JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online) : 2454 -7190 Vol.-21, No.-08, August (2026) pp 156-179 ISSN (Print) 0973-8975

AN EFFICIENT MULTIMODAL FRAMEWORK FOR HUMAN ACTION RECOGNITION USING TCN AND BILSTM WITH LATE FUSION

Vijay Singh Rana[1], Ankush Joshi[2] , Kamal Kant Verma[3]

[1]Department of Computer Applications, College of Smart Computing, COER University Roorkee, India-247667.

[2] Department of Computer Applications, College of Smart Computing, COER University Roorkee, India-24766.

[3]SCSE IILM University, Greater Noida, India-201310.

Email: [1]vijayranasrb25@gmail.com, [2]ankushjoshi1987@gmail.com
[3]dr.kamalverma83@gmail.com

Corresponding Author: **Kamal Kant Verma**

https://doi.org/10.26782/jmcms.2026.08.00010

(Received: May 16, 2026; Revised: July 29, 2026; Accepted: August 08, 2026)

Abstract

In recent years, researchers have become interested in human action recognition because of the broad spectrum of its applications in surveillance, healthcare monitoring, and human-computer interaction. A highly robust, efficient, and innovative multimodal system for recognizing human actions is proposed in this study based on the Florence 3D Action dataset. The proposed system leverages the RGB, Depth, and Skeleton multi-modalities. For the case of RGB and Depth video sequences, spatial information is extracted by a pre-trained ResNet50 model, and then the extracted high-level visual features are fed to a Temporal Convolutional Network (TCN) to capture temporal patterns at a large scale. Color and depth video sequences are processed in parallel. Temporal information is also captured in the skeleton data, which is fed to a Bi-directional Long Short-Term Memory (BiLSTM) model after features are extracted in the form of joint angles, velocities, and accelerations, which describe the motion of human actions. For the case of the three modalities, the results are fused at the decision level using a weighted sum for the final prediction. The results show that the proposed system has an impressive recognition accuracy of 96.8% on the Florence 3D dataset, with stable convergence, low validation loss and overfitting, and a strong generalization capability, which meets the requirements for use in real-time, resource-constrained systems.

Keywords: HAR, ResNet50, Temporal Convolution, BiLSTM, Multimodal Fusion, RGBD, Florence 3d Dataset

I. Introduction

Human Activity Recognition has become a fundamental aspect of computer vision research, especially as the need for automated video analysis grows in fields like healthcare, smart environments, and surveillance ([XXXVIII] [XXX] [XXXI]). Initially, methods concentrated on RGB image sequences, but contemporary techniques have transitioned to utilizing multi-modal data from RGB-D sensors, such as the Microsoft Kinect, which offer synchronized color and depth data ([XXII] [XXXIII]). Depth sensors are particularly beneficial as they deliver 3D structural information that remains consistent under different lighting conditions and better protects user privacy compared to traditional video. These sensors often extract 3D skeleton data, which represents the human body through a series of joint coordinates (usually ranging from 17 to 39 joints), capturing key spatio-temporal dynamics with minimal computational demand [XVIII].

Deep learning architectures, particularly Bidirectional Long Short-Term Memory (Bi-LSTM) and Temporal Convolutional Neural Networks, have predominantly shaped the modeling of skeleton-based temporal evolutions [XIV]. Bi-LSTMs, a type of Recurrent Neural Network, are crafted to manage sequential data by employing memory cells that retain long-term dependencies [XXIX]. Over the past five to seven years, research has progressed towards hierarchical LSTM models that segment the human skeleton into separate components—like the spine, arms, and legs—processing each individually before integrating them into a comprehensive body representation [XXII]. This multi-scale strategy facilitates the examination of actions across various temporal granularities, enabling the differentiation of intricate daily activities that might exhibit similar motion patterns but vary in long-term contextual nuances [XII]. For instance, Bi-LSTM models have been enhanced with data augmentation and gradient injection methods to boost recognition accuracy, even when the training data is scarce [III].

Despite the effectiveness of RNNs, they face many challenges due to their sequential nature, such as high computational costs and difficulties in capturing long-term dependencies. TCNs use dilated curves along the time axis, allowing the model to have a much larger input field and to have access to past and future information throughout the sequence. Recent developments, such as the Res-TCN architecture, have included residual connections to enhance the interpretation of model parameters and functions, often a "black box" problem in traditional deep learning. TCN has been shown to exceed LSTM-based networks in both accuracy and training efficiency for tasks such as action segmentation and fine-grained recognition[XXIX]. In addition, modern TCN-based frameworks often use a 1×1 convolution layer as the first step for adjusting input dimensions before attempting to extract spatio-temporal features [XXXVI]. During the most recent research period (2022–2025), there has been a significant shift to hybrid architectures and attention mechanisms to further improve the performance of recognition. Models such as hybrid dual-branch networks combine various deep learning streams to model skeleton data robustly, while attention modules are used to automatically select dominant joints and assign certain degrees of importance to frames. Furthermore, the integration of transformers with hierarchical structures allows for global dependency modeling of the entire skeleton graphic [XXXII]. As the field progresses towards 2026, the focus will remain on designing effective spatio-temporal

modelling frameworks capable of handling the complexity of everyday activities and remaining computationally practical for real-time applications.

II. Literature Review

The discipline of Human Action Recognition has moved toward multimodal recognition techniques as a result of single-modality data restrictions, such as occlusion sensitivity and ambient fluctuations. Recent academic studies demonstrate that the more thorough extraction of temporal and geographical feature maps might result from the incorporation of multi-modal information such as RGB, skeleton, and depth. The adoption of effective fusing procedures is one of the main components of these contemporary approaches. Recent research has demonstrated the effectiveness of late fusion of the skeletons and RGB features. By concurrently learning intricate spatial and temporal relationships separately using optical flow-based key frame extraction and self-attention modules, late fusion enhances the model's intelligence. These modules are then combined for final classification [XXV]. By using lightweight 3D Convolutional Neural Networks (3DCNN) for space-time feature extraction in models, this modular method is further improved. These properties are utilized to train a BiLSTM network in these kinds of systems, which produce activity scores that are fused at the decision level. This specific combination has been tested on benchmark datasets such as UTKinectAction3D with an accuracy of 96.72%, and with a reduction in the complexity of the computation [XX].

There is also a paradigm shift in the architectural level of HAR, where traditional CNN-based models are giving way to more advanced sequential modeling and attention. Recent surveys have shown that CNNs have been the king of visual modeling during the past decade, and the usage of Transformer building blocks as well as efficient fusion designs has recently resulted in better performance in multimodal human action recognition [VII]. TCNs learn from the local spatial and temporal characteristics to reflect fine-grained changes in behavior, keeping the model sensitive to external changes such as changes in illumination. For instance, certain TCN prototypes have shown 100 % precision and recall on widely-used datasets for modelling these spatio-temporal features [XXXVII].

Future work on blended networks has studied the bidirectional encoding of the skeletal data using BiLSTMs, and the representation of the RGB and depth data with CNNs. A novel idea in this direction is the use of distance-based metrics such as the L2 distance metric instead of standard score fusion to select the probability distribution generated by a number of inputs. This method was proved to enhance the recognition rate up to 94.73% [VI].

In order to enhance categorization, more complex multimodal systems also use a range of pre-trained models and weighted rules. Inception-ResNet-V2 in conjunction with multi-head attention and BiLSTM for appearance information extraction and intensity images from front, side, and top perspectives for skeleton modalities are examples of such frameworks. These streams are typically merged using weighted product rules in a way to ensure that the features that are most discriminative to the final decision are incorporated more into the merging. These make use of Temporal Causal Self-Attention to ensure causal consistency in time, which leads to better discrimination of action sequences and low computation cost [XXVI]. Empirical studies executed in recent years are still verifying the superiority of multimodal data fusion over traditional

single-mode data fusion. Algorithms such as MM-CNN-LSTM have been proven to achieve performance improvements of over 15% and higher F1-scores by successfully incorporating both video and sensor data [XXIV]. This is especially true in architectures where TCN is used in conjunction with a BiLSTM to address problems such as gradient explosion and inadequate feature extraction. These architectures have been used to achieve up to 99.1% accuracy in the UCI-HAR [XVI] dataset by using a suitable-sized receptive field with TCN and a context from past and future steps with BiLSTM.

III. Proposed Methodology

This section provides a detailed discussion of the proposed three-stream fusion network for identifying human actions and their main components in the RGB-D video frame sequence containing various actions using the proposed ResNet50-TCN model. The proposed system works at three levels: First, ResNet50-TCN is used for feature extraction from a sequence of frames from RGB and depth simultaneously in the first two streams. Second, feature learning is performed using a temporal convolutional neural network (TCN). However, the skeleton data is processed by a Bi-LSTM network in the third stream. Then, finally, a feature-level fusion strategy is applied to the feature extraction from all three streams and passed through the Softmax for classification. Below is the explanation for all three streams one by one.

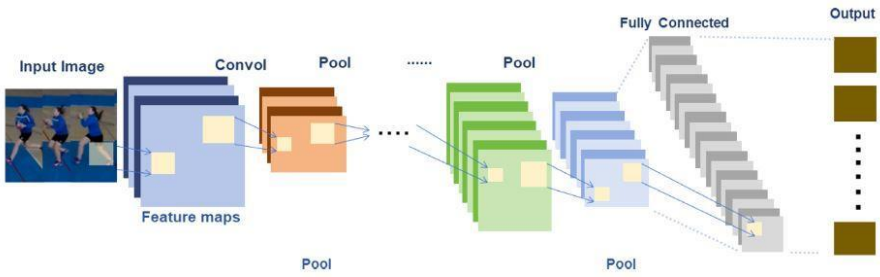

Fig. 1. represents the architecture of the ResNet50 model

III. i. Preprocessing and Spatial Feature extraction using ResNet50 from video sequences

a) Preprocessing

The actual resolution of the frames collected from RGB-D video sequences of the Florence Action 3D dataset is 640x480. The proposed ResNet50 architecture processes the RGB-D videos by resizing them into a new measurement of 224×224×3, which is the prerequisite for the model using the OpenCV library in Python. The ResNet50 operates on 3-channel RGB-D videos. In order to extract features with dimensions of 20×2048, a fixed length of 20 frames was set as an input since the suggested ResNet50 architecture runs on fixed-length input video sequences.

b) RGBD Feature Extraction using ResNet50

The ResNet-50 model has strong feature extraction abilities since it is a deep network. It has the ability to convert input images into higher-dimensional feature representations that incorporate texture and semantic information. Subsequent

computational task like analyzing motion and recognition of patterns, can make use of these feature maps. The ResNet-50 formula is as follows:

$$y_l = F(x_l, W_l) + xl\ xl + 1\ = H(y_l) \quad (1)$$

Where the input feature map is represented by x_l at the *l*th level. The output feature map is represented by $y_l$of the *l*th layer. The weight parameter is represented by $W_l$at the *l*th layer. The convolution and batch normalization joint operations are represented by F. The combined activation and downsampling operations are represented by H. The input feature map of the (l+1) layer is represented by x_{l+1}. The residual connections in the ResNet-50 model are described by these formulas. In order to obtain $F(x_l, W_l)$ in each residual block, the input feature map xl passes through convolution and batch normalization operations, which are subsequently added to the input feature map to produce the output feature map y_l. The input feature map of the subsequent layer, x_{l+1}, is then obtained by passing the output feature map through the activation function and down-sampling process. The architecture of ResNet50 is shown in Fig.1.

c) Skeleton-based Feature Extraction

In the present section, the set of relevant and observable features of the skeleton joint sequence has been extracted and made into a feature vector V_t. A feature vector (F_T)is made using three features that are based on geometric configuration, joint movement, and transitions between actions. To obtain F_T we used the joint angle feature (F_t), joint velocity feature (V_t) and joint acceleration feature (A_t). The final computed feature vector is shown below in equation (2).

$$F_T = \{F_t, V_t, A_t\} \quad (2)$$

- **Joint Angle Feature**

In this work, the human body is represented using a 20-joint skeleton model, where each joint corresponds to a specific anatomical landmark obtained from a pose estimation system. For a given frame t, the skeleton is defined as $S_t = \{J_1, J_2, J_3 \ldots\ldots\ldots\ldots J_{20}\}$, where each joint $J_i = (x_i, y_i, z_i)$ represents its spatial coordinates in 3D space. This study employs joint angle features, which capture the relative geometric relationships between connected joints and provide a more invariant and discriminative representation of human motion [XIX].

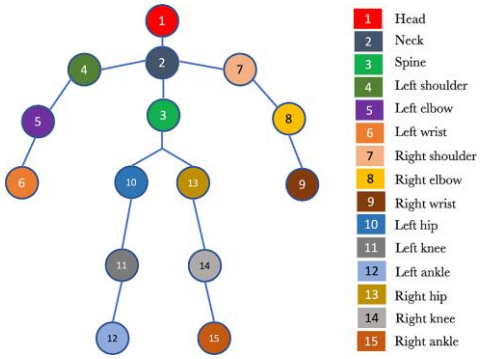

Fig. 2. shows the 15-joint skeleton generated by the Kinect V2. [X]

To extract joint angle features, a set of meaningful joint triplets is defined, where each triplet consists of three anatomically connected joints. The middle joint in each triplet acts as the pivot point at which the angle is computed. For the 20-joint skeleton model as shown in Fig. 2 used in this work, joint triplets are selected to cover all major body articulations, including the upper limbs (e.g., shoulder–elbow–wrist), lower limbs (e.g., hip–knee–ankle), and torso (e.g., hip–spine–shoulder center). Specifically, a total of fourteen joint triplets are defined to comprehensively represent the human body's motion dynamics. The selected triplets are listed in Table 1.

Table 1: Depicts the selected triplet name from the skeleton posture

S No.	Body parts	Joints number	Triplet Name
1	Upper Body	(1, 2, 3)	Head–Neck–Spine
2		(2, 3, 10)	Neck–Spine–Left Hip
3		(2, 3, 13)	Neck–Spine–Right Hip
4	Left Arm	(2, 4, 5)	Spine–Left Shoulder–Left Elbow
5		(4, 5, 6)	Left Shoulder–Left Elbow–Left Wrist
6	Right Arm	(2, 7, 8)	Spine–Right Shoulder–Right Elbow
7		(7, 8, 9)	Right Shoulder–Right Elbow–Right Wrist
8	Left Leg	(3, 10, 11)	Spine–Left Hip–Left Knee
9		(10, 11, 12)	Left Hip–Left Knee–Left Ankle
10	Right Leg	(3, 13, 14)	Spine–Right Hip–Right Knee
11		(13, 14, 15)	Right Hip–Right Knee–Right Ankle

For each triplet (J_a, J_b, J_c), where J_a is the central joint, two vectors are constructed from the skeleton coordinates: one from J_b and J_a and another from J_b to J_c. These vectors represent the direction and orientation of adjacent body segments. The angle at the central joint is then computed using the cosine similarity between these vectors, which effectively measures the degree of articulation. This formulation provides a compact representation of joint movement and is invariant to global translation and scaling of the skeleton.

Mathematically, let $\overrightarrow{v1} = J_a - J_b$ and $\overrightarrow{v2} = J_c - J_b$. The joint angle θ is computed using the cosine inverse function applied to the normalized dot product of the vectors. The angle at joint J_b is computed using the cosine similarity, as given in equation (4) below:

$$\theta = \cos^{-1}\left(\frac{\overrightarrow{v1} * \overrightarrow{v2}}{\| \overrightarrow{v1} \| [\![\overrightarrow{v2}]\!]}\right) \qquad (4)$$

The resulting angle values typically lie in the range [0, π], representing the full range of joint articulation from fully extended to fully bent configurations.

After computing the angles for all predefined joint triplets, the resulting values are concatenated to form a feature vector representing a single skeleton frame. For the 20-joint frame, this results in a 14-dimensional feature vector shown in equation (5) below:

$$F_t = \{\theta_1, \theta_2\ \theta_3 \ldots\ldots \theta_{14}\} \qquad (5)$$

To maintain consistency and improve convergence during model training, the angle values are further normalized to a fixed range, typically [0,1], by dividing each angle

by π. This normalized representation ensures that all features contribute equally to the learning process, as given in (6).

$$\theta_{norm} = \frac{\theta}{\pi} \qquad (6)$$

To capture the temporal dynamics of human actions, the frame-level feature vectors are aggregated over time to form a sequence $F = \{F_1, F_2, F_3, \ldots\ldots\ldots, F_T\}$, where T denotes the number of frames in the sequence.

- **Joint Velocity Feature**

In addition to joint angle features, joint velocity features are employed to capture the temporal dynamics of human motion. Joint velocities encode how fast each joint is moving across consecutive frames. Recent studies have shown that incorporating motion-based features such as velocity significantly improves the performance of skeleton-based action recognition systems [IV]. For a 20-joint skeleton, the velocity of each joint J_i at time t is computed as the difference between its positions in successive frames and is represented using equation (7) given below:

$$\overrightarrow{v1^t} = J_i^{t+1} - J_i^t \qquad (7)$$

The magnitude of the velocity vector is then calculated to represent the speed of motion given in equation (8)

$$\left\|\overrightarrow{v1^t}\right\| = \sqrt{(x_i^{t+1} - x_i^t)^2 + (y_i^{t+1} - y_i^t)^2 + (z_i^{t+1} - z_i^t)^2} \qquad (8)$$

After computing the velocity for all 20 joints, the values are concatenated to form a frame-level velocity feature vector mentioned in equation (9) given below.

$$V_t = [\|\overrightarrow{v_1}\|, \|\overrightarrow{v_2}\|\|\overrightarrow{v_3}\|, \ldots\ldots\ldots\ldots\ldots\ldots \|\overrightarrow{v_{20}}\|] \qquad (9)$$

This results in a 20-dimensional feature vector for each frame. To maintain consistency across sequences and improve numerical stability during training, the velocity values can be normalized using min-max normalization.

To model temporal dynamics over an entire action sequence, the frame-level velocity vectors are aggregated into a sequence $V = \{V_1, V_2, V_2, \ldots\ldots\ldots, V_{T-1}\}$, where T is the total number of frames. This sequential representation captures the evolution of joint motion and can be effectively utilized by a Temporal Convolutional Network (TCN) model.

- **Joint Acceleration Feature**

Joint acceleration features are utilized to represent the rate of change of joint velocity over time. While joint velocities describe motion speed, acceleration provides insight into variations in movement, such as sudden changes or transitions between actions. For a 20-joint skeleton, the acceleration of each joint J_i at time t is computed as the difference between consecutive velocity vectors as given in equation (10).

$$\overrightarrow{a_i^t} = \overrightarrow{v_i^{t+1}} - \overrightarrow{v_i^t} \qquad (10)$$

Substituting the velocity formulation, the acceleration can be directly expressed in terms of joint positions as mentioned in equation (11).

$$\overrightarrow{a_i^t} = J_i^{t+2} - 2J_i^{t+1} + J_i^t \qquad (11)$$

The magnitude of the acceleration vector, representing the intensity of motion change, is computed as mentioned in equation (12).

$$\left\|\overrightarrow{a_i^t}\right\| = \sqrt{(x_i^{t+2} - 2x_i^{t+1} + x_i^t)^2 + (y_i^{t+2} - 2y_i^{t+1} + y_i^t)^2 + (z_i^{t+2} - 2z_i^{t+1} + z_i^t)^2} \qquad (12)$$

The acceleration magnitudes of all 20 joints are concatenated to form a frame-level feature vector, which is then aggregated over time to model motion variations.

- **Action Score prediction via SSTCN with ResNet50 Feature for RGBD**

A TCN employs a fully-convolutional network architecture that is 1D (1-dimensional). The output of each hidden layer is the same length as the input layer, and each layer's equal length is maintained by using zero padding of length (kernel size - 1). As a result, the network is capable of processing an input sequence of any length and providing an output of the same length, which is advantageous for the action recognition task. As shown in Fig. 3, 1D convolutions are performed across the time axis in a TCN. The network accepts a 3D tensor of shape (batch_size, sequence_length, input_channels) and generates a tensor of shape (batch_size, sequence_length, input_channels) with the help of 2D kernels. Fig. 3 (a) depicts a situation in which the input channels are equal to 1, and Fig. 3 (b) shows a situation in which the input tensor has more than one channel (input channels = 2). The output is computed at each step by taking the dot product of the learnt weights of the kernel vector and the subsequence that falls inside the sliding window. One input element (i.e., stride = 1) shifts the window to the right in order to retrieve the next output element. An output sequence with the same length as the input is produced by repeating this technique throughout the full sequence. As a result, the prediction at each frame depends on a predetermined amount of time (the receptive field), meaning that a subset of the input elements affects a certain output element. Depending on the requirements of the problem, the receptive field changes.

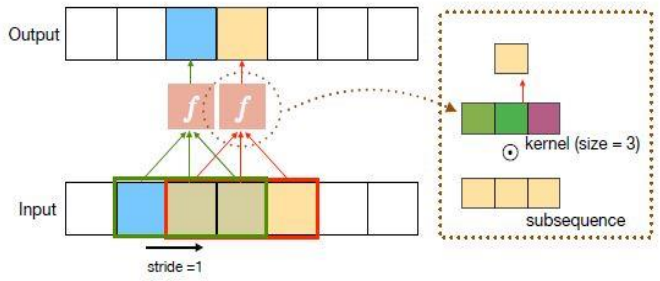

Fig. 3(a) Single-channel input processing using a TCN model

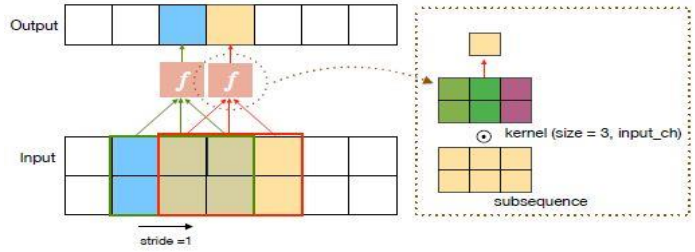

Fig. 3(b) Multi-channel input processing using a TCN model

By using dilation, the receptive field can be expanded while keeping the number of layers quite small. According to the model given in [IV], dilated convolution is sometimes referred to as convolution with holes. The receptive field size of a layer with a dilation factor of d and a kernel of size k spreads over a length of l, where l is defined as in equation (13) below:

$$l = 1 + d * (k - 1) \tag{13}$$

Fig. 4 illustrates this process using a basic convolutional layer with a receptive field of three elements and neighboring input elements selected to compute an output element for d = 1 and the kernel size = 3. If d is 2, the receptive field responds to input covering 5 elements, and when d is 4, that receptive field covers 9 elements. In practice, the dilation factor grows very quickly, which means the receptive field at each layer increases as the dilation factor is raised exponentially. For instance, the receptive field (r_field^l) at layer 1 (l€$[1, l]$ and $kernel\_size$=3 is defined by the formula given in equation (14).

$$r\_field^l = 2^{l+1} - 1 \tag{14}$$

However, because of this huge receptive field, networks can only have a small number of layers in order to prevent overfitting.

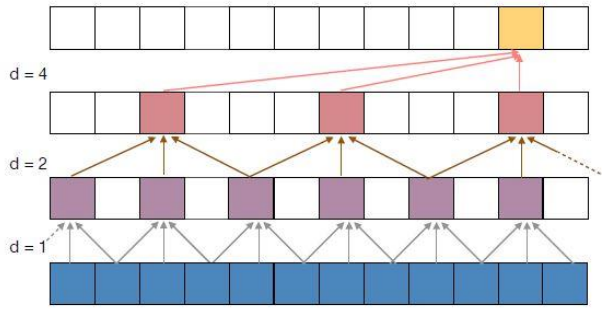

Fig. 4. Various dilation factors(d) are used in dilated operations with a kernel size of 3 to change the receptive field.

It has been demonstrated in recent work [XI] that TCN may be additionally improved by including residual connections. This has especially helped deeper networks by making gradient flow easier. This eliminates the requirement for the layer to learn a whole transform, allowing it to learn appropriate adjustments to the representation.
In this work, the proposed SS-TCN model consists of a single TCN layer with different dilated convolution blocks. The TCN layer utilizes a fixed convolution kernel size of 3, which means each convolution operation considers three consecutive time steps at a

time to capture local temporal patterns. The features extracted by the ResNet50 model are fed into the SS-TCN layer for temporal dependency learning. The number of filters used in the SS-TCN layer is 64, where each filter learns a different pattern from the input data.

To further enhance the capability of the SS-TCN network to learn both short- and long-term dependencies, four successive dilated convolutional blocks with dilation factors of 1, 2, 4, and 8 are used, respectively, while keeping the kernel size at 3. The theoretical receptive field of the TCN is calculated using RF =$1 + (k - 1) \sum_i d_i$ where k is the kernel size. For dilation factors [1,2, 4, 8] and k =3, the receptive field would be RF = 1 + (3-1) [1+2+4+8] = 31 frames. With a very small number of convolutional parameters, the network can collect temporal information over increasingly broad contexts due to the steadily growing dilation factors. All sequences are temporally normalized to 20 frames before being supplied to the network because the Florence 3D action sequences consist of 7–34 frames. As a result, the effective temporal coverage of each normalized sample covers the whole 20-frame input sequence, even though the theoretical receptive field is 31 frames. While longer-range temporal dependencies across the action sequence are captured by the higher dilation factors, local motion transitions are captured by the lower dilation factors. As a result, this method avoids the needless computer complexity involved in processing lengthier sequences while offering full temporal coverage of the normalized input.

After the temporal features are extracted by the TCN layer, the output is sent to the fully connected dense layer containing 128 neurons with the ReLU activation function. Finally, a dense output layer with num_classes neurons and a softmax activation function is created in the model. The output values are transformed into probability scores for each class by the softmax function, which makes sure that the sum of all probabilities equals to one. The predicted class is chosen to be the one with the greatest probability.

d) Action score prediction using Bi-LSTM with skeleton features

The use of Recurrent Neural Networks (RNNs) is very common in analyzing sequential patterns of both spatial and temporal data [XIII]. RNNs can handle sequential information, but they usually have difficulty handling long sequences because they will forget information in the previous time steps. This is referred to as the vanishing gradient problem. In order to address this problem, we employ a variant of RNN, which is referred to as the Long Short-Term Memory (LSTM) network. LSTM is one of the recurrent neural networks (RNN) that can be used to determine the long-term dependencies in a sequence of data. A different form of LSTM that is used in this work is called a Bidirectional Long Short-Term Memory (Bi-LSTM) network, and it makes use of both past and future contexts in the sequence. This approach is especially effective when recognition of video actions is involved. The LSTM cell that is used in our system can be described as shown by equations (15) to (20).

$$i_t = \sigma(b_i + U_i x_t + W_i h_{t-1}) \tag{15}$$

$$f_t = \sigma(b_f + U_i x_t + W_f h_{t-1}) \tag{16}$$

$$o_t = \sigma(b_o + U_o x_t + W_o h_{t-1}) \tag{17}$$

$$g_t = \sigma(b_g + U_g x_t + W_g h_{t-1}) \tag{18}$$

$$c_t = f_t c_{t-1} + g_t i_t \tag{19}$$

$$h_t = \tanh(c_t) o_t \tag{20}$$

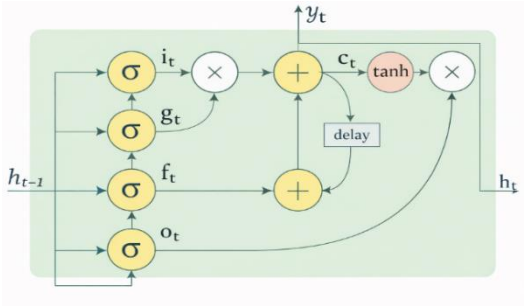

Fig. 5. Illustration of LSTM cell structure with its internal recurrence mechanism through input, forget, and output gates. [XXVII]

where s is the sigmoidal function, W_i W_f W_o are the weights of the input, forget, and output gates, respectively. In a similar manner, b_i , b_f, and b_o are the bias vectors of the input gate, forget gate, and output gate, respectively. The input gate i_t determines the flow of the new information to the cell, and the forget gate f_t determines the information to be forgotten by the cell. The o_t distribution controls the information, which is relayed to the output. The input modulation gate g_t is considered to be a major input to the cell. c_t is the internal state which controls internal recurrence of the cell, and $h_t$is the hidden state which stores the past information. Fig.5 indicates the structure of the LSTM cell.

Two Bi-LSTM layers have been used in our methodology with 64 nodes in each layer. The BiLSTM layer shape is displayed in the Fig.5 in which the bidirectional layer is a layer that contains forward and backward passes. A dense layer of 32 neurons then comes after the second Bidirectional Long Short Term Memory (BiLSTM) layer and is followed by a dropout layer with a value of 20%. A dropout rate of 40% has also been used between the two BiLSTM layers. The Softmax layer that comes after this dense layer is used for output classification. In both BiLSTM layers, the network training ReLU activation function is used. Fig. 6 illustrates the 2-layer BiLSTM network that will be used in modeling skeleton joint activity sequences on the basis of the spatio-temporal features.

Fig. 6. Shows the Bi-LSTM Architecture for spatial-temporal features learning from Skeletons Joints Sequences

Once the suggested BiLSTM network has been trained using the extracted spatio-temporal joint features (joint angle, joint velocity, and joint acceleration), the model would provide a probability distribution across all action classes of every input

sequence. In particular, the last Softmax layer generates a vector $P = [p_1, p_2, p_3 \ldots\ldots\ldots p_c]$, which is denoted by C to mean the number of classes of action and p_i, to indicate the estimated likelihood of the input sequence to be the i^{th} class. Such probabilities are normalized in the sense that $\sum_{i=1}^{C} p_i = 1$. The resultant class-wise probabilities are further used in feature fusion with the other modalities to get the overall recognition performance.

e) Late Score Fusion (Decision Level Score Fusion)

Late fusion, also known as decision-level fusion, is used in this work to combine the outputs of several streams of data, that is, RGB, Depth, and Skeleton streams. This method uses a deep learning model on each modality and gets a probability distribution over classes of actions through a Softmax layer. Let $P_m(c)$, represent the predicted modality probability of class c, where m $\in$ {rgb, depth, skeleton} and $c = 1, 2, \ldots\ldots\ldots, C$ modality, where Each modality is a normalized probability vector defined in equation (21) given below:

$$\sum_{c=1}^{C} P_m(c) = 1 \qquad (20)$$

The main strategy that is implemented in this work is the Weighted Sum Fusion (WSF) method because of its simplicity, efficiency, and high performance in recent studies of multimodal action recognition [XVII]. In this approach, the weight of each modality is determined by its relative significance, and the resulting fused probability is determined by a linear combination of the scores of each modality, as mentioned in equations (21) and (21).

$$P_{final}(C) = w_{rgb} P_{rgb}(c) + w_{depth} P_{depth}(c) + w_{skel} P_{skel}(c) \qquad (21)$$

$$w_{rgb} + w_{depth} + w_{skel} = 1 \qquad (22)$$

Where w_{rgb}, w_{depth} and w_{skel} are the represent the contribution of every modality. The final predicted class label is obtained using the maximum a posteriori (MAP) decision rule mentioned in equation (23) mentioned below:

$$\hat{y} = \arg max\, P_{final}(c) \qquad (23)$$

Equation (23) mathematically represents the final predicted class label. It picks the c class with the best fused probability score of all the classes. The model, in simple terms, allocates the input to the maximum probability class value. In this paper, we tested the fusion strategies (at early and intermediate, and decision levels) on the Florence 3d dataset. The accuracy of late fusion (WSM) was the highest, at 96.72% in relation to 93.68% of early fusion and 92.59% of intermediate fusion.

IV. Results and Discussion

The publicly available Florence 3D Actions [V] RGB-D data is used in this study to test the proposed approach. It is classified as RGB-D data because it was captured with the help of a Microsoft Kinect sensor. Modalities, such as RGB, depth, and skeleton, have their activities consistently aligned in the dataset. To train the model, an Intel 12th Gen Intel Core i7-1255U (2.40 GHz) processor and 16 GB RAM were utilized, which was powered by the Ubuntu 22.04 LTS operating system. Also, the use of the ResNet50 and Bi-LSTM networks was conducted in Google Colaboratory.

VI.i Florence 3d Actions Dataset

One of the most popular human action recognition benchmarks is the Florence 3D Actions Dataset [XXI], recorded at the University of Florence in 2012, with a Microsoft Kinect camera. It includes 215 samples of activities of 9 daily behaviors in all three formats such as RGB, depth and skeleton including '(answer) phone', 'bow', 'clap', 'drink out of a bottle', 'read watch', 'sit down', 'stand up', 'tie lace', and 'wave' repeated by ten subjects 2-3 times each as shown in the Fig 7. The data gives three synchronized RGB, depth, and 3D skeleton modalities where a single frame has the 3D position of 15 body joints that represent the important parts of the human body. A-frame is made up of video identity, actor identity, activity label, and 3D coordinates of 15 joints. A video segment is composed of a number of consecutive frames of an activity. The dataset has 215 video segments, and each video segment has between 7 and 34 frames, which makes it a total of 4016 frames. The three modalities are temporally synchronized at the input level. Fig. 8 depicts the frequency of 215 action video sequences. .001 learning rate and the SGD optimizer were used to train the network with categorical cross-entropy loss.

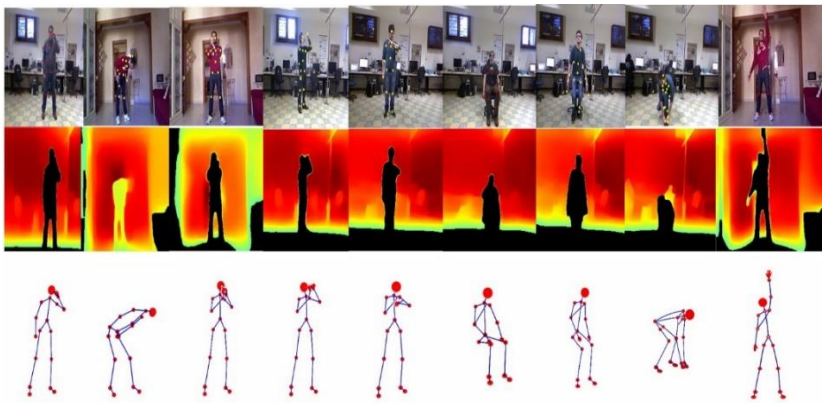

Fig. 7. Samples of all 9 actions of Florence 3D dataset in all three formats: RGB, depth, and skeleton

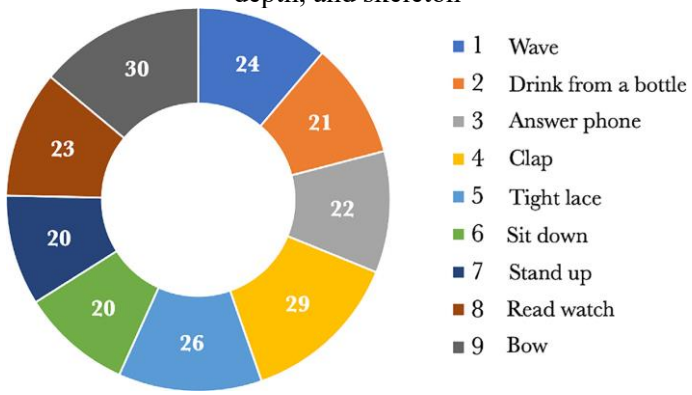

Fig. 8. shows the frequency of 215 action video sequences

VI.ii Experimental Results on Florence 3d action dataset

The overall aim of the suggested methodology is to identify human actions using 3D data, taking advantage of other complementary information provided by various modalities, i.e., RGB videos, depth maps, and 3D skeleton joints. To attain this, the proposed framework is divided into three key phases, which include: (i) spatial feature extraction, (ii) temporal sequence modeling and activity score prediction, and (iii) decision-level fusion. In the first phase, spatial features are obtained with a pretrained ResNet50 model applied to RGB and depth sequences. The video sequences are steadily sampled to produce fixed lengths of frame sequences. In order to explore the effect of temporal resolution, various sequence lengths of 18, 19, and 20 frames of RGB and depth modalities were tested. Out of these, a sequence length of 20 frames always gave the highest number of discriminative feature representations and the highest recognition performance. ResNet50 combines that the feature a 2048-dimensional feature vectors, which means that the sequence-level representations of both RGB and depth streams are of size (20×2048). Simultaneously, skeleton-based features are also obtained to represent the human activities and structure, and temporal dynamics. In particular, three forms of joint-level representation are performed, such as joint angle, joint velocity, and joint acceleration. A sequence length of 20 frames is used consistently with RGB-D data. The dimensions of these extracted 3D joint skeleton features, such as joint angles, are (20×14), joint velocity is (20×15), and joint acceleration is (20×15). These features are concatenated together to create a complete skeleton feature representation with size (20×44), which in effect encodes both spatial configuration and temporal motion features.

In the second step for RGB and depth feature sequences, a Temporal Convolutional Network (TCN) will be used to learn long-range temporal features with dilated causal convolutions, and for skeleton data, a BiLSTM network is used. The BiLSTM takes the input sequence of (20×44) skeleton features as input and discovers discriminative temporal representations of human motion. Lastly, at stage three, a late fusion strategy is employed using a weighted sum rule to fuse the scores of the prediction of the three modalities.

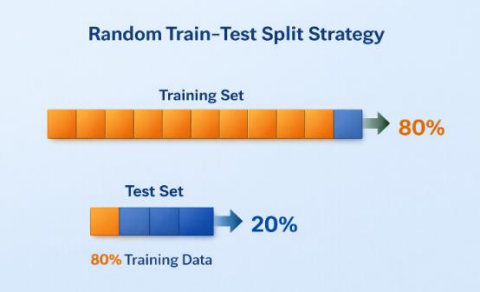

Fig. 9. Random split strategy for training and test set

In order to validate our method, we adopted a random train-test split validation with an 80% train set and 20% test set, as shown in Fig. 9. As an example, there are 172 sequences that are used to train and 43 sequences that are used to test.

Fig.10. Visualization of feature maps after the first TCN layer of 'wave' action for Florence 3D action dataset.

Fig.10 shows the visualization of our proposed approach's feature maps of RGB sequences of images of the action 'wave' of the Florence 3D action dataset after the first TCN layer. Fig. 11 demonstrates the ROC curve of the Florence 3D actions dataset. The curve drawn between true positive rate and false positive rate is the measure of performance of our model at different threshold levels. The curves near the top-left corner and high values of AUC (~0.94 0.97) denote that the model is very discriminative on all classes of actions. In general, the model is much better than the random classifier baseline, proving excellent classification abilities. Moreover, the similarity of the behavior of the curves in various classes indicates the stability of the model and its capacity to be applied to various human behaviors.

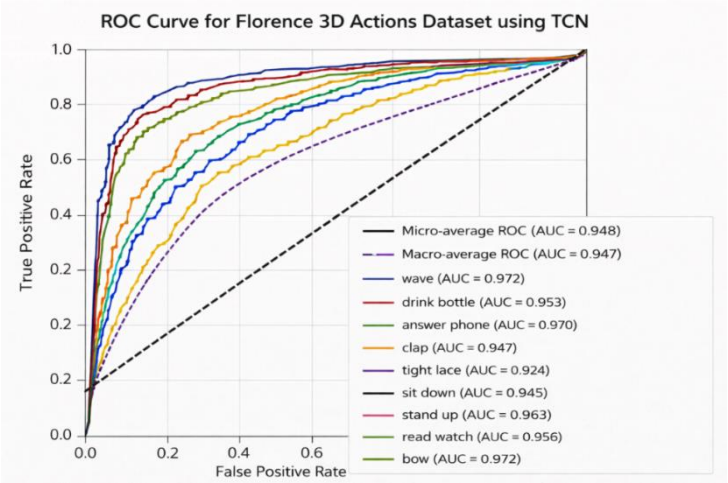

Fig. 11. ROC curve of Florence 3D action dataset

The confusion matrix of the Florence 3D actions dataset is presented in Fig. 12, where the dominance of the diagonal values, between 0.94 and 0.98, is strong evidence that the model is characterized with high recognition accuracy in all categories of actions.

This is an indication that the proposed framework has the capability of capturing the spatial and temporal characteristics of human actions. The additional fact that the high diagonal values are consistent across various classes also proves that the model is reliable even in various and complex activities like wave, clap, sit down, and read watch. The robustness of the model is guaranteed by the overall recognition accuracy of about 96.8%.

The elements that appear off-diagonal in every column of the confusion matrix indicate false positives, that is, samples of other classes are falsely predicted as a specific class. The false positive values are very low, with an average rate of about 0.01 to 0.02, meaning that the model is less prone to get one action confused with the other. There are, however, minor false positives given by similarities in motion patterns of some actions. As an example, 'wave' and 'answer phone', similar movements of the hand are involved, whereas 'sit down' and 'tight lace' may have similar poses of body bending. These are similarities that may result in mild misclassification.

The off-diagonal values in each row indicate false negatives, and the samples of a specific class are falsely categorized as the others. As well as in the case of false positives, the false negative values are also negligible and are usually in the range 0.01 to 0.03. This means that the model is very effective in the accurate recognition of the majority of cases of each action. The false negatives are very minimal and can be explained by the differences in the level of performance of various subjects performing the same action, view angle, or partial view, which may influence the extraction of the features and the process of temporal modeling.

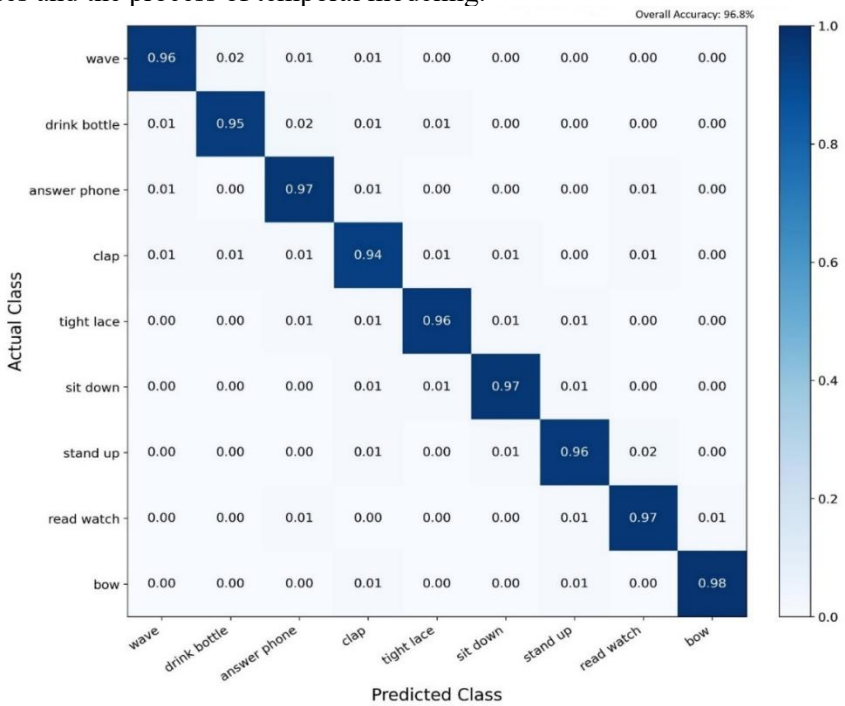

Fig. 12. Confusion matrix of Florence 3D action dataset of proposed approach

The confusion matrix shows little misclassification and good classification accuracy overall. The success of the presented multimodal strategy is confirmed by the low false

positive and false negative rates, which show high discriminative power and good generalization across action classes.

The training and validation accuracy curves show an increase with the epochs, which means that the model successfully learns discriminative spatio-temporal features using the Florence 3D Action dataset. The validation accuracy is similar to the training accuracy and indicates that the model has a high ability to generalize with low levels of overfitting, and the model attains a maximum accuracy of 96.8%.

Also, the training and accuracy curves of the Florence 3D Action dataset are illustrated in Fig. 13 and Fig. 14, respectively. As can be seen, the model training process takes almost 200 epochs, with the lowest validation loss of 0.083 being achieved at epoch 173, at which the model also achieves the highest accuracy of 96.8%, as shown in the accuracy curve.

Moreover, it is observed that both training and validation loss reduce steadily in the loss curve, which indicates that the model converges. The minor distance between training and validation loss shows that the model is well balanced between fitting and generalization without severe overfitting.

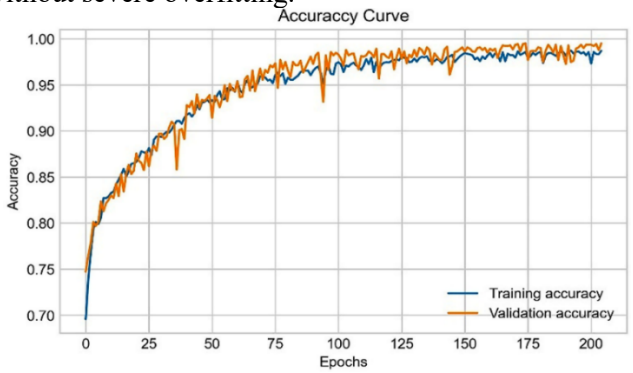

Fig. 13. Accuracy curve of the proposed model on the Florence 3D action dataset

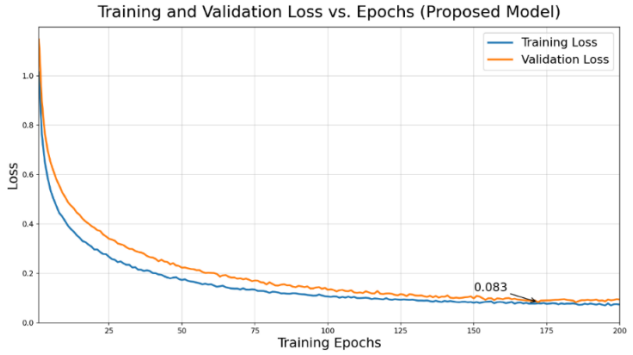

Fig. 14. Training curve of the proposed model on Florence 3D action dataset

Table 2: shows Layer-wise Architecture of the Proposed TCN Model.

S No.	Layer (type)	Output Shape	Parameters	Activation
1	input_layer_1 (InputLayer)	(None, 20, 2048)	0	ReLU
2	tcn_1 (TCN)	(None, 64)	610,880	ReLU

3	dense_2 (Dense)	(None, 128)	8,320	ReLU
4	dense_3 (Dense)	(None, 9)	645	Softmax
5	**Total parameters:**		619,845 (2.36 MB)	
6	**Trainable parameters:**		619,845 (2.36 MB)	
7	**Non-trainable parameters:**		0 (0.00 B)	

The architecture of the suggested deep learning model for action recognition is shown in Table 2. Temporal feature vectors taken from each video are represented by a feature shape (20, 2048), which are accepted by the input layer. After processing these sequences to find temporal relationships, the TCN layer creates a 64-unit compact representation. Fully connected (Dense) layers come next, which perform classification into 9 action classes and further enhance the features. The network is fully optimized during training with no fixed (non-trainable) components because all 6.2M of the model's parameters are trainable. Our suggested model on TCN has been proven to be more efficient and effective than most of the current temporal models like LSTM, GRU, CNN, CNN-LSTM hybrids, and more traditional and latest variants of TCN. The limitation of traditional LSTM and GRU models is the sequential computation and greater complexity, whereas CNN-based models are not effective in terms of temporal models. The proposed model has similar performance using a simpler and lightweight architecture compared to previous TCN-based architectures given in [XV] and more recent models that add attention modules and hybrid modules to extract features better, such as TCN-Attention-HAR [XXXV] and TCN-Inception [I]. It achieves high accuracy 96.8 percent and converges more quickly, and is less computationally demanding, making it more applicable in real-time human action recognition applications.

In this study, RGB, Depth, and Skeleton modalities are intrinsically temporally synchronized, meaning that the same state of the preformed action is represented by equivalent temporal positions. The multimodal observations for a series of T temporal samples can therefore be expressed as $\{RGB_T, RGB_T, Skeleton_T\}_{t=1}^{T}$. Without the need for an extra frame-level alignment mechanism, this intrinsic synchronization offers temporal equivalence across the three modalities.

The suggested framework autonomously learns modality-specific temporal characteristics instead of imposing a common feature representation early on. TCNs are used to model the RGB and Depth representations, and BiLSTM is used to model the skeleton representation. The heterogeneous nature of the three modalities, which possess fundamentally different feature characteristics such as RGB mainly offering appearance information, depth recording three-dimensional structure and displacement cues, and skeleton depicting body-joint configuration and motion dynamics, motivates this modality-specific design. Late fusion is then used to integrate the separately learnt temporal representations, enabling complementary data from the synchronized modalities to contribute to the final action classification.

Thus, synchronized temporal indexing is used in the suggested framework to establish multimodal temporal correspondence, and late fusion is used to merge the generated temporal representations. By avoiding needless restrictions, this approach preserves the complementary information of various modality-specific feature spaces.

VI.iii Ablation study

To determine the role of each component in the proposed multimodal action recognition framework, we performed an extensive ablation study on their contribution to the Florence 3D Action dataset, as shown in Table 3. The impacts of different modalities (RGB, Depth, Skeleton), fusion directions (early, middle, and decision-level fusion), and temporal modeling directions (TCN and LSTM) are carefully examined. Throughout this research, one gets a vivid idea of how every one of the components adds to the overall performance and to the design of the final model.

To begin with, the individual modalities' performance was examined. Individual use of the RGB features (ResNet50 + TCN) showed high performance since the model is capable of learning the spatial representation. Depth features only gave strength to background differences and lighting differences but reduced discriminative capacity slightly. Equally, Skeleton-based features were also able to capture motion dynamics effectively but had no rich appearance information. These observations affirm that the various modalities give complementary information regarding action recognition.

Moreover, the influence of temporal modeling was also considered according to the comparison of the TCN and LSTM-based models. The findings have also shown that TCN is more accurate and faster converging compared to LSTM since it is capable of capturing long-range temporal interactions with dilated convolutions and parallel processing. This shows that TCN is effective in non-linear time-series modeling of human behavior.

In addition, alternative strategies of fusing were investigated, such as early fusion, intermediate fusion, and decision-level (late) fusion based on using weighted sum fusion. The decision-level fusion approach had the highest performance among them, since it enables independent learning by a modality, followed by the combination of these predictions, resulting in enhanced discriminative ability. The early and intermediate fusion approaches displayed relatively poor performance because of feature-level interference and poor modality-specific learning.

Lastly, the designated proposed model comprising RGB, Depth, and Skeleton modalities in combination with TCN-based temporal modeling and weighted sum fusion exhibits the highest accuracy of 96.8% and the lowest validation loss, which represents consistent convergence and good generalization capabilities. These ablation outcomes clearly point out that every element of the system- multimodal input, efficient temporal modelling, and optimal fusion strategy is essential in the overall performance of the system.

Table 3: A summary of these findings is mentioned in the table below, and it gives a clear comparative view of the effects of different architectural choices on recognition performance.

Model Configuration	Modalities	Temporal Model	Fusion Method	Accuracy (%)	Remarks	Source
ResNet50 (no temporal modeling)	RGB	None	None	85.21	Only spatial features	[XXVIII]
ResNet50 + TCN	RGB	TCN	None	91.34	Temporal modeling improves performance	[II]

ResNet50 + TCN	Depth	TCN	None	92.10	Depth robust to variations	[XXXIII]
Skeleton Features + Classifier	Skeleton	None	None	88.76	Captures motion only	[IV]
ResNet50 + LSTM	RGB + Depth	LSTM	Early Fusion	93.85	Sequential modeling	Proposed
ResNet50 + TCN	RGB + Depth	TCN	Intermediate Fusion	95.12	Better than early fusion	Proposed
Proposed Model (ResNet50 + TCN + Fusion)	RGB + Depth + Skeleton	TCN	Decision-Level (Weighted Sum)	**96.8**	Final model (Best Performance)	**Proposed (Final)**

VI.iv Comparison with Existing Literature

The suggested approach is compared with a number of state-of-the-art methods as shown in Table 4 that were tested on the Florence 3D Action dataset. Previous approaches like Vemulapalli et al. [XXVIII], Anirudh et al. [II], and Wang et al. [XXXIII] are based on manually designed skeleton-based features, and they deliver low accuracy, which implies low skills in modeling complex spatial and temporal patterns. Other methods such as Wei et al. [XXXIV] use LSTM-based time modeling, and they have a better performance (91.3%), although they are limited by the loss of high capabilities to model long-range dependencies.

Table 4: Performance Comparison of State-of-the-Art Methods on the Florence 3D Action Dataset

Methods	Technique	Accuracy (%)
Vemulapalli et al. (2014) [XXVIII]	Lie group representation (skeleton-based)	90.9
Anirudh et al. (2015) [II]	Elastic functional coding (skeleton features)	89.7
Wang et al. (2012) [XXXIII]	Actionlet ensemble (joint-based features)	91.6
Wei et al. (2018) [XXXIV]	WTP + R-LSTM (spatio-temporal fusion)	91.3
Chen et al. (2021) [VIII]	Skeleton-based CNN with data augmentation	96.59
Dhiman et al. (2021) [IX]	Spatio-temporal CNN with attention	~96.0
Proposed (RGB + Depth + Skeleton) – Late Fusion	ResNet50 + TCN + BiLSTM	**96.8**

Recent methods based on deep learning [VII I] and [IX] use CNN-based frameworks and data augmentation techniques, with higher accuracies of 96.59% and approximately 96, respectively. Nevertheless, they work with input skeleton or depth (single-modality) and have a higher computational cost.

The approach suggested, in contrast, combines the use of multi-modal information (RGB, Depth, and Skeleton) and ResNet50 to extract spatial features, TCN to model the time dynamics effectively, and BiLSTM to model skeleton dynamics. The combination of complementary features between all modalities using decision-level (late) fusion would be the most accurate, with the highest accuracy of 96.8. This illustrates that the suggested method not only outperforms the existing methods but also offers a more robust and scalable framework by making effective use of both spatial and temporal information about the different modalities.

V. Conclusions and Future Work

A powerful multimodal human action recognition system is proposed in this paper and tested on the Florence 3D Action dataset. The model comprises the integration of RGB, Depth, and Skeleton modalities, where spatial features are obtained through ResNet50, temporal dependencies are extracted through TCN, and skeleton dynamics are trained with BiLSTM with the help of joint angle, velocity, and acceleration features. Decision-level (late) fusion is successfully used to integrate heterogeneous information between all modalities so that it improves the overall discriminative ability of the system. As the experimental findings reveal, the presented approach is also able to reach a high recognition rate of 96.8, which seems to be higher than some of the existing state-of-the-art solutions, and the architecture is lightweight and efficient. The curves of training and validation also show that the convergence is stable with low overfitting, which proves a good generalization capacity. Comprehensively, the suggested framework offers a good tradeoff between accuracy, computational power, and scalability and hence is applicable to real-world human action recognition systems.

Despite the positive outcomes of the suggested approach, there are multiple ways to go and improve it. To confirm the validity of the large and complex scale, the model could be applied to bigger and more complex data like the NTU RGB+D Dataset at the first stage. Second, the input of each modality can be dynamically prioritized by introducing higher-level fusion approaches, such as attention-based or adaptive fusion processes. Third, long-range dependency learning can be further improved by integrating transformer-based architectures for learning temporal models. Moreover, the suggested framework can be streamlined to a real-time implementation on edge devices at the cost of minimizing the computational complexity and the model size. Last but not least, a study of self-supervised or unsupervised learning algorithms might help mitigate the reliance on labeled data and enhance extrapolation in a variety of settings.

Conflict of Interest:

There was no relevant conflict of interest regarding this paper.

References

I. Al-qaness, Mohammed AA, et al. "TCN-inception: temporal convolutional network and inception modules for sensor-based human activity recognition." *Future Generation Computer Systems* 160 (2024): 375-388. 10.1016/j.future.2024.06.016.

II. Anirudh, Rushil, et al. "Elastic functional coding of human actions: From vector-fields to latent variables." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015. 10.1109/CVPR.2015.7298934.

III. Bai, Ruwen, et al. "Hierarchical graph convolutional skeleton transformer for action recognition." *2022 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2022. 10.1109/ICME52920.2022.9859781.

IV. Bai, Shaojie, J. Zico Kolter, and Vladlen Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling." *arXiv preprint arXiv:1803.01271* (2018). 10.48550/arXiv.1803.01271.

V. Batool, Mouazma, et al. "Multimodal human action recognition framework using an improved CNNGRU classifier." *IEEE Access* 12 (2024): 158388-158406. 10.1109/ACCESS.2024.3481631.

VI. Bijrothiya, Sadhna, and Vaibhav Soni. "An architecture for human activity recognition using TCN-BI-LSTM HAR based on wearable sensor." *Procedia Computer Science* 260 (2025): 805-813. 10.1016/j.procs.2025.03.261.

VII. Biswas, Sougatamoy, Anup Nandy, and Asim Kumar Naskar. "Lightweight multimodal feature fusion and spatiotemporal learning for human action recognition on edge devices." *IEEE Transactions on Emerging Topics in Computational Intelligence* (2025). 10.1109/TETCI.2025.3616054.

VIII. Chen, Junjie, et al. "A data augmentation method for skeleton-based action recognition with relative features." *Applied Sciences* 11.23 (2021): 11481. 10.3390/app112311481.

IX. Dhiman, Chhavi, Dinesh Kumar Vishwakarma, and Paras Agarwal. "Part-wise spatio-temporal attention driven CNN-based 3D human action recognition." *ACM Transactions on Multimidia Computing Communications and Applications* 17.3 (2021): 1-24. 10.1145/3441628.

X. Farha, Yazan Abu, and Jurgen Gall. "Ms-tcn: Multi-stage temporal convolutional network for action segmentation." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019. 10.1109/CVPR.2019.00369.

XI. Funahashi, Ken-ichi, and Yuichi Nakamura. "Approximation of dynamical systems by continuous time recurrent neural networks." *Neural networks* 6.6 (1993): 801-806. 10.1016/S0893-6080(05)80125-X.

XII. Hidayanto, Nur Awal, Adhi Prahara, and Riky Dwi Puriyanto. "Hierarchical long short-term memory for action recognition based on 3D skeleton joints from Kinect sensor." *Jurnal Informatika Ahmad Dahlan* 15.1 (2021): 17-27. 10.26555/jifo.v15i1.a20106.

XIII. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780. 10.1162/neco.1997.9.8.1735.

XIV. Hou, Jingxuan, Tae Soo Kim, and Austin Reiter. "Train, diagnose and fix: Interpretable approach for fine-grained action recognition." *arXiv preprint arXiv:1711.08502* (2017). 10.48550/arXiv.1711.08502.

XV. Lea, Colin, et al. "Temporal convolutional networks for action segmentation and detection." *2017 IEEE conference on computer vision and pattern recognition (CVPR)*. IEEE, 2017. 10.48550/arXiv.1611.05267.

XVI. Liu, Chengming, et al. "Angle information assisting skeleton-based actions recognition." *PeerJ Computer Science* 10 (2024): e2523. 10.7717/peerj-cs.2523.

XVII. Murad, Abdulmajid, and Jae-Young Pyun. "Deep recurrent neural networks for human activity recognition." *Sensors* 17.11 (2017): 2556, 10.3390/s17112556.

XVIII. Musallam, Mohamed Adel, et al. "Temporal 3d human pose estimation for action recognition from arbitrary viewpoints." *2019 international conference on computational science and computational intelligence (csci)*. IEEE, 2019. 10.1109/CSCI49370.2019.00052.

XIX. Oord, Aaron van den, et al. "Wavenet: A generative model for raw audio." *arXiv preprint arXiv:1609.03499* (2016). 10.48550/arXiv.1609.03499.

XX. Prasad Raghava, "Fusion of RGB and Skeletal Data Using Gated Features for Human Action Recognition". International Journal of Food and Nutritional Sciences 11.6(2022): 1-6. 10.1007/s12652-019-01239-9.

XXI. Rahayu, Endang Sri, et al. "Human activity classification using deep learning based on 3D motion feature." *Machine Learning with Applications* 12 (2023): 100461. .1016/j.mlwa.2023.100461.

XXII. Rana, Vijay Singh, Ankush Joshi, and Kamal Kant Verma. "A Unified Deep Learning Approach for Human Activity Recognition with RGB-D Data." 2025 International Conference on Innovations and Emerging Technologies In AI & Communication Systems (IETACS). IEEE, 2025. 10.1109/IETACS68750.2025.11385548.

XXIII. Sanchez-Caballero, Adrian, David Fuentes-Jimenez, and Cristina Losada-Gutiérrez. "Real-time human action recognition using raw depth video-based recurrent neural networks." *Multimedia Tools and Applications* 82.11 (2023): 16213-16235. 10.1109/ACCESS.2024.3481631.

XXIV. Seidenari, Lorenzo, et al. "Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses." *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2013. 10.1109/CVPRW.2013.77.

XXV. Shaikh, Muhammad Bilal, et al. "From CNNs to transformers in multimodal human action recognition: A survey." *ACM Transactions on Multimedia Computing, Communications and Applications* 20.8 (2024): 1-24. 10.1145/3664815.

XXVI. Shotton, Jamie, et al. "Real-time human pose recognition in parts from single depth images." *CVPR 2011*. Ieee, 2011. 10.1109/CVPR.2011.5995316.

XXVII. Siami-Namini, Sima, Neda Tavakoli, and Akbar Siami Namin. "The performance of LSTM and BiLSTM in forecasting time series." *2019 IEEE International conference on big data (Big Data)*. IEEE, 2019. 10.1109/BigData47090.2019.9005997.

XXVIII. Vemulapalli, Raviteja, Felipe Arrate, and Rama Chellappa. "Human action recognition by representing 3d skeletons as points in a lie group." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014. 10.1109/ICALIP.2016.7846646.

XXIX. Verma, Kamal Kant, and Brij Mohan Singh. "Deep Multi-Model Fusion for Human Activity Recognition Using Evolutionary Algorithms." *International Journal of Interactive Multimedia and Artificial Intelligence* 7.2 (2021): 44-58. 10.9781/ijimai.2021.08.008.

XXX. Verma, Kamal Kant, Brij Mohan Singh, and Amit Dixit. "A review of supervised and unsupervised machine learning techniques for suspicious behavior recognition in intelligent surveillance system." *International Journal of Information Technology* 14.1 (2022): 397-410. 10.1007/s41870-019-00364-0.

XXXI. Verma, Kamal Kant, et al. "Two-stage human activity recognition using 2D-ConvNet." *IJIMAI* 6.2 (2020): 125-135. 10.9781/ijimai.2020.04.002.

XXXII. Vijay Singh Rana, Ankush Joshi, Kamal Kant Verma, "Lightweight 3DCNN-BiLSTM Model for Human Activity Recognition using Fusion of RGBD Video Sequences", International Journal of Information Technology and Computer Science(IJITCS), Vol.17, No.6, pp.176-193. 2025. 10.5815/ijitcs.2025.06.10

XXXIII. Wang, Jiang, et al. "Mining actionlet ensemble for action recognition with depth cameras." *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012. 10.1109/CVPR.2012.6247813.

XXXIV. Wei, Lianglei, et al. "A novel 3d human action recognition framework for video content analysis." *International Conference on Multimedia Modeling*. Cham: Springer International Publishing, 2018. 10.1007/978-3-319-73603-7_4.

XXXV. Wei, Xiong, and Zifan Wang. "TCN-attention-HAR: Human activity recognition based on attention mechanism time convolutional network." *Scientific Reports* 14.1 (2024): 7414. 10.1038/s41598-024-57912-3.

XXXVI. Xie, Dongwei, et al. "MAF-Net: A multimodal data fusion approach for human action recognition." *PloS one* 20.4 (2025): e0319656. 10.1371/journal.pone.0319656.

XXXVII. Zhan, Tianyu. "Research on Action Recognition Algorithm Based on Multimodal Data Fusion." *2025 IEEE 5th International Conference on Electronic Technology, Communication and Information (ICETCI)*. IEEE, 2025. 10.1109/ICETCI64844.2025.11084160.

XXXVIII. Zhang, Yumin, and Yanyong Wang. "A comprehensive survey on RGB-D-based human action recognition: algorithms, datasets, and popular applications." *EURASIP Journal on Image and Video Processing* 2025.1 (2025): 15. 10.1186/s13640-025-00677-0.