HAR models under constant limelight [3], also by developing architectures for sport purposes. Tennis stroke recognition has been of great interest recently, providing models involving video-based, vision-based sensors, markerless systems, as well as wearable devices. The THETIS data have been widely identified by Dynamic Time Warping with k-NN, Long Short-Term Memory network, InceptionV3, InceptionResNetV2, ResNet152V2 [4, 5, 6], CNN-LSTM, Xception, and the 3-layered LSTM [7]. The LSTM or Spatio-Temporal Graph Convolutional Network (ST-GCN) with an attention mechanism for extracting the most significant features were applied with great success to tennis movement classification [8, 9]. Multimodal approaches for tennis movement detection involving various types of MoCap data have also been widely applied [10, 11], including Deep Neural Networks, Decision Trees, and SVMs [12, 13]. Extensive studies have involved the detection of forehand, backhand, volley forehand and volley

TRUST-AI: The Second European Workshop on Trustworthy AI. Organized as part of the International Joint Conference on Artificial Intelligence - IJCAI/ECAI 2026. August 2026, Bremen, Germany.
*Corresponding author.
p.powroznik@pollub.pl (P. Powroznik); maria.paszkowska@pollub.pl (M. Skublewska-Paszkowska); m.barszcz@pollub.pl (M. Barszcz); k.dziedzic@pollub.pl (K. Dziedzic); s.korga@pollub.pl (S. Korga)
0000-0002-5705-4785 (P. Powroznik); 0000-0002-0760-7126 (M. Skublewska-Paszkowska); 0000-0002-3428-9805 (M. Barszcz); 0000-0003-4909-9980 (K. Dziedzic); 0000-0002-1327-0320 (S. Korga)

backhand captured using a passive optical Vicon system and gathered into the 3DTennisDS dataset [14], utilising the adjusted ST-GCN [15, 16] and their modifications with attention modules to enhance models performance [17, 8]. DL models struggle to accurately represent the dynamic nature of MoCap data, including both short-term and long-term changes. Effectively extracting information from various scales, particularly during dynamic movements, is another issue to solve. Thus, developing trustworthy, robustness and more accurate HAR model is still a challenging task. The traditional convolution operation needs a large amount of data for effective training. Moreover, as the step size increases, the feature map becomes smaller, which obscures lower-level details [18]. For MoCap data, obtaining an appropriately large dataset is highly time-consuming and expensive. Therefore, this study focuses on applying Kronecker Convolution to the ST-GCN to investigate how this modification influences tennis movement performance classification. In tennis shot recognition, we understand reliability as a system that supports a coach's or player's decision-making. The proposed model can support feedback on training, monitor its technical aspects, or indicate movements requiring additional attention. An incorrect, poorly calibrated, or noise-sensitive model can provide misleading recommendations, overemphasise abnormal movement patterns, or reduce the practical value of automated training analysis. Therefore, in this work, reliability is considered at five complementary levels: i) predictive reliability, ii) interpretability at the marker level, iii) generalisation across datasets, iv) probabilistic calibration, and v) robustness to noise. The main contributions of this study are as follows:

- We propose the new Spatio-Temporal Graph Kronecker Convolutional Neural Network (ST-GKCNN) with various Cauchy activation functions (CAF) for tennis forehand, backhand, and volley identification. Instead of the standard spatial convolution, we introduce a Kronecker-parameterised convolution, which provides a structured and parameter-efficient alternative for data-limited classification.
- We assess the developed model utilizing trustworthy evaluation protocol that combines 1) accuracy, precision, recall, and F1-score metrics, 2) model's interpretability using SHAP visualization, 3) model generalization with cross-dataset testing on THETIS, Tennis-Mocap and 3DTennisDS, 4) model calibration employing Expected Calibration Error (ECE), and 5) evaluation of the robustness model. We perform a set of experiments with CAF utilising 6 types of parameters.

2. Materials and Methods

2.1. Datasets

The **THETIS** dataset [19] consists of data registered by Microsoft's Kinect MoCap system at 60 Hz. They were gathered from 31 amateurs and 24 professional tennis players. In this study, the ONI files were used that define 15 points of the tennis player model (Fig. 4). The **Tennis-Mocap** dataset [10] consists of data captured from 17 athletes (5 high performers and 12 regular ones) in the Caldas-Colombia tennis league using the 6-camera optical passive Optitrack Flex V100, with the frequency set to 100 Hz. Each participant had 34 markers attached to the body, defined by Arena software. Each participant performed the strokes as similar as it was possible to those on tennis court. The **3DTennisDS** dataset [14] includes strokes from 10 tennis players, recorded at 100 Hz using an 8-camera passive optical Vicon motion capture system with a 39-marker Plug-in Gait model. In total, 208 backhand, 170 forehand, 121 volley forehand, and 121 volley backhand samples were used.

2.2. Problem definition and preliminaries

The task considered in this work is skeleton-based classification of tennis strokes. Each motion sample is represented as a temporal sequence of markers assigned to one of four stroke classes: forehand, backhand, volley forehand, or volley backhand. Since the datasets used in this study were recorded with various motion-capture systems and feature different marker configurations, we use a common graph-based notation to describe the model's input.

A motion sequence is represented as a tensor $f_{in} \in \mathbb{R}^{C\times T\times N}$, where C denotes the number of coordinate channels, T is the number of frames, and N is the number of joints or markers used by a given dataset after preprocessing. A graph encodes the anatomical structure of the body $G = (V, E)$, where V is the set of body joints or markers and E denotes their connections. An adjacency matrix represents these connections $A \in \mathbb{R}^{N\times N}$. In this way, the same model formulation can be used for all datasets, even though the original MoCap systems differ in the number and placement of markers.

Graph Convolutional Networks are suitable for this type of data because they model not only the temporal evolution of the movement, but also the spatial relations between connected body parts. In a standard graph convolution layer, the input features are propagated through the normalised skeleton graph and then transformed by learnable weights. This operation can be written as $f_{out} = \sigma\left(\Lambda^{-\frac{1}{2}}(A+I)\Lambda^{-\frac{1}{2}} f_{in} W\right)$, where I is the identity matrix, Λ is the degree matrix used for normalization, W stands for a learnable weight matrix, and σ is a nonlinear activation function. This formulation provides the basis for the ST-GCN architecture, which extends graph convolution with temporal convolution in order to capture the dynamics of tennis strokes across consecutive frames [17, 20, 21, 22].

2.3. Kronecker convolution

Kronecker convolution (KC) is used in this study as a structured parameterisation of the convolutional weights. Instead of learning a dense convolution kernel directly, the kernel is represented by smaller factors combined through the Kronecker product. This type of decomposition can reduce the number of learnable parameters and introduce an additional inductive bias, which is particularly useful in data-limited motion-capture problems. In the proposed model, KC replaces the standard spatial convolution inside the ST-GCN block, while the temporal convolution and residual connections follow the original ST-GCN design.

For two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$, the Kronecker product is defined as $A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix} \in \mathbb{R}^{mp\times nq}$, where each element a_{ij} of matrix A scales the entire matrix B. This operation allows a larger weight matrix to be represented by smaller components. A related coefficient interaction can also be illustrated for two polynomials ($f(x) = \sum_{i=0}^{n} f_i x^i$) and ($g(x) = \sum_{j=0}^{m} g_j x^j$), whose product is given by

$$f(x)g(x) = \sum_{i=0}^{n}\sum_{j=0}^{m} f_i g_j x^{i+j}. \tag{1}$$

This illustrates how pairwise interactions between smaller sets of coefficients can generate a richer representation [23]. In a standard convolutional layer, the output for an input tensor X, a convolution kernel W, and a bias b is $out = X * W + b$, where $*$ denotes the convolution operation. In a Kronecker-parameterised convolution, the kernel W is approximated by one or more Kronecker products of smaller kernels: $W \approx \sum_{r=1}^{R} W_1^{(r)} \otimes W_2^{(r)}$. For a single Kronecker component, this gives $out = X * (W_1 \otimes W_2) + b$, where W_1 and W_2 are smaller sub-kernels [24]. More generally, using R components increases the expressive capacity of the decomposition while still preserving a structured form of the kernel.

This parameterisation is beneficial when the number of parameters in the Kronecker factors is smaller than in the corresponding dense kernel. For example, if a dense matrix $W \in \mathbb{R}^{m_1 m_2\times n_1 n_2}$ is represented as $W_1 \otimes W_2$, where $W_1 \in \mathbb{R}^{m_1\times n_1}$ and $W_2 \in \mathbb{R}^{m_2\times n_2}$, the number of learnable parameters changes from $m_1 m_2 n_1 n_2$ to $m_1 n_1 + m_2 n_2$. Therefore, KC should be understood here primarily as a structured and parameter-efficient replacement for the standard spatial convolution, rather than as an unconditional guarantee of lower runtime. Its practical computational cost (Tab. 5) depends on the implementation and the selected decomposition rank. Previous studies have shown that Kronecker-based convolution can be effective in deep learning models, especially when efficient feature extraction are required [18].

2.4. Cauchy-based activation function

The Cauchy approximation for an analytic function $f : \mathbb{C}^N \to \mathbb{C}$ is obtained by replacing the surface integral in the Cauchy formula with a finite weighted sum. Taking the real part gives a representation (Eq. (2)) that is valid when the target function is real-valued [25]:

$$f(x_1, x_2, \ldots, x_N) \approx \mathrm{Re}\left[\sum_{k=1}^{m} \frac{\lambda_k}{(\xi_{k,1} - x_1)\ldots(\xi_{k,N} - x_N)}\right], \qquad (x_1, x_2, \ldots, x_N) \in \mathbb{R}^N. \tag{2}$$

Here, every weight $\lambda_k \in \mathbb{C}$ and each pole $\xi_{k,j} \in \mathbb{C}$ has a strictly positive imaginary part, which keeps the denominators away from zero. In the single-variable case $(N = 1)$ Eq. (2) reduces to Eq. (3):

$$f(x) \approx \mathrm{Re}\left[\sum_{k=1}^{m} \frac{\lambda_k}{\xi_k - x}\right], \qquad x \in \mathbb{R}. \tag{3}$$

Moreover, each complex coefficient and pole can be written as $\lambda_k = u_k + iv_k$ and $\xi_k = a_k + id$, where $u_k, v_k, a_k, d \in \mathbb{R}$ and $d > 0$. The real part of the single-pole Cauchy term is then given by:

$$\mathrm{Re}\left(\frac{\lambda_k}{\xi_k - x}\right) = \frac{u_k(a_k - x) + v_k d}{(a_k - x)^2 + d^2}. \tag{4}$$

For $a_k = 0$, this expression becomes

$$\mathrm{Re}\left(\frac{\lambda_k}{\xi_k - x}\right) = \frac{-u_k x + v_k d}{x^2 + d^2}. \tag{5}$$

Using the reparameterisation $\lambda_1 = -u_k$ and $\lambda_2 = v_k d$, the Cauchy activation function used in this work can be written as

$$\phi(x) = \frac{\lambda_1 x + \lambda_2}{x^2 + d^2}, \qquad d > 0. \tag{6}$$

In the experimental tables, λ_1 and λ_2 denote the reparameterised coefficients of the implemented activation function. The coefficient λ_1 scales the odd (linear) component of the map, λ_2 provides an even offset that shapes the response near the origin, while d dilates the denominator and thereby controls both the width of the kernel and the smoothness of the activation. An example of a Cauchy function depending on trainable parameters is depicted in Fig. 2.

2.5. Model architecture

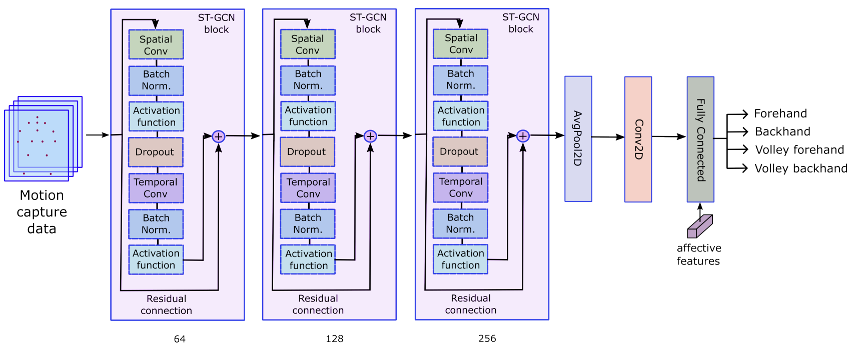

Figure 1: The proposed ST-GCN model for input motion capture data.

Our model is built on the Spatial-Temporal Graph Convolutional Network (ST-GCN) architecture (see Fig. 1). It classifies four tennis moves: forehand, backhand, forehand volley, and backhand volley. The network has three stacked ST-GCN blocks. Each block mixes spatial graph convolution with temporal convolution, and includes batch normalisation, an activation function, dropout, and a residual shortcut for stability. For the spatial part, we test two options: the standard graph convolution and a new Kronecker-based convolution (see Sec. 2.3). For time modelling, we use a 1D convolution with a kernel size of 9, yielding a joint space-time feature map with 256 channels. We compare several activation functions: ReLU, Sigmoid, Tanh, Swish, GeLU, Leaky ReLU, and Cauchy. Dropout is applied before temporal convolution to reduce overfitting, and residual connections improve gradient flow and training stability. The final part of the network consists of adaptive averaging pooling, a fully connected layer with 256 units, and a Softmax layer that returns class probabilities for four stroke categories. Mathematically, the single-block operation of ST-GCN with Kronecker convolution is given by Eq. (7):

$$H^{(l+1)} = \sigma\left(BN\left(KronConv\left(A, H^{(l)}, W^{(l)}\right)\right)\right), \tag{7}$$

where $H^{(l)}$ is the data representation from layer l, A is the symmetric adjacency matrix of the graph, $W^{(l)}$ denotes the weight matrix, while σ symbolizes the selected activation function, and BN is the batch normalization operation.

3. Experiments and results

In order to verify the potential of the created model, a series of experiments are performed applying various settings of the ST-GKCNN with the CAF model. MoCap datasets, THETIS, the Tennis-Mocap, and 3DTennisDS, are compared in order to verify how various types of data acquisition affect the accuracy of the HAR. 5-fold cross validation is applied to eliminate the randomness in the results and ensure the optimal selection of model parameters. The model was implemented in PyTorch and evaluated on a workstation with an AMD Ryzen Threadripper PRO 7965WX CPU and a single NVIDIA RTX 4090 GPU. It was trained for up to 100 epochs using Adam with learning rate 10^{-3}, $\beta_1 = 0.9$, $\epsilon = 10^{-8}$, batch size 16, a five-epoch warm-up, learning-rate decay by 0.1 after epoch 40, and early stopping after five epochs without validation improvement. We assessed the ST-GKCNN with and without CAF model utilizing trustworthy evaluation protocol that combines 1) accuracy, precision, recall, and F1-score metrics, 2) model interpretability using SHAP visualization, 3) model generalization with cross-dataset testing on THETIS, Tennis-Mocap, and 3DTennisDS, 4) model calibration employing ECE, and 5) evaluation of the model robustness (Tab. 4). The whole implementation is available on GitHub (https://github.com/AnonymousCodesForPapers/ST-GCNwKC).

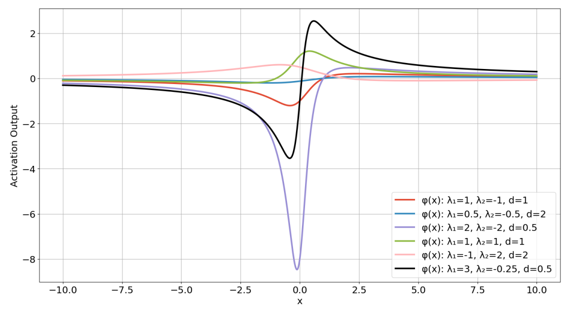

Figure 2: Cauchy activation function depending on trainable parameters λ_1, λ_2, d.

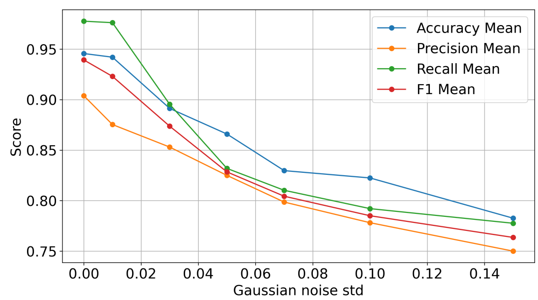

Figure 3: The impact of Gaussian noise on classification results

The results show that the model achieved high test accuracy across all datasets, while the fitted calibration temperatures were mostly below 1 (Tab. 3). This indicates that the uncalibrated model is underconfident: its predicted probabilities are lower than its empirical accuracy would suggest. Although underconfidence is often less dangerous than overconfidence in high-risk decisions, it can still reduce its practical utility. Therefore, ECE reduction should be interpreted in the context of the intended coach-in-the-loop use case.

The ST-GKCNN with CAF enhances the tennis movements recognition for each shot (Tab. 2). The transparency of the model including the importance of each marker (Fig. 5) in the final identification of

Table 1
Mean accuracy results on validation datasets for ST-GKCNN and ST-GCN with CAF model for THETIS, Tennis-Mocap, and 3DTennisDS datasets. All numerical values are given in %

Parameters	THETIS		Tennis-Mocap		3DTennisDS	
	ST-GKCNN	ST-GCN	ST-GKCNN	ST-GCN	ST-GKCNN	ST-GCN
$\lambda_1 = 1, \lambda_2 = -1, d = 1$	82.13	80.27	79.28	76.44	84.63	81.95
$\lambda_1 = 0.5, \lambda_2 = -0.5, d = 2$	89.23	85.32	90.01	88.11	92.41	90.12
$\lambda_1 = 2, \lambda_2 = -2, d = 0.5$	87.60	86.12	86.45	83.30	88.95	85.80
$\lambda_1 = 1, \lambda_2 = 1, d = 1$	92.51	89.45	92.65	89.90	94.37	92.15
$\lambda_1 = -1, \lambda_2 = -2, d = 2$	83.06	81.66	84.93	81.95	84.99	82.50
$\lambda_1 = 3, \lambda_2 = -0.25, d = 0.5$	93.60	90.77	92.95	89.40	96.85	93.90

Table 2
Obtained average results for individual classes for each of the tested databases. F stands for forehand stroke, B for backhand, VF for volley forehand, VB for volley backhand. All values in %

Dataset	Class	Kronecker Convolution				Standard Convolution			
		Acc.	Prec.	Recall	F1	Acc.	Prec.	Recall	F1
THETIS	F	93.94	97.89	93.94	95.87	91.91	96.81	91.92	94.30
	B	95.96	96.94	95.96	96.45	92.92	96.84	92.93	94.84
	VF	81.81	72.97	81.82	77.14	75.75	64.10	75.76	69.44
	VB	96.97	94.12	96.97	95.52	96.96	88.89	96.97	92.75
Tennis-Mocap	F	93.48	97.73	93.48	95.56	91.30	93.33	91.30	92.30
	B	93.62	95.65	93.62	94.63	89.36	93.33	89.36	91.30
	VF	82.61	79.17	82.61	80.85	86.96	80.00	86.96	83.33
	VB	100.00	92.00	100.00	95.83	91.31	95.45	91.30	93.33
3DTennisDS	F	97.50	97.50	97.50	97.50	97.50	95.12	97.50	96.29
	B	97.67	97.67	97.67	97.67	95.35	95.35	95.35	95.35
	VF	94.44	97.14	94.44	95.77	91.67	91.67	91.67	91.67
	VB	97.22	94.59	97.22	95.89	94.44	97.14	94.44	95.76

forehand, backhand, and volley clearly indicates that the key features are aligned with the body parts the most involved in the correct execution of these movements. The developed model achieved high MoCap data generalisation (Tab. 6). Utilising accuracy results, it can be stated that the best Cauchy activation function parameters are: $\lambda_1 = 3$, $\lambda_2 = -0.25$ and $d = 0.5$ (Tab. 1). In order to verify the potential of this function among others, six most commonly applied functions are utilised (Tab. 3). The obtained results confirm that the Cauchy activation function finds the greatest application in classification of MoCap data, both for the standard and Kronecker operations, significantly improving its effectiveness by 3.35%. The model's robustness was verified by assessing how the test data degradation quality using Gaussian noise influences on the tennis stroke recognition performance (Fig. 3). Gaussian noise was added independently to skeleton coordinates, while the graph topology was kept unchanged. The results show graceful but non-negligible degradation as noise increases. The probabilistic reliability of the four-class Kronecker-based classifier was assessed using the top-label ECE. For each sample, the predicted label was obtained as the class with the maximum softmax probability, and the corresponding confidence was defined as this maximum probability. The confidence interval $[0, 1]$ was divided into ten equal-width bins. For each bin, empirical accuracy and average confidence were computed, and ECE was calculated as: $\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} |\mathrm{acc}(B_m) - \mathrm{conf}(B_m)|$, where B_m denotes the set of samples assigned to the m-th confidence bin, n is the total number of samples, $\mathrm{acc}(B_m)$ is the empirical accuracy in the bin, and $\mathrm{conf}(B_m)$ is the mean confidence. Temperature scaling was used as a post-hoc calibration method.

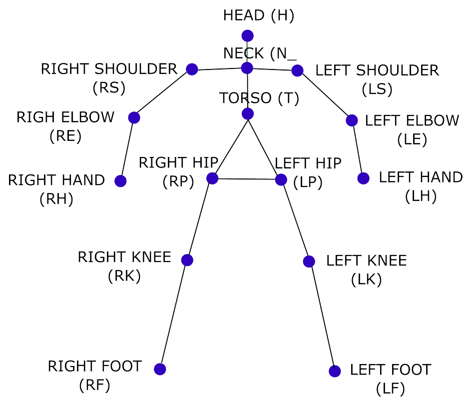

Figure 4: THETIS tennis player model with body key positions

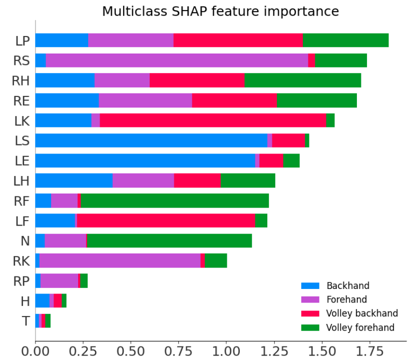

Figure 5: Markers average impact on tennis movements identification utilizing SHAP

Table 3
The influence of the selection of activation functions and the type of convolution on the quality of classification of MoCap data on the example of tennis strokes (Mean values for all datasets). All values are given in %

Convolution	Function	Acc.	Prec.	Recall	F1	ECE Before	ECE After	T
Kronecker	ReLU	87.90	86.83	88.22	87.26	0.1596	0.0858	0.744
	Sigmoid	80.50	76.48	77.44	76.01	0.2069	0.1753	0.864
	Tanh	82.20	80.99	83.13	81.27	0.1716	0.1468	0.903
	Swish	90.86	89.23	91.53	90.01	0.2013	0.1304	0.748
	GeLU	78.90	78.06	79.89	77.80	0.2047	0.1491	0.801
	Leaky ReLU	87.10	85.13	87.17	85.84	0.1967	0.1092	0.660
	Cauchy	94.71	93.35	94.87	94.03	0.1600	0.0687	0.688
Standard	ReLU	84.80	83.53	85.28	82.81	0.1742	0.1323	0.874
	Sigmoid	76.64	75.39	77.32	75.00	0.2218	0.2125	1.072
	Tanh	79.39	77.83	80.02	77.66	0.1716	0.1468	0.903
	Swish	87.32	85.48	87.71	86.25	0.2069	0.2066	1.001
	GeLU	76.46	76.88	79.08	76.59	0.2430	0.2305	0.940
	Leaky ReLU	84.32	82.29	84.60	82.94	0.2222	0.1578	0.813
	Cauchy	91.36	89.27	91.22	90.07	0.1995	0.1273	0.726

Table 4
Trustworthiness-oriented evaluation protocol

Reliability	Acc., Prec., Rec., F1
Generalization	Cross-dataset testing
Robustness	Gaussian coordinate noise
Calibration	ECE, temperature scaling
Interpretability	SHAP marker importance
Human-centred	Coach-in-the-loop support

Table 5
Computational cost and accuracy of ST-GCN variants

Model	Params	FLOPs	Time [ms]	Acc.
ST-GCN	1.20 M	0.82 G	0.34	84.80
ST-GKCNN	0.84 M	0.66 G	0.32	87.90
ST-GCN+CAF	1.20 M	0.82 G	0.35	91.36
ST-GKCNN+CAF	0.84 M	0.66 G	0.33	94.71

The temperature parameter was fitted on the validation set by minimising the negative log-likelihood and then applied to the independent test set. Since the outputs were stored as softmax probabilities, calibration used the probability-space form $\hat{p}_i^{(T)} = \frac{\hat{p}_i^{1/T}}{\sum_{j=1}^{K} \hat{p}_j^{1/T}}, \quad K = 4$. The ST-GKCNN with CAF achieved a mean accuracy above 92% (Tab. 2). Differences across datasets may result from marker placement and MoCap system precision. Compared with ST-GCN-based architectures, the proposed Kronecker convolution with Cauchy activation improved tennis stroke recognition performance (Tab. 7).

Table 6
ST-GKCNN cross-dataset testing

Training dataset	Testing dataset	Acc.	Prec.	Recall	F1
THETIS	Tennis -Mocap	87.3 ±0.6	89.8 ±0.7	89.1 ±0.5	89.5 ±0.7
	3DTennisDS	89.8 ±0.8	88.3 ±0.7	90.9 ±0.5	89.6 ±0.5
Tennis -Mocap	THETIS	88.4 ±0.8	89.9 ±0.7	87.2 ±0.9	88.6 ±0.8
	3DTennisDS	90.8 ±1.2	87.5 ±1.1	87.8 ±1.9	87.7 ±1.0
3DTennisDS	Tennis -Mocap	90.7 ±0.8	90.9 ±0.6	90.8 ±0.7	90.9 ±0.7
	THETIS	88.6 ±1.4	89.6 ±0.8	88.6 ±0.9	89.0 ±0.9

Table 7
Comparison with the state-of-the-art

Dataset	Model	Acc.
THETIS	ST-GCN	88.9±0.7
Tennis-Mocap		84.9±0.9
3DTennisDS		84.5±1.0
THETIS	4s-ShiftGCN	87.0±0.6
Tennis-Mocap		85.2±0.7
3DTennisDS		88.6±0.7
THETIS	A3T-GCN	87.3±0.6
Tennis-Mocap		89.0±0.7
3DTennisDS		91.8±0.7
THETIS	FFGCAN	92.3±3.1
Tennis-Mocap		92.3±0.8
3DTennisDS		95.1±0.7

4. Conclusions

This paper proposes ST-GKCNN with Cauchy activation for MoCap-based tennis stroke recognition. The model enhances the recognition performance compared to standard ST-GCN variants and shows promising cross-dataset generalisation. It also provides calibration and SHAP-based interpretability evidence under a trustworthiness-oriented protocol. But it should be considered as a coach-in-the-loop decision-support component and not as an autonomous assessment system. The primary limitations are the synthetic nature of the datasets, weaker recognition of volley forehand, sensitivity to stronger coordinate perturbations, and the need to tune the Cauchy parameters $(\lambda_1, \lambda_2, d)$. Future work will include subject-independent external validation, markerless pose-estimation settings, and uncertainty-aware feedback.

Declaration on Generative AI

During the preparation of this work, the author(s) used Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] K. Host, M. Ivašić-Kos, An overview of human action recognition in sports based on computer vision, Heliyon 8 (2022).
[2] P. Wang, Research on sports training action recognition based on deep learning, Scientific Programming 2021 (2021) 3396878.
[3] G. Bhola, D. K. Vishwakarma, A review of vision-based indoor har: state-of-the-art, challenges, and future prospects, Multimedia Tools and Applications 83 (2024) 1965–2005.
[4] J. Cai, J. Hu, X. Tang, T.-Y. Hung, Y.-P. Tan, Deep historical long short-term memory network for action recognition, Neurocomputing 407 (2020) 428–438.
[5] S. Vinyes Mora, Computer vision and machine learning for in-play tennis analysis: framework, algorithms and implementation, Ph.D. thesis, Imperial College London, 2019.
[6] S. Vinyes Mora, W. J. Knottenbelt, Deep learning for domain-specific action recognition in tennis, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2017, pp. 114–122.

[7] A. Sen, S. M. M. Hossain, R. Uddin, K. Deb, K.-H. Jo, Sequence recognition of indoor tennis actions using transfer learning and long short-term memory, in: International Workshop on Frontiers of Computer Vision, Springer, 2022, pp. 312–324.
[8] M. Skublewska-Paszkowska, P. Powroźnik, M. Barszcz, K. Dziedzic, Dual attention graph convolutional neural network to support mocap data animation, Advances in Science and Technology. Research Journal 17 (2023).
[9] M. Ullah, M. M. Yamin, A. Mohammed, S. D. Khan, H. Ullah, F. A. Cheikh, Attention-based lstm network for action recognition in sports, Electronic Imaging 33 (2021) 1–6.
[10] C. K. Valencia-Marin, J. D. Pulgarin-Giraldo, L. F. Velasquez-Martinez, A. M. Alvarez-Meza, G. Castellanos-Dominguez, An enhanced joint hilbert embedding-based metric to support mocap data classification with preserved interpretability, Sensors 21 (2021) 4443.
[11] W. Yang, Precise recognition and feature depth analysis of tennis training actions based on multimodal data integration and key action classification, IEEE Access (2025).
[12] A. Ganser, B. Hollaus, S. Stabinger, Classification of tennis shots with a neural network approach, Sensors 21 (2021) 5703.
[13] K. Ma, A real time artificial intelligent system for tennis swing classification, in: 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), IEEE, 2021, pp. 000021–000026.
[14] M. Skublewska-Paszkowska, P. Powroznik, E. Lukasik, J. Smolka, Tennis patterns recognition based on a novel tennis dataset–3dtennisds, Advances in Science and Technology. Research Journal 18 (2024).
[15] M. Skublewska-Paszkowska, P. Powroznik, Temporal pattern attention for multivariate time series of tennis strokes classification, Sensors 23 (2023) 2422.
[16] M. Skublewska-Paszkowska, P. Powroznik, E. Lukasik, Learning three dimensional tennis shots using graph convolutional networks, Sensors 20 (2020) 6094.
[17] M. Skublewska-Paszkowska, P. Powroznik, E. Lukasik, Attention temporal graph convolutional network for tennis groundstrokes phases classification, in: 2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), IEEE, 2022, pp. 1–8.
[18] K. K. Patro, J. P. Allam, B. C. Neelapu, R. Tadeusiewicz, U. R. Acharya, M. Hammad, O. Yildirim, P. Pławiak, Application of kronecker convolutions in deep learning technique for automated detection of kidney stones with coronal ct images, Information Sciences 640 (2023) 119005.
[19] S. Gourgari, G. Goudelis, K. Karpouzis, S. Kollias, Thetis: Three dimensional tennis shots a human action dataset, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2013, pp. 676–681.
[20] J. Ma, Y. Zhang, H. Zhou, H. Yang, X. Wu, Multi-granularity spatial temporal graph convolution network with consecutive attention for human motion prediction, Applied Soft Computing 165 (2024) 112126.
[21] L. Shi, Y. Zhang, J. Cheng, H. Lu, Skeleton-based action recognition with directed graph neural networks, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 7912–7921.
[22] S. Yan, Y. Xiong, D. Lin, Spatial temporal graph convolutional networks for skeleton-based action recognition, in: Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.
[23] Z. Tang, F. Jiang, M. Gong, H. Li, Y. Wu, F. Yu, Z. Wang, M. Wang, Skfac: Training neural networks with faster kronecker-factored approximate curvature, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 13479–13487.
[24] M. J. Ali, B. Raza, A. R. Shahid, Multi-level kronecker convolutional neural network (ml-kcnn) for glioma segmentation from multi-modal mri volumetric data, Journal of Digital Imaging 34 (2021) 905–921.
[25] X. Li, Z. Xia, H. Zhang, Cauchy activation function and xnet, Neural Networks (2025) 107375.