Check for updates

Wiley
International Journal of Intelligent Systems
Volume 2024, Article ID 2323337, 19 pages
https://doi.org/10.1155/2024/2323337

WILEY

Research Article

ESTS-GCN: An Ensemble Spatial–Temporal Skeleton-Based Graph Convolutional Networks for Violence Detection

Nourah Fahad Janbi,[1] **Musrea Abdo Ghaseb**,[2] **and Abdulwahab Ali Almazroi**[1]

[1]*Department of Information Technology, College of Computing and Information Technology at Khulais, University of Jeddah, Jeddah, Saudi Arabia*
[2]*King Abdulaziz University, Faculty of Computing and Information Technology, Jeddah, Saudi Arabia*

Correspondence should be addressed to Nourah Fahad Janbi; nfjanbi@uj.edu.sa

Received 7 March 2024; Revised 15 September 2024; Accepted 17 September 2024

Academic Editor: Gennaro Vessio

Copyright © 2024 Nourah Fahad Janbi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Surveillance systems are essential for social and personal security. However, monitoring multiple video feeds with multiple targets is challenging for human operators. Therefore, automatic and smart surveillance systems have been introduced to support or replace traditional surveillance systems and build safer communities. Advancements in artificial intelligence techniques, particularly in the field of computer vision, have boosted this area of research. Most existing works have focused on image-based (RGB-based) machine learning and deep learning algorithms for detecting anomalous and violent events. In this study, we propose a unique Ensemble Spatial–Temporal Skeleton-Based Graph Convolutional Networks (ESTS-GCNs) model for violence detection that automatically uses spatial and temporal data to detect violence in surveillance videos. Skeleton-based algorithms are less sensitive to pixel-based noise and background interference, making them excellent candidates for activity and anomaly detection. Our proposed ensemble-based architecture utilizes Graph Convolutional Networks (GCNs) and comprises multiple spatial and temporal modules. Three different spatial pipelines are exploited: channel-wise topologies, self-attention mechanism, and graph attention networks. The models were trained and evaluated using two skeleton-based datasets introduced by us: Skeleton-based Real-Life Violence Situations (RLVS) and NTU-Violence (NTU-V). Our model achieved a maximum accuracy of around 93% and outperformed existing models by more than 10%.

Keywords: graph attention networks; graph convolutional networks; safe community; self-attention; skeleton; smart city; smart surveillance; violence detection

1. Introduction

Smart surveillance systems depend on computer vision technologies to identify anomalous events in surveillance video streams [1]. Although surveillance cameras can be found in almost all sensitive locations, the resulting video stream is challenging to monitor and analyze and costly to transmit and store [2–4]. One of the most important events to be identified in surveillance video streams is violence, such as fighting, shooting, explosions, abuse, or riots. Detecting violent events would allow security agencies, police, or even national defense in the case of national-level situations to react quickly and maintain community safety.

Early studies on violence detection systems relied on hand-crafted features such as histograms, sparse codification, trajectories, optical flow, and interest points to identify violence [5]. Subsequently, research on violence detection applications has been led by advancements in artificial intelligence (AI) technologies, specifically in the field of computer vision. Most existing works use color features (e.g., 3D color features red, green, and blue [RGB]) of images (frames) from video streams as input to identify violent scenes [6–11]. These image-based (or RGB-based) violence detection algorithms suffer from pixel-based noise, lighting, viewpoint, and background interference because they are sensitive to environmental changes [6].

An alternative approach involves skeleton-based methods in which human biomechanical characteristics and parameters are collected to understand the geometry and motion of the human body. Skeleton-based methods utilize one of the pose estimation models (e.g., OpenPose [12]) to identify humans in images (frames) and extract their key point (joint) coordinates (x and y) and scores (see Figure 1). Afterward, violent scenes are identified by comparing the joint relationship within a frame and across multiple frames in a video stream. Skeleton-based algorithms have proven to be robust in motion and event detection problems [13]. However, skeleton-based algorithms for violence detection [14–16] for smart surveillance are still in their early stages and require research attention.

Existing studies primarily depend on deep learning technologies, including convolutional neural networks (CNN), deep neural networks (DNN), and recurrent neural networks (RNN). After creating a pseudoimage of the skeleton coordinates, they may be utilized to process skeleton data directly as joint coordinates. However, they fail to capture the internal relationship between joints and complex multidynamic and multiperson violent scenes [17]. In addition, methods that depend on predefined rules or patterns are difficult to generalize to other actions that were not trained for [18]. Therefore, the joint relationships of both spatial (intraframe) and temporal (interframe) spaces are essential for understanding human actions, and patterns should be automatically captured.

Graph Convolutional Networks (GCNs) have emerged as generalized CNNs for addressing arbitrary and unstructured data. They have proven their effectiveness for graph-structured data as they can capture both the features of a node and its locality to make predictions. They are also one of the most widely adopted approaches for action recognition problems that use skeleton data [19]. However, to the best of our knowledge, GCNs have not been explicitly studied for violence detection.

Motivated by the limitations of existing image-based (or RGB-based) violence detection algorithms, the success of skeleton-based methods in motion and event detection problems, the emergence of new AI technologies (e.g., GCN), and to overcome the gap in the literature of skeleton-based algorithms for violence detection, this study explores various GCN mechanisms as violence detection approaches. We designed and investigated a novel Ensemble Spatial–Temporal Skeleton-Based Graph Convolutional Networks (ESTS-GCNs) model for violence detection that automatically uses spatial and temporal data to detect violence in surveillance videos. ESTS-GCN adopts ensemble-based architecture and utilizes three cutting-edge approaches for feature transformation and aggregation: channel-wise topologies, self-attention mechanism, and graph attention networks (GATs). Channel-wise topologies eliminate the need to aggregate features from all channels and increase the feature extraction flexibility. Self-attention mechanisms do not require prior knowledge of the relationships between items to learn global dependencies and recognize important features, which make it a flexible approach for discovering useful patterns. Self-attention takes into account the connection between every joint, irrespective of the inherent structure of human joints, which may potentially limit the model's ability to be generalized and mask the true graph design of the skeletal data. Therefore, graph attention networks (GAT) are introduced to enforce the graph structure of the human skeleton and its naturally linked joints into the attention mechanism.

In addition, we did not find any public skeleton-based datasets for violent events, at the time this article was written, which could be a reason for the limited work in this area. Therefore, in this study, we created two skeleton-based datasets for violent events, namely, Skeleton-based Real-Life Violence Situations (RLVS) and NTU-Violence (NTU-V), and provided them to the public to boost the research in this field.

A summary of this paper's main contributions is provided as follows:

- Designed and investigated a novel ESTS-GCNs model for violence detection, including three different spatial pipelines: channel-wise topologies, self-attention, and GATs.
- The first study in which decoupled spatial–temporal skeleton-based GCN was specifically proposed for violence detection. We designed independent spatial and temporal modules that utilize cutting-edge techniques aiming to reduce computational complexity and improve model performance.
- Created two skeleton-based violence event datasets and provided them to the public to boost the research in this area.

The remainder of this paper is structured as follows. Studies on violence detection and GCNs are included in Section 2. A detailed explanation of the proposed spatial–temporal skeleton-based GCN models is provided in Section 3. The assessment procedure of our simulations, including dataset collection and preliminary processing, setting up experiments, and development and evaluation outcomes, is covered in Section 4. Finally, in Section 5, conclusions and future work are presented.

2. Related Works

The work on GCNs in general and violence detection in particular is covered in this section. Though GCN is a renowned technique for general action recognition problems, it has not been explicitly addressed for violence identification. Here, we review studies related to GCNs, as we propose them for violence detection. We also cover studies related to transformers, including self-attention and GATs, as they are related to our work.

2.1. Violence Detection. Violence detection for surveillance applications using video data was introduced by Datta, Shah, and Da Vitoria Lobo [20], in which an Acceleration Measure Vector (AMV) algorithm was adopted. Since then and until 2013, the focus has been on domain knowledge and handcrafted features such as histograms, sparse codification,

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

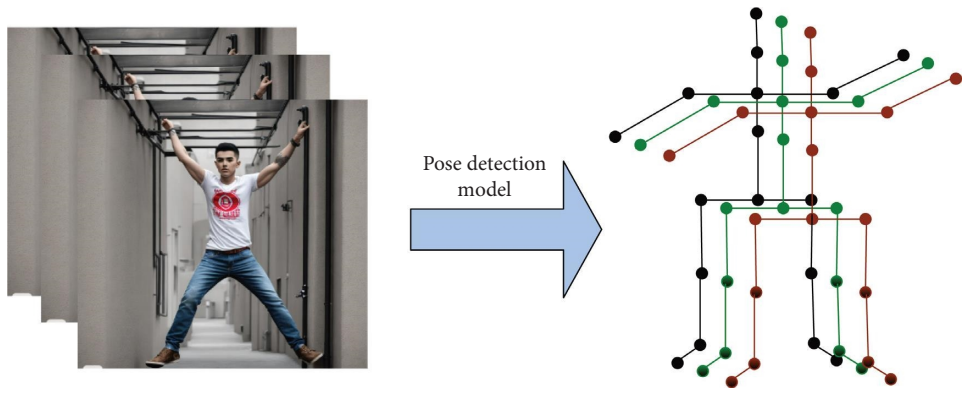

Figure 1: Process of extracting skeleton data from each frame in videos using OpenPose detector. Each frame is represented by a skeleton with 25 joints.

trajectories, optical flow, and interest points [5]. Subsequently, research has been led by advancements in deep learning and machine learning approaches. Three primary classes may be distinguished from the existing studies: RGB-based image-based systems [6–11], skeleton-based [14, 15], and hybrid (image and skeleton) [16] algorithms. Table 1 summarizes the existing literature.

Multiple approaches have used the color features of images (frames) as inputs for classifying violent videos. Choqueluque-Roman and Camara-Chavez [8] adopted a weakly supervised approach for action-detection and action-segmentation stages that depend only on video-level labels. They leveraged a pretrained human detector and dynamic images to discard background noise. The spatio-temporal attributes were captured using a 3D-CNN, while the features related to space were captured using a 2D-CNN. Similar to this, Ali [7] used background removal using a convolutional autoencoder and an object detection stream utilizing YOLOv5 to identify anomalous frames. Huszar et al. [6] examined the fine-tuned X3D-M and transfer-learned X3D-M models and two 3D convolution models for violence detection. Long short-term memory (LSTM) and deep learning techniques have been merged in several previous research studies. Sharma and Baghel [10] developed a technique for deep learning that consists of a ConvLSTM block and a pretrained feature extraction model. Similar to this, Soliman et al. [21] suggested using a model trained with LSTM for acquiring time-based characteristics and a pre-trained neural network strategy to collect spatial attributes. Rendon-Segador et al. [11] integrated adversarial neural structured learning with vision transformer to develop a violence detection model.

The literature has also taken into consideration transformers and both time and space attention. For instance, Huillcen Baca et al. [9] presented a deep learning model consisting of attention modules for both spatial and temporal spaces that were fed into a 2D CNN. While the temporal perception module uses the median across all RGB channels to capture temporal characteristics in a single channel, the spatial attention module uses the variation between two frames that follow to collect spatial information. Abdali and Aggar [24] proposed a data-efficient video transformer (DEVTr) model that can be adapted to small datasets. They also introduced two methods for data augmentations: random frame erasing and frame position shifting. Alternatively, Kang, Park, and Park [23] proposed a preprocessing pipeline with consecutive channel-averaged frame grouping that feeds into a 2D CNN. To improve the model's efficiency, lighter temporal (temporal squeeze and excitation) and spatial (map of motion salience) focus components are used. The temporal attention module naturally draws attention to the correlated time periods with a target event, while the spatial focus component emphasizes the prominent areas of the feature maps. Rendón-Segador et al. [22] utilized a three-dimensional version of DenseNet, self-attention with multiple heads, and bidirectional (forward and backward) convolutional LSTM. They also examined the effect of different input formats on the results, specifically, optical flow and adjacent frames' subtraction.

Existing image-based violence detection algorithms suffer from pixel-based noise, lighting, viewpoint, and background interference because they are sensitive to environmental changes. Therefore, as we saw in previous works, they focus on background and noise elimination by extracting important features and detecting changes in frames.

An alternative to image-based approaches is the skeleton-based approach. However, only a few works found in the literature adopt skeleton-based approaches to detect violence in video data. Human pose information is usually extracted using one of the well-known pose estimation models such as OpenPose [12], PoseNet [25], and MoveNet [26]. Extracted human skeletons, that is, the coordinates of the identified joints, can be represented in different ways: either by displaying the identified skeletons as an image, such as skeleton images or certainty heat maps, or as a collection of connected points (such as a graph). Omarov et al. [15] used PoseNet [25] for pose estimation key points' extraction and designed a neural network that accepts skeleton key points as input to identify violence and non-violence scenes. Su et al. [14] represented the coordinates of joints as a cluster of 3D point cloud data and studied the

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 1: Summary of violence detection related works.

Reference	Year	Image (RGB)	Skeleton	DNN	Self-attention	Graph attention	GCN	Channel-wise
Soliman et al. [21]	2019	✓		✓				
Sharma and Baghel [10]	2020	✓		✓				
Su et al. [14]	2020		✓	✓	✓			
Rendón-Segador et al. [22]	2021	✓		✓	✓			
Kang, Park, and Park [23]	2021	✓		✓	✓			
Choqueluque-Romanet and Camara-Chavez [8]	2022	✓		✓	✓			
Abdali and Aggar [24]	2022	✓		✓	✓			
Omarov et al. [15]	2022		✓	✓				
Garcia-Cobo and SanMiguel [16]	2023	✓	✓	✓				
Huillcen Baca et al. [9]	2023	✓		✓	✓			
Ali [7]	2023	✓		✓				
Huszar et al. [6]	2023	✓		✓				
Our proposal	2023		✓	✓	✓	✓	✓	✓

Abbreviations: DNN, deep neural networks and GCN, graph convolutional networks.

interrelationships between them using their proposed Skeleton Points Interaction Learning module. It aims to capture the relation of both feature and spatial–temporal position using weight distribution. In addition, they adopted a multiple heads mechanism to capture features from different points.

In another direction, both the image and skeletons (hybrid) were used for violence detection. Garcia-Cobo and SanMiguel [16] designed an architecture consisting of two streams: one for skeleton extraction and the other for dynamic change detection among the frames. The extracted skeletons were rendered over a black background with colored human parts. The two streams were combined using a convolutional LSTM and a final classifier.

2.2. GCN. GCN is an alternative approach to CNN that has emerged to address non-Euclidean structured data using convolutional network technologies. Due to the Euclidean structure of most images and video data, CNN-based techniques work well with them. Nevertheless, the CNN-based approaches face major difficulties when the data are graph-based and exhibit a non-Euclidean structure with intricate linkages and interdependency [27].

The human skeleton's joints may be seen as a graph as they naturally connect and arranged. Yan, Xiong, and Lin [18] are one of the earlier authors who proposed the use of GCN for action recognition using skeletal data. Their proposed model, the spatial–temporal GCN (ST-GCN), builds a temporal graph from a series of skeletons. The graph's spatial component was built using the human skeleton's real structure, while its temporal component was created by connecting the same joints across skeletons that came afterwards. However, ST-GCN and similar GCN methods based on body structures have multiple drawbacks. First, as they focus only on the neutral human body structure and directly connected joints, they fail to capture the relationship between distant joints (not physically connected) which can provide key information for identifying human actions. In addition, static and predefined graphs would affect the ability of a model to extract diverse features.

Therefore, recent works have aimed to solve these issues by introducing adaptive learning [28–31], channel-wise topologies [17, 32], and attention mechanisms [28, 29, 33, 34]. To capture richer dependencies, an Actional Structure Graph Convolution Networks (AS-GCN) model was developed by Li et al. [28] by utilizing an encoder-decoder model in an A-link inference module to capture the undetected relationships of certain actions. A two-stream adaptive graphing convective system (2s-AGCN) with a uniformly or individually learned topology was published by Shi et al. [29]. Subsequently, it utilized a reverse propagation strategy. The 2s-AGCN improves topology learning by dynamically modeling joint dependencies based on their respective features. Xie et al. [30] developed a dynamic semantic-based GCN (DS-GCN) where two graphs are dynamically generated: node-aware and edge-aware. On the other hand, Ye et al. [31] created the context-encoding network (CeN) as a method to skeletonize the structure dynamically. They took into account the historical significance of all joints in addition to the reliance between connected joints in order to accurately represent the dependence among both joints. Geng et al. [34] considered learning the dependencies of angular information as a supplement to known joint and bone information and developed self-attention enhanced GCN.

Shared or global topologies that aggregate different channel features in the same topology may limit the model performance. The problem may be solved through the use of distinct topology in separate channels (i.e., a nonshared topology). Channel-wise topology refinement graph convexity (CTR-GC), created by Chen et al. [32], improves topology learning. They simultaneously utilized channel-specific correlations and shared topology. First, the shared topology for all channels was constructed using an adjacency matrix, and specific topologies were constructed using a correlation modeling function. The shared topology was then refined with specific topologies. Finally, the features of each channel graph were aggregated. Similar to this, a pseudo graphical CNN with spatial and channel-wise activation (PGCN-TCA) was created by Yang et al. [17]. To accurately represent joint functional and informal interactions, they used a learnable matrix for each layer instead

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

of a fixed adjacent matrix. Jang et al. [35] designed a common intersection graph convolution (CI-GC) to capture the overlapping information between neighbors. Later, these local-global features are aggregated to deliver a full representation.

Many skeletal action recognition domains have embraced transformer approaches after their notable success in natural language processing (NLP) and computer vision [17, 28, 36]. Transformer methods can capture long-range spatiotemporal dependencies and correlations of joint motion patterns for more accurate performance. They help in making appropriate decisions as they focus on informative parts of the input space. We have seen works that utilized spatial and temporal attention and transformers for violence detection applications [9, 22–24]. For each of the temporal and space–time streams, the process of self-attention was taken into account in these investigations. However, in the area of action recognition, scientists have recently developed advanced skeleton-based processes such as channel attention [37, 38] and GATs [39, 40]. Pang et al. [33] proposed a global attention network that extracts the global graph using both spatial and temporal attentions. Alsarhan et al. [37] designed attentive channel-wise correlation graph convolution (ACC-GC) that uses channel-wise correlations for feature extraction and enhances the learned shared topology. Sun, Wang, and Dai [38] designed a channel-wise attention module that fuses a topological map with multichannel joint weights to capture channel-wise node attention for different actions.

Addressing the task of the action identification problem, a skeleton-based GAT was presented in [39, 40]. Hu, Liu, and Feng [39] presented the spatial Time Tree Attentiveness Net (STGAT), a system that connects nodes from local neighbors in both geographical and temporal domains to create local graphs, and their relationships are dynamically constructed. This means that nodes can aggregate messages from all spatial–temporal neighbors. However, because GAT computes static attention only, this might limit the expressivity of the attention mechanism. Therefore, Rahevar et al. [40], based on the ST-GCN network, created the Spatial Time Dynamic Graphite Attention Network (ST-DGAT). They were input into the ST-DGAT networks following the creation of the spatiotemporal graph, and the dynamic attention coefficient was computed by fixing the internal operation order in the GAT. On the other hand, Huo, Cai, and Meng [41] designed an independent dual GAT to ensure that different attention modules do not conflict with one another. Pang, Lu, and Lyu [42] proposed a GCN-transformer network that ensembles two parallel streams, specifically GCN and Transformer.

3. Proposed Method

This section presents our proposed ESTS-GCN model for violence detection. Figure 2 illustrates the general architectural design of the proposed model. The proposed architecture comprises three parallel pipelines: Spatial Self-Attention with Multi-Scale Temporal (STML), Spatial Channel-wise with Multi-Scale Temporal (SCML), and Spatial Graph attention with Multi-Scale Temporal (SGML). All pipelines receive a sequence of skeletal graphs as input, and their output is fed into a fusion layer for the identifying of violent occurrences. In each pipeline, we adopted a decoupled architecture in which the spatial and temporal features are processed independently. Three different spatial modules were designed: spatial channel-wise topologies (SC), spatial self-attention (ST), and spatial graph (SG) attention. We used a multilevel temporal modeling (ML) framework for the temporal module. The temporal module output is pooled using an overall average pooling and input into a nonlinear activation function to generate the final result of each pipeline. The following sections discuss our architecture and its different pipelines in detail.

3.1. Preliminaries. $G \in \mathbb{R}^{N\times C\times T}$ is the format of the input skeletal graph order, where N, C, and T stand for the number of nodes (joints), channels, and frames, accordingly. When the skeleton graph is first shown as $G = (V, E, X)$, the set of edges is E, the collection of vertex attributes is X, and the set of nodes (vertices) depicting the joints in humans are $V = \{v_1, v_2, v_3, \ldots, v_N\}$. Calculations for E and X change based on the model structure.

3.2. Ensemble Model Architecture. The architecture of the proposed ESTS-GCN model comprises three parallel pipelines STML, SCML, and SGML (see Figure 2). Each pipeline comprises spatial and multiscale temporal modules. We designed three unique modules (SC, ST, and SG) for the spatial part of each pipeline and used a multilevel temporal modeling framework for the temporal module. Multiple blocks (B) of the spatial and temporal modules are utilized in the architecture. The input-mapped convolutions are first fed into the spatial module. The generated results (Z) are then fed into the temporal module. The temporal module output is pooled using an overall average pooling and fed into a nonlinear activation function (see equation (1)).

$$X^{\text{out}} = \sigma(ML(Z)), \tag{1}$$

where ML is the multiscale temporal module, Z is the output from one of the three spatial modules (SC, ST, or SG), and the function of activation σ carries out an irregular change to get the final output of each pipeline X^{out}. Finally, results generated from all pipelines are fused to obtain the ESTS-GCN final prediction of the violent occurrence using equation (2).

$$\text{Prediction} = \sum \theta_i X_i^{\text{out}}, \tag{2}$$

where i represents the pipelines (SCML, STML, or SGML), θ_i is the pipeline's weight, and X_i^{out} is the output of a specific pipeline. In the following sections, we will discuss our spatial and multiscale temporal modules in detail.

3.2.1. Spatial Channel-Wise Topologies Module (SC). In most GCN existing studies, features were aggregated from all channels in a single topology. However, this approach limits

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

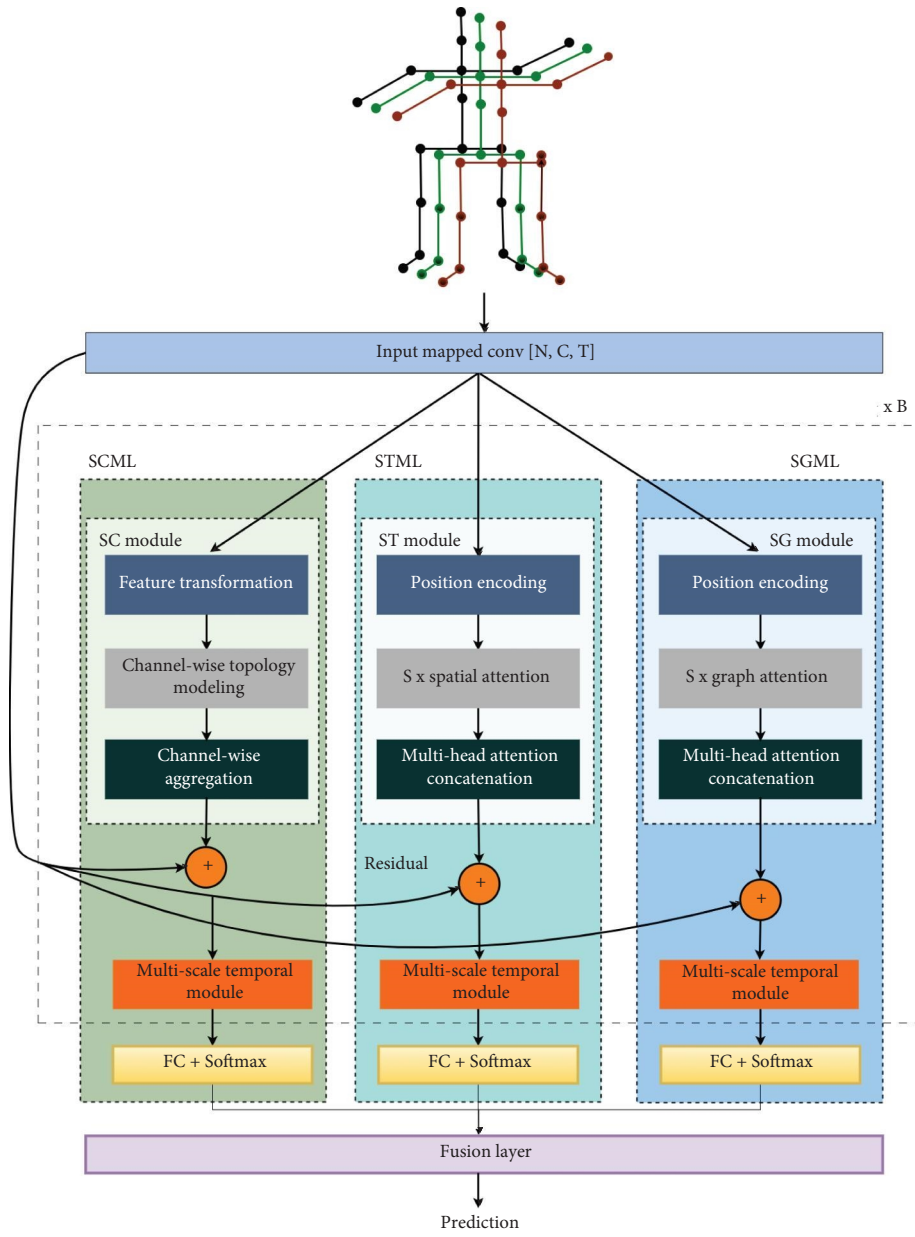

Figure 2: The ESTS-GCN model overall architecture. Three parallel pipelines are utilized: spatial channel-wise with multi-scale temporal (SCML), spatial Self-Attention with Multi-scale Temporal (STML), and spatial graph attention with multi-scale temporal (SGML). Each pipeline comprises a spatial module that extracts local spatial data and a multiscale temporal analysis module for capturing long-term temporal relationships. A sequence of B blocks is deployed for each pipeline, and results are fed into a global average pooling layer. Lastly, FC and soft maximum procedures are employed to obtain the final prediction of violent occurrences.

feature extraction flexibility, as different motion types can be represented by different channels. For each motion feature, correlations between joints might vary for different channels, including self-loop connections, inward connections, outward connections, and all-connections (all joints). Therefore, in this module, we generate a channel-specific (channel-wise) topology for each channel, along with the shared topology. The topologies of the channels are refined with the shared topology, and finally, the features of all channels are aggregated. Figure 3 presents an overview of the architecture of our spatial channel-wise topologies module. This module follows an architecture similar to that in [32] and has three main steps: feature transformation, channel graph modeling, and feature aggregation.

3.2.1.1. Initial Graph and Shared Topology. This module has a learnable shared topology that is learned through back-propagation and is presented as a matrix-adjacency

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

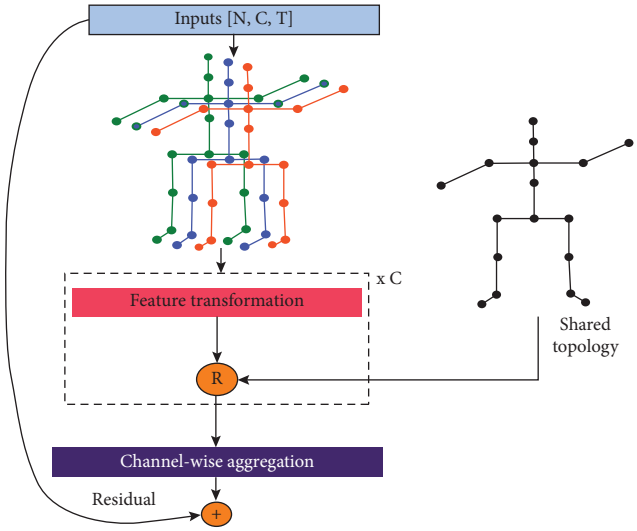

FIGURE 3: The architecture overview diagram of the spatial channel-wise topologies module.

$M \in \mathbb{R}^{N\times N}$ where m_{ij} is the correlation coefficient of (v_i, v_j). The set of neighbors for vertex (v_i) is identified as $\{v_j \mid m_{ij} \neq 0\}$. The set of features for vertex v_i is represented as $x_i \in \mathbb{R}^C$.

3.2.1.2. Feature Transformation. A simple linear transformation function is utilized. It takes $X \in \mathbb{R}^{N\times C}$ as an input and transforms it into $\tilde{X} \in \mathbb{R}^{N\times C'}$ using the weight matrix $W \in \mathbb{R}^{C\times C'}$. Note that $x_i \in \mathbb{R}^C$ represents the features of vertex v_i and the transformation XW computes the correlation information between different channels (see equation (3)).

$$\tilde{X} = XW, \tag{3}$$

where $\tilde{X} \in \mathbb{R}^{N\times C'}$ is the features after transformation and $W \in \mathbb{R}^{C\times C'}$ is the weight matrix.

3.2.1.3. Channel Graphs. Channel-wise topologies R are calculated using the learned correlation of specific channels $L \in \mathbb{R}^{N\times N\times C'}$. Two correlation modeling functions are utilized: distance correlation and multilayer perceptron (MLP), as in [32]. After calculating L, it is refined with M to obtain R (see equation (4)). The intensity of the refinement is adjusted using a trainable scalar α. Finally, channel graphs are constructed using the refined topologies R and transformed features $\tilde{X}$ of that specific channel, which represent the joint relations for specific types of motion features.

$$R = \mathscr{R}(L, M) = M + \alpha.L, \tag{4}$$

where M is a shared topology and L is channel-specific correlations.

3.2.1.4. Feature Aggregation. Equation (5) shows the final function that includes all the refined channel graphs. The aggregated features for each channel graph are concatenated to attain the final output Z.

$$Z = \mathscr{A}(\tilde{X}, R), \tag{5}$$

where $\mathscr{A}$ is the aggregation function, $\tilde{X}$ is the transformed input features, and R is channel-wise topologies.

3.2.2. Spatial Self-Attention Module (ST). One of Google's deep learning architectures, the transformer, has seen notable success in NLP. Self-attention is the primary mechanism behind the transformer model. It can learn input global dependencies and identify the important part of the sequence input by considering several other inputs simultaneously. In addition, it is not compulsory with self-attention to have previous knowledge of the relations between elements, which makes it a flexible approach to discover useful patterns. Moreover, the parallelizability and computational complexity of the self-attention mechanism are superior to those of other methods [43]. Moreover, because the number of human joints is limited, computational costs associated with using a self-attention mechanism will be comparatively low in skeleton data. These reasons make self-attention a great candidate for detecting violent actions.

This module utilizes the self-attention technique to determine the relative position relationship of all joints regardless of the natural human body structure. Figure 4 illustrates how the self-attention mechanism identifies strong and weak relations of the right shoulder (joint) with all other joints, regardless of human naturally linked joints (neck and right elbow). In the figure, we used thicker lines to illustrate stronger relationships. The relationship of joints with both naturally connected and unconnected joints would reveal important information about the human pose and actions (e.g., violent actions).

3.2.2.1. Position Encoding. Before feeding skeleton data into the neural networks as tensors, a position encoding module is used to provide identity for all joints in the frame because there is no prior information about joint indexes, orders, or structures. To give each joint a unique identity, we apply cosine and sine functions using diverse frequencies as in [43] (see equations (6) and (7)).

$$\mathrm{PE}(e, 2j+1) = \cos\left(\frac{e}{10000^{(2j/D_{\mathrm{in}})}}\right), \tag{6}$$

$$\mathrm{PE}(e, 2j) = \sin\left(\frac{e}{10000^{(2j/D_{\mathrm{in}})}}\right), \tag{7}$$

where D_{in} is the amount of the input channels (dimensions of embedding), e is the element's status, j is the dimension of the vector for every location, and PE is the function of positional encoding.

3.2.2.2. Self-Attention Map. The fundamental attentiveness function introduced in [44] is used to calculate the skeletal attention map $A \in \mathbb{R}^{M\times M}$ in this spatial module. The given input sequence is converted into three features vectors: value (**V**), key (**K**), and query (**Q**). Equation (8) illustrates how

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

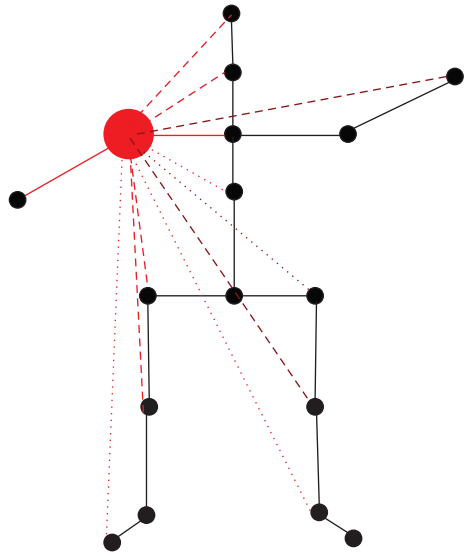

FIGURE 4: The joints relationship in the spatial self-attention module. The edges are of different sizes, representing various degrees of connectivity strength.

the mapping of the query with an array of (key, value) pairings is the function's result.

$$A = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\top}}{\sqrt{D}}\right)\mathbf{V}, \tag{8}$$

where $\mathbf{Q}, \mathbf{K}^{\top}$, and $\mathbf{V}$ are the query's feature vectors, transposed key, and value's feature vectors, respectively, and A is the attention score. Furthermore, D denotes the number of channels (dimensions) for the key $(\mathbf{K})$.

For obtaining an attention map for a node i to all other nodes, we use the following equation:

$$A_i = \text{Attention}\left(\mathbf{q}_i, \mathbf{k}_j, \mathbf{v}_j\right) = \sum_{j=1}^{N} \text{softmax}\left(\frac{\mathbf{q}_i \mathbf{k}_j^{\top}}{\sqrt{D}}\right)\mathbf{v}_j, \tag{9}$$

where A_i, $\mathbf{q}_i, \mathbf{k}_j^{\top}$, and $\mathbf{v}_j$ are the attention score for node i, the feature vector of query for node i, the transposed key, and value of corresponding node j, respectively.

The dynamic weight for each frame t is calculated using $\text{softmax}(\mathbf{Q}\mathbf{K}^{\top}/\sqrt{D})$ which represents the relative importance of a key $(\mathbf{K})$ for that particular query $(\mathbf{Q})$. Multiplying the calculated weights by the values $(\mathbf{V})$ will weigh the original input and by calculating the sum of the weighted vectors, we obtain the final output (see equation (10)).

$$Z = \sigma\left(AX^{\text{in}}\right), \tag{10}$$

where A is the weight of attention in the matrix, X^{in} is the order of inputs, and σ is a function of activation that applies nonlinear processing to get the desired result.

3.2.2.3. Multihead Concatenation. To provide our attention module with greater power to encode multiple relationships, we implement a multihead attention mechanism. Multiple independent and separated attention calculations (heads) are performed in parallel, and the final attention score is calculated by concatenating the data from those heads.

3.2.3. Spatial Graph Attention Module (SG). In the self-attention module, we considered the relationship between all joints regardless of the natural structure of the human joints. However, this might reduce the generalizability of the model and obscure the actual graph structure of the skeleton data. GATs are new architectures of neural networks designed specifically to leverage the attention mechanism on graph-structured data.

In our model, we leverage the attention mechanism into our topology by updating the vertex input features with the learned self-attention scores (the dynamic weight) with its neighboring vertices using equation (11).

$$Z = AX^{\text{in}}M, \tag{11}$$

where A is the dynamic weight of each value calculated using equation (8), X^{in} is the vertex input features, and $M \in \mathbb{R}^{N\times N}$ is the matrix-adjacency of the shared topology. Figure 5 illustrates the graph attention mechanism showing the masking process of the original and transformed features.

3.2.4. Multiscale Temporal Module (ML). The temporal module adopts a multiscale temporal modeling architecture similar to that in [45], although we have fewer branches to accelerate the inference speed. The module comprises four branches, all of which have a 1×1 convolution for dimensional reduction. A second layer of 1×5 temporal convolution is adopted in the first two branches with one and two as a dilation rate. The third branch has a 1×3 max pooling layer. Finally, concatenating the results from all branches gives the final result for this module.

4. Experimental Results

The assessment procedure of our proposed ESTS-GCN model for violence detection is discussed in this section, including the three spatial pipelines: Spatial STML, SCML, and SGML. The following sections explain dataset selection and preprocessing, experimental settings, and evaluation results.

4.1. Datasets. Multiple benchmark datasets have been used in the literature for training and evaluating violence detection models. The most commonly used datasets are Real-Life Violence Situations (RLVS) [21], Crowd Violence (CV) [46], Hockey Fights (HF) [47], Movie Fights (MF) [47], and CCTV-Fights [48]. Table 2 provides a brief summary of the features of the datasets that are currently available. For our experiments, as none of the existing datasets is skeleton-based, we created two datasets to train and validate our model, skeleton-based RLVS and NTU-V, both available on the Kaggle platform [50, 51].

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

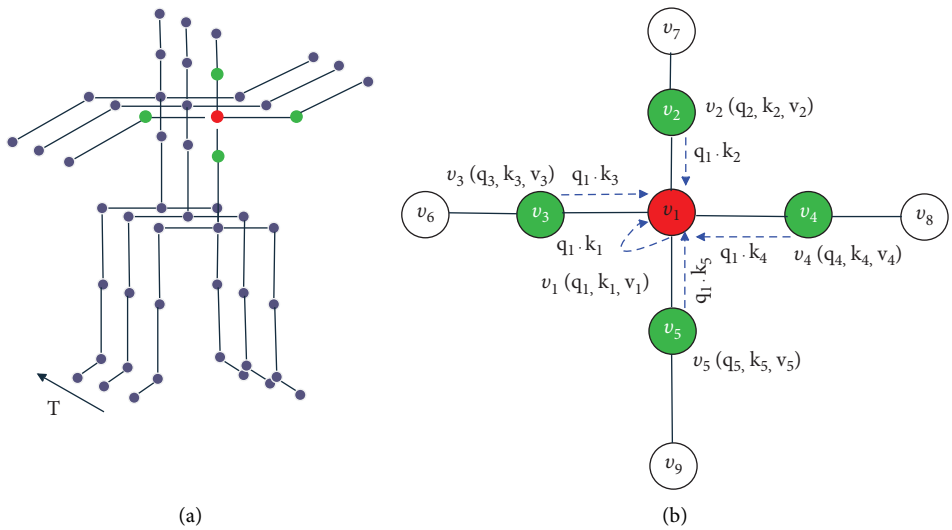

Figure 5: (a) The spatiotemporal sequence of skeletons. The spatial attention graph comprises red vertices connected to multiple green vertices. (b) The graph attention network (GAT) mechanism. A linear projection layer for each vertex in the graph generates **Q**, **K**, and **V** feature vectors. Vertex features are aggregated and updated using the scores from neighboring vertices.

Table 2: Overview of violence detection benchmark datasets.

Dataset	Videos	Violence	Nonviolence	Source
Real-life violence situations (RLVS) [21]	2000	1000	1000	Real street fights from YouTube
Crowd violence (CV) [46]	246	123	123	Violence in crowds from YouTube
Hockey fights (HF) [47]	1000	500	500	Players fighting in hockey match (USA's NHL)
Movie fights (MF) [47]	200	100	100	Scenes from action movies
CCTV-fights (CF) [48]	280	280	0	CCTV real fight from YouTube
Skeleton-based RLVS (ours)	2000	1000	1000	Generated from RLVS [21]
NTU-violence (NTU-V) (our)	11,372	5721	5651	Subset of NTU-RGB + D 120 [49]

4.1.1. Skeleton-Based RLVS. A collection of YouTube videos depicting actual violent incidents is called RLVS [21]. There are 2,000 films in all, both violent and nonviolent, for people of all ages, genders, and ethnicities. As RLVS is a video-based dataset, we generated our skeleton-based RLVS dataset in this study as part of the data preprocessing stage.

4.1.2. NTU-Violence (NTU-V). A human action dataset based on skeletons, NTU-RGB + D 120 [49], is used to train and evaluate action recognition methods. This is a well-known, extensive dataset that is employed in the action recognition industry. It includes 114,481 video clips categorized into 120 different kinds of actions. The videos featured 106 different actors (subjects) performing at 32 different settings and three distinct camera angles. However, NTU-RGB + D was not specifically designed for violence detection tasks. Therefore, we created NTU Violence (NTU-V), a subset of NTU-RGB + D 120, to be suitable for training and evaluating violence detection models. We selected 11,372 samples, 5721 with violent actions, and 5651 with nonviolence actions. The selected violent actions include action classes such as kicking another person, shooting with a gun, punching/slapping another person, wielding a knife, hitting with an object, and pushing another person. The nonviolence samples were selected randomly from other action classes that were not considered violent. Two benchmark datasets are used, Cross Setup (X-Set) and Cross Subject (X-Sub), that concentrate on samples from different actors and contexts.

4.2. Dataset Preprocessing. For the RLVS dataset, we used the OpenPose [12] model as a human pose detection model to generate the skeleton data. As shown in Figure 1, a series of RGB frames (from a video file) is fed into the OpenPose model which generates a JSON file for each frame, containing key points' data. OpenPose provides 25 human body key points in 2D (x, y) along with their confidence value (scores). It is also a multiperson detection model that can detect multiple skeleton data from the same frame. However, in this study, the maximum number of people is set to two for simplicity and to reduce the computational cost. Subsequently, a single video data are combined from the separated JSON files and reformatted to be fed to our models.

To visually evaluate the dataset preprocessing stage, we visualized a sample of our proposed dataset (skeleton-based RLVS dataset) with its corresponding skeleton data (see Figure 6). The video sample contains a violent action between two persons. Figure 6(a) shows a part of a video sample for image-based data. However, Figure 6(b) shows the skeleton sequence, which represents human bodies within key points.

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

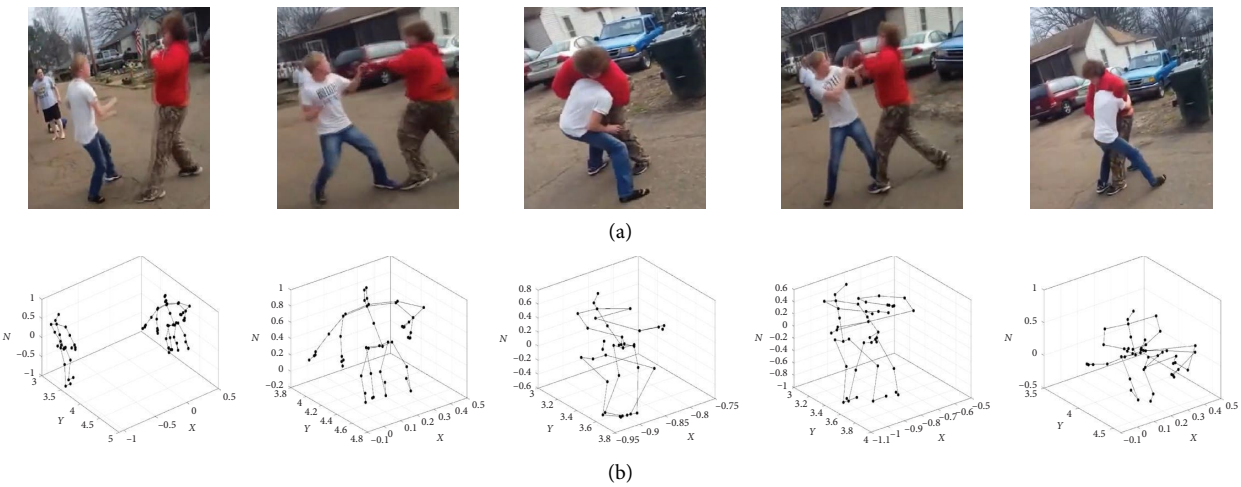

Figure 6: Visualizations for a violent action. (a) A part of a video sample for image-based data. (b) The skeleton sequence, which represents human bodies within key points.

For the NTU-V dataset, it was originally (from NTU-RGB + D 120 [49]) formatted as three-dimensional skeletal data, and 25 joint coordinates were recorded for each person in a single frame.

4.3. Implementation Details. We used Python and PyTorch frameworks to build the proposed architecture. The experiments were executed on Google Colab and the Aziz supercomputer (Jeddah, Saudi Arabia). The general configuration for our SCML, STML, and SGML networks was stacked using 9 SC, 9 ST, and 9 SG blocks, respectively. Each block in the STML and SGML modules has four self-attention heads. In the temporal module, we set dilations to one and two with a kernel size equal to five to obtain multiple branches of temporal convolution. The output channels for our networks were 64, 64, 64, 128, 128, 128, 256, 256, and 256.

For both training and evaluation, the RLVS dataset was broken down into 1600 and 400 video samples, respectively. For every sample, the batch sizes had been set at 32 and 300 frames. Two benchmarks, X-Set and X-Sub, were used from the NTU-V dataset. The X-Set was divided into two sets of samples, one with even set IDs and another with uneven IDs, and they were utilized for training and testing, respectively, depending on their IDs. Conversely, X-Sub was divided equally into 53 training participants and 53 testing individuals.

The learning initial rate was fixed at 0.1, while the weight decaying was fixed at 0.0004. SGD and Nesterov momentum were coupled for optimization, and 0.9 was chosen as the hyperparameter. Eighty epochs were employed in the training stage, with the loss function being the cross entropy loss.

4.4. Results. This section discusses our proposed ESTS-GCN model findings on the RLVS and NTU-V datasets. First, we present an evaluation of the performance of each pipeline (SCML, STML, and SGML) for violence and nonviolence samples. Then, we provide a comparative evaluation of all pipelines using a multi-stream structure and a fusion strategy. These evaluations assisted us in selecting the best version of each pipeline for our ESTS-GCN model. After that, an ablation study was conducted to evaluate our spatial and temporal modules' performance independently, including trade-offs between the accuracy and computational workload. Finally, a comparative evaluation of the proposed ESTS-GCN model against other existing approaches in the literature is provided.

4.4.1. Comparative Evaluation of Violence and Nonviolence Samples. This section separately presents each pipeline's performance in the ESTS-GCN model for violence and nonviolence samples. Tables 3, 4, and 5 present the violence, nonviolence, and overall accuracy for the RLVS, NTU-V X-Sub, and NTU-V X-Set datasets, respectively. In general, the accuracy of the violence class was better than that of the nonviolence class, except for the STML pipeline with the RLVS dataset, where the nonviolence accuracy was 90.5% and the violence accuracy was 87%. Moreover, the difference between the accuracy of violence and nonviolence events was insignificant (less than 1.5%) for all pipelines and datasets except for the X-Sub benchmark. The STML and SGML pipelines trained with the X-Sub benchmark identified violence events more accurately than nonviolence events with 7.04% and 3.4% accuracy differences, respectively.

As the accuracy metric value represents only the model's overall accuracy, we extended our evaluation and investigated the confusion matrix, precision, recall, and F1-score for the pipelines of the ESTS-GCN model. Figures 7 and 8 show confusion matrices of the results of the SCML pipeline when trained using the NTU-V (X-Sub (a) and X-Set (b)) and the RLVS datasets, respectively. These confusion matrices display the true positive along with false positive categorization for violence and nonviolence samples.

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 3: Evaluation of maximum accuracy of each pipeline in the ESTS-GCN model on the RLVS dataset for violent and nonviolence samples in the joint stream.

Model	Violence	Nonviolence	Overall accuracy
SCML	**88.5**	88.42	88.46
STML	87.0	**90.52**	88.72
SGML	**87.5**	87.37	87.45

Note: The bold values are the best/highest values.
Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

Table 4: Evaluation of maximum accuracy of each pipeline in the ESTS-GCN model on the NTU-V dataset (X-Sub benchmark) for violent and nonviolence samples in the joint stream.

Model	Violence	Nonviolence	Overall accuracy
SCML	**91.30**	90.32	90.86
STML	**91.65**	88.25	90.13
SGML	**93.40**	86.36	90.24

Note: The bold values are the best/highest values.
Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

Table 5: Evaluation of maximum accuracy of each pipeline in the ESTS-GCN model on the NTU-V dataset (X-Set benchmark) for violent and nonviolence samples in the joint stream.

Model	Violence	Nonviolence	Overall accuracy
SCML	**91.98**	91.40	91.69
STML	**90.29**	88.94	89.62
SGML	**90.73**	89.62	90.18

Note: The bold values are the best/highest values.
Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

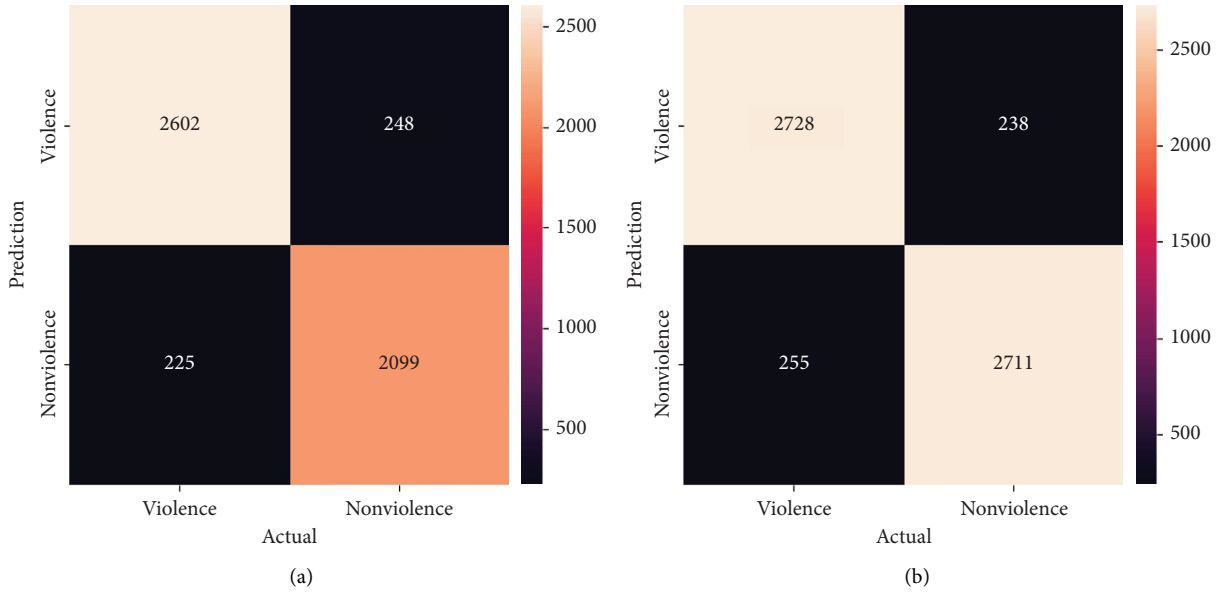

Figure 7: Confusion matrices (a) and (b) demonstrate the results of the SCML pipeline using the NTU-V dataset for the benchmarks X-Sub and X-Set, respectively.

Using the true/false positive (violence) and negative (nonviolence) values, we calculated the precision, recall, F1-score, and accuracy values for each pipeline in the ESTS-GCN model on NTU-V (X-Sub benchmark), NTU-V (X-Set benchmark), and the RLVS datasets (see Table 6). It can be observed from the table that precision values were generally high across all pipelines and datasets, with SGML achieving the highest precision on NTU-V (X-Sub) (93.40%). This indicates that all pipelines were good at avoiding false positives. The variability of recall was greater than that of precision, and it was generally lower than or equal to precision in the majority of situations. The SCML

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

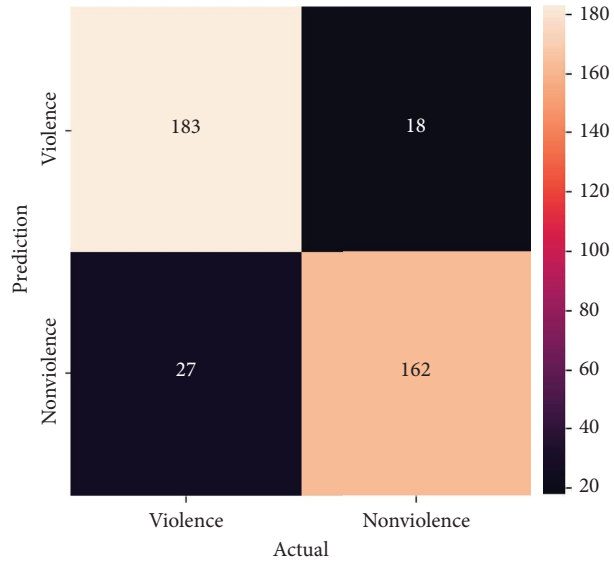

Figure 8: The confusion matrix demonstrates the results of the SCML pipeline for the RLVS dataset.

pipeline achieved the highest recall on NTU-V (X-Sub) (92.04%), which means it caught the most true positives. When precision is higher than recall, it suggests that the model is more conservative in making positive predictions. This means it is better at avoiding false positives but may miss some true positives (lower recall). This trend was mainly seen in the SGML pipeline. On the other hand, when recall is higher than precision, it means that the model captures more true positives but also makes more false positives, which occurred in some STML configurations.

The F1-score tends to be close to the precision and recall values because it combines both metrics in a balanced way. For example, on the NTU-V (X-Sub) dataset, SCML achieved an F1-score of 91.67%, which is very close to its precision (91.30%) and recall (92.04%). The highest F1-score across all datasets was in SCML on NTU-V (X-Set) (91.71%). Accuracy is usually aligned with high precision and recall. The SCML pipeline on NTU-V (X-Set) had the highest accuracy of 91.69% and generally correlates well with the F1-score since both depend on a balance of precision and recall.

4.4.2. Comparative Evaluation of Pipelines Using a Multistream Structure. In this section, we evaluated each pipeline of the ESTS-GCN model using a multistream fusion technique, as in [52]. Four different streams (inputs) were used: (1) a joint stream comprising coordinates from the original skeleton input; (2) a bone stream comprising the difference of spatial coordinates; (3) a joint motion stream; and (4) a bone motion stream. The third and fourth streams consider differentials in the temporal dimension. The results from all streams were also fused with different weights depending on their performance. Finally, the final weighted sum score was used for prediction.

Table 7 displays the accuracy results of three pipelines (SCML, STML, and SGML) for the four streams on the RLVS dataset. All pipelines exhibit the outcomes of ensemble learning for both joint and bone streams (J + B) as well as all four streams (4S). The STML pipeline had the best accuracy 88.72%, followed by SCML (88.46%), and SGML (87.69%). The joint stream produced the best accuracy, approximately 89% for both SCML and STML over all other streams. For the SGML pipeline, the bone motion stream had the best accuracy (87.69%) over other streams. In general, the multistream strategy had little effect on the accuracy of pipelines trained using the RLVS dataset, which might be attributed to the diversity of dataset samples.

However, the performance of the pipelines was significantly different using the NTU-V dataset. Table 8 shows the outcomes of the pipelines trained using the NTU-V dataset for the X-Sub benchmark, and Table 9 shows the outcomes for the X-Set benchmark. The fusion results of the four streams (4S) had the best accuracy on both benchmarks, where the accuracy increased by 1% to 2% compared with the best single steam result. The best overall accuracy among all pipelines was 92.36% for the SCML pipeline with four streams' fusion on the X-Set benchmark. The other pipelines had closer results for the four streams with an average accuracy of 91.3%. In contrast, combining the joint and bone flows did not improve the results.

In terms of single-stream accuracy, the joint stream outperformed all other single streams across all pipelines on both benchmarks. The joint stream accuracy scores on the X-Sub benchmark were 90.8%, 90.13%, and 90.24%, and on the X-Set benchmark, they were 91.69%, 89.62%, and 90.18% for SCML, STML, and SGML. This proves that all pipelines perform well with the original skeleton coordinates, and manipulating the input data is not necessary in this case unless the four-stream results are preferred and a 1%-2% improvement cannot be compromised. A trade-off between the accuracy and computational workload should be considered. In our ESTS-GCN model, we used the joint stream as an input as it proved to be the most reliable and informative across all pipelines, providing consistent and high performance.

Additionally, we have expanded our assessment and provided a comparative analysis of all pipelines (SCML, SGML, and STML) using different streams of data (Joint,

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 6: Precision, recall, F1-score, and accuracy for each pipeline in the ESTS-GCN model on the joint stream of NTU-V (X-Sub benchmark), NTU-V (X-Set benchmark), and the RLVS datasets.

Pipeline	Dataset	Precision	Recall	F1-score	Accuracy
SCML	NTU-V(X-Set)	91.98	91.45	91.71	91.69
SCML	NTU-V(X-Sub)	91.30	92.04	91.67	90.86
SCML	RLSV	91.04	87.14	89.05	88.46
STML	NTU-V(X-Set)	90.29	89.09	89.69	89.62
STML	NTU-V(X-Sub)	91.65	90.54	91.09	90.12
STML	RLSV	87.00	90.63	88.78	88.72
SGML	NTU-V(X-Set)	90.73	89.73	90.23	90.17
SGML	NTU-V(X-Sub)	93.40	89.36	91.34	90.24
SGML	RLSV	87.50	87.94	87.72	87.44

Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

Table 7: Comparison of maximum accuracy of each pipeline in the ESTS-GCN model with four different streams on the RLVS dataset.

Model	Joint	Bone	Joint motion	Bone motion	J + B	4S
SCML	**88.46**	87.18	86.67	86.66	88.06	88.97
STML	**88.72**	87.18	81.10	86.15	87.95	87.85
SGML	87.44	86.41	84.36	**87.69**	86.94	87.55

Note: The bold values are the best/highest values.
Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

Table 8: A comparison of maximum accuracy of each pipeline in the ESTS-GCN model with four distinct streams on the NTU-V dataset (X-Sub benchmark).

Model	Joint	Bone	Joint motion	Bone motion	J + B	4S
SCML	90.86	86.68	86.75	84.87	90.69	**91.63**
STML	90.13	84.80	86.10	82.88	90.63	**91.10**
SGML	90.24	85.58	85.59	84.31	90.80	**91.27**

Note: The bold values are the best/highest values.
Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

Table 9: A comparison of maximum accuracy of each pipeline in the ESTS-GCN model with four distinct streams on the NTU-V dataset (X-Set benchmark).

Model	Joint	Bone	Joint motion	Bone motion	J + B	4S
SCML	91.69	87.95	88.76	85.18	91.64	**92.36**
STML	89.62	84.68	87.24	82.55	90.16	**91.00**
SGML	90.18	87.02	87.51	84.12	91.34	**91.61**

Note: The bold values are the best/highest values.
Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

Bone, Joint motion, Bone motion, and 4s) with our proposed ESTS-GCN model. Table 10 presents the performance metrics (precision, recall, F1-score, and accuracy) for all pipelines and the ESTS-GCN model on the NTU-V (X-Sub) datasets. The joint stream consistently showed the highest performance across all pipelines, particularly in precision and accuracy, with SGML achieving the best precision (93.40%) in this stream. The bone motion stream exhibited the lowest overall performance, particularly in the SCML and STML pipelines. The 4s stream, which combines multiple feature types, performed very well across all pipelines, indicating that using combined features boosts performance. The proposed ESTS-GCN (ensemble of the three pipelines) model delivered the highest overall performance across all metrics. It achieved the best precision (94.18%), recall (91.73%), F1-score (92.94%), and accuracy (92.11%). This suggests that integrating the strengths of multiple pipelines can lead to superior classification results, outperforming any individual pipeline.

4.4.3. Ablation Study. In this section, we independently evaluate the performance of our Spatial and Temporal modules, including trade-offs between the accuracy and computational workload. We have built and trained models composing independent modules of our ESTS-GCN model to verify their effectiveness. Table 11 lists the accuracy, training time, and number of parameters for the joint stream on the NTU-V dataset (X-Set benchmark). First, we present the results of the independent spatial (SC, ST, and SG) and

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 10: Precision, recall, F1-score, and accuracy for all pipelines and the ESTS-GCN model with four distinct streams on the NTU-V dataset (X-Sub benchmark).

Pipeline	Stream	Precision	Recall	F1-score	Accuracy
SCML	Joint	91.3	92.04	91.67	90.86
SCML	Bone	86.35	89.13	87.72	86.68
SCML	Joint motion	88.49	87.57	88.03	86.74
SCML	Bone motion	84.63	86.98	85.79	84.56
SCML	4s	92.46	92.36	92.41	91.63
SGML	Joint	93.4	89.36	91.34	90.24
SGML	Bone	85.61	87.9	86.74	85.58
SGML	Joint motion	83.54	89.58	86.46	85.58
SGML	Bone motion	87.65	84.45	86.02	84.31
SGML	4s	93.72	90.94	92.31	91.4
STML	Joint	91.65	90.54	91.09	90.12
STML	Bone	84.18	87.71	85.91	84.79
STML	Joint motion	88.11	86.8	87.45	86.06
STML	Bone motion	85.02	84.07	84.54	82.88
STML	4s	93.12	90.89	91.99	91.07
ESTS-GCN	3 pipelines	94.18	91.73	92.94	92.11

Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

temporal (ML) modules. The Spatial Channel-wise Topologies Module (SC) has the least number of parameters and gives the lowest accuracy (85%), while the Spatial Graph Attention Module (SG) has the largest number of parameters and provides the highest (87%) accuracy among all spatial modules. On the other hand, the temporal module (ML) has less than 1M parameters and gives the highest accuracy (88%) among all independent modules. We have also listed the results of the three spatial-temporal pipelines, STML, SCML, and SGML, as well as the proposed ESTS-GCN model. It is observed that integrating the temporal module comprising 0.7 M parameters in the spatial-temporal pipelines has improved the accuracy by 5% on average for all methods.

Our proposed ESTS-GCN model achieved the highest accuracy (93%) among all methods, underscoring the effectiveness of integrating multiple models and various feature types. The training time and parameter details are not provided because the ESTS-GCN model comprises three parallel pipelines (SCML, SGML, and STML) that were trained independently. During the inference/production phase, pipelines can run simultaneously in a parallelized way. Each pipeline can make predictions on a given input data at the same time, and the final ensemble decision can be made once all predictions are available. In this way, computational time will be minimized. In future work, we will investigate the trade-offs between accuracy and computational workload during deployment and consider practical optimization techniques.

Figure 9 provides a visual representation of the model accuracy and the number of parameters for all independent and spatial-temporal models. The SCML model outperformed all other spatial-temporal models with the least number of parameters. It also showed around 7% improvement from its independent spatial and temporal models. On the other hand, STML and SGML models have similar results in terms of both accuracy and the number of parameters. Figure 10 compares the maximum accuracy values of STML, SCML, SGML, and ESTS-GCN models for RLVS, NTU-V(X-Set), and NTU-V(X-Sub) datasets. The proposed ESTS-GCN model outperformed all independent spatial–temporal models for all datasets. The SCML model was the second-best for the NTU-V(X-Set) and NTU-V(X-Sub) datasets, while the STML was the second-best for the RLVS dataset.

4.4.4. Comparative Evaluation of ESTS-GCN With Existing Approaches. In this section, we evaluate the performance of ESTS-GCN against other existing approaches in the literature. Table 12 lists maximum accuracy values and the number of parameters of various models trained with skeleton-based RLVS, NTU-V(X-Set), and NTU-V(X-Sub) datasets.

As the skeleton-based GCN technique is regarded as a new approach we are introducing to the violence detection problem, the performance comparison with existing work is not direct. In particular, there is no public skeleton-based dataset for violent events, and we used our modified version of the existing datasets in this study.

First, we listed the results of three image-based (RGB-based) models (Soliman et al. [21], AlDahoul et al. [53], and Romas, Raudonis, and Dervinis [54]) that were trained with the RLV dataset to provide a baseline for our comparison. However, as these methods were designed for image-based (RGB-based) datasets, we were unable to train them using the skeleton-based NTU-V(X-Set) and NTU-V(X-Sub) datasets.

To overcome the limitation of the existence of studies that adopted a skeleton-based approach and to provide a fair comparison, we trained three different skeleton-based models that were not trained before for the violence detection problem. The ST-GCN [18] model was designed to recognize actions based on skeleton data and adopted a pure spatial-temporal GCN approach. DC-GCN [55] is a decoupling GCN model for action recognition that was proposed to overcome the coupling aggregation limitation of GCN.

Table 11: Ablation study results on the NTU-V dataset (X-Set benchmark) for the joint stream.

Model configuration	Accuracy (%)	Training Time (M)	Parameters
SC (spatial)	85	42	1 M
ST (spatial)	86	42	2.1 M
SG (spatial)	87	104	2.3 M
ML (temporal)	88	157	0.7 M
SCML (spatial–temporal)	92	198	1.4 M
STML (spatial–temporal)	90	182	2.6 M
SGML (spatial–temporal)	90	186	2.7 M
ESTS-GCN (spatial–temporal)	93	N/A	N/A

Abbreviations: NTU-V, NTU-violence and RLVS, real-life violence situations; SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

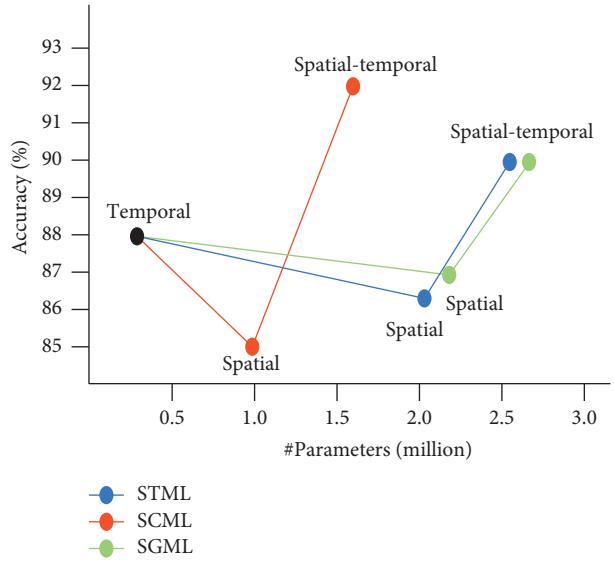

Figure 9: Evaluation of the performance of our spatial and temporal modules on the NTU-V dataset (X-Set benchmark) for the joint stream. SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

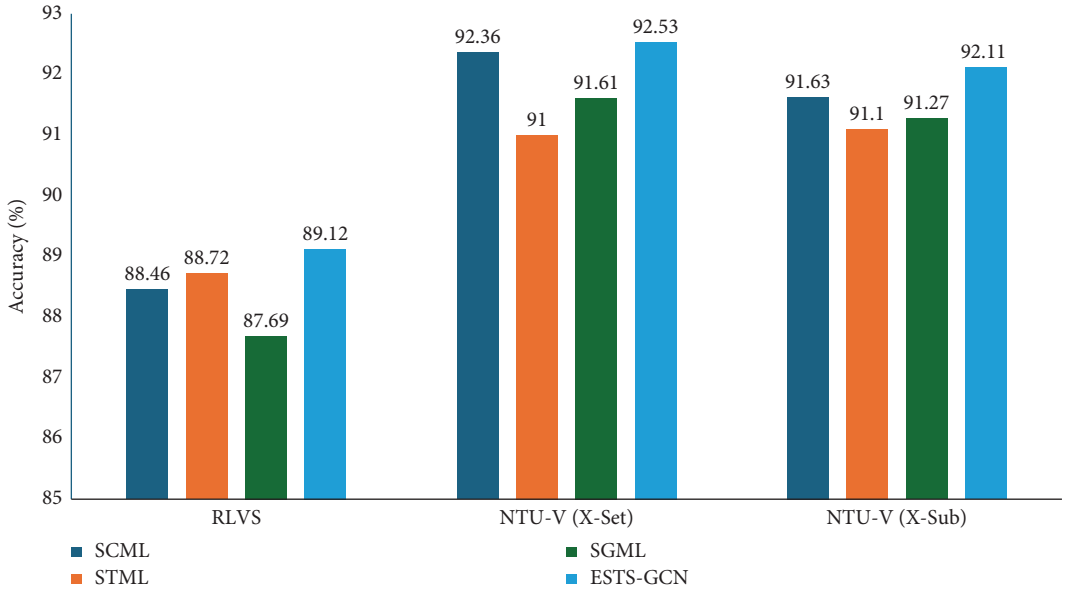

Figure 10: Evaluation of the ESTS-GCN models for RLVS, NTU-V(X-Set), and NTU-V(X-Sub) datasets. NTU-V, NTU-violence; RLVS, real-life violence situations; X-Set, cross setup; X-Sub, cross subject.

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Table 12: A comparison of maximum accuracy of ESTS-GCN with other existing approaches.

Model	Dataset			Parameters (M)
	RLVS	NTU-V(X-set)	NTU-V(X-Sub)	
Soliman et al. [21]	88.20	N/A	N/A	2.1
AlDahoul et al. [53]	73.35	N/A	N/A	1.3
Vijeikis, Raudonis, and Dervinis [54]	82.0	N/A	N/A	4.1
ST-GCN [18]	86.92	88.30	81.50	3.1
DC-GCN [55]	83.1	85.7	84.3	3.4
HD-GCN [19]	87.6	92.4	91.7	1.6
Channel-wise (SCML)	88.46	92.36	91.63	1.4
Self-attention (STML)	88.72	91.00	91.10	2.6
Graph-attention (SGML)	87.69	91.61	91.27	2.7
ESTS-GCN	**89.12**	**92.53**	**92.11**	

Note: The bold values are the best/highest values.
Abbreviations: SCML, spatial channel-wise with multiscale temporal; SGML, spatial graph attention with multiscale temporal; STML, spatial self-attention with multiscale temporal.

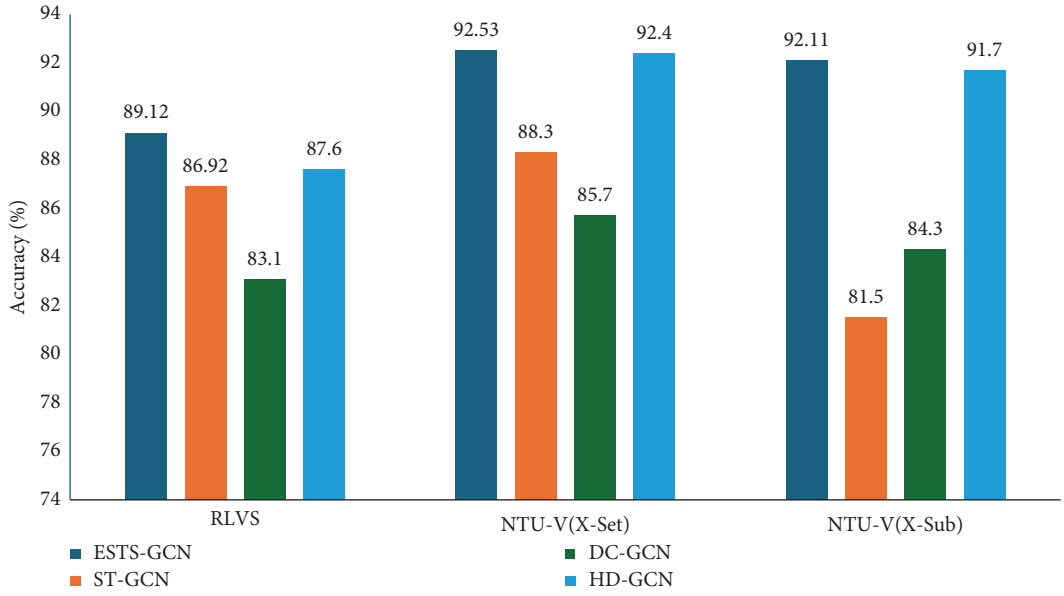

Figure 11: Evaluation of the maximum accuracy of ESTS-GCN and other existing models. DC-GCN, decoupling graph convolutional networks; ESTS-GCN, ensemble spatial-temporal skeleton-based graph convolutional networks; HD-GCN, hierarchically decomposed graph convolutional networks; ST-GCN, spatial-temporal graph convolutional networks.

HD-GCN [19] is also a skeleton-based action recognition model that adopted a hierarchically decomposed-based GCN. In addition, to enrich our comparison, we listed the results of our implementation of three cutting-edge approaches for feature transformation and aggregation: channel-wise topologies (SCML), self-attention (STML), and graph-attention networks (SGML). These models were implemented and trained independently as separate models, and their results were provided for comparison.

As can be observed from Table 12, the ESTS-GCN model outperformed all other approaches. ESTS-GCN maximum accuracy ranged from around 89% to 93% with more than 10% improvement in some cases. For the RLVS dataset, channel-wise topologies (SCML) and self-attention (STML) were the second-best approaches after ESTS-GCN with less than 1% difference. In general, channel-wise topologies (SCML), self-attention (STML), and graph-attention networks (SGML) approaches were better than the pure spatial-temporal (ST-GCN) approach. It is also worth pointing out that channel-wise topologies (SCML) performed well with the NTU-V (X-Set) dataset; however, for RLVS and NTU-V (X-Sub) datasets, performance was similar.

Figure 11 compares the maximum accuracy values of ESTS-GCN and other existing skeleton-based models (ST-GCN [18], DC-GCN [55], and HD-GCN [19]) for RLVS, NTU-V(X-Set), and NTU-V(X-Sub) datasets. The proposed ESTS-GCN model outperformed all the existing skeleton-based models for all datasets. The HD-GCN model was the second-best model after ESTS-GCN, with less than 1% difference for the NTU-V(X-Set) and NTU-V(X-Sub) datasets, and around 2% difference for the RLVS dataset.

These findings demonstrate the efficacy of our proposed model in enhancing model accuracy while detecting violent occurrences, especially when various relationships and dependencies between skeleton data are considered. The main

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

concern with the ESTS-GCN model is the computation workload, which can be resolved by parallelism as pipelines are independent and can run simultaneously. In future work, we will examine the deployment-related trade-offs between accuracy and computing workload and take into account useful optimization strategies.

5. Conclusion

Human safety is one of the essential needs of human beings, and living in a safe community is considered an indicator of the quality of life. Developing robust and accurate smart surveillance systems will help quickly detect and respond to violent actions, which will help in increasing community safety. Smart surveillance systems rely heavily on detection models to understand the context of a scene and to identify certain situations, including violent events.

Existing studies have relied on computer vision technologies to identify violent situations. Specifically, they analyzed the color features of a series of images from video streams to identify violent events. However, these image-based violence detection algorithms suffer from pixel-based noise, lighting, viewpoint, and background interference because they are sensitive to environmental changes. Therefore, in this study, we proposed a skeleton-based violence detection model. Although skeleton-based algorithms approved their robustness in motion and event detection problems, they are still in their early stages for smart surveillance systems and require research attention.

To overcome this gap in the literature, this study explored various GCN methods for violence detection. We designed and investigated a unique ESTS-GCN model for violence detection. ESTS-GCN adopts ensemble-based architecture and utilizes three different approaches for feature transformation and aggregation: channel-wise topologies, self-attention mechanism, and graph-attention networks. In addition, we created two skeleton-based datasets for violent events, skeleton-based RLVS and NTU-V, and provided them to the public, aiming to boost research in this area.

The proposed model architecture was implemented with the three pipelines SCML, STML, and SGML and trained using the RLVS and NTU-V datasets. In general, the ESTS-GCN model outperformed all other approaches. Its maximum accuracy ranged from around 89% to 93% with more than 10% improvement in some cases. A multistream fusion strategy was adopted in the training and evaluation process. The joint stream had the best accuracy among all other single streams for all models, with a maximum accuracy of 91.7% for the SCML model trained with the X-Set benchmark. Similarly, the SCML model achieved the highest accuracy of approximately 92.4% when four different streams were fused.

In the future, we aim to improve the dataset quality and collect more data containing violence and nonviolence events. In addition, we will explore other GCN mechanisms and deeply investigate various metrics such as model size, processing time, and energy efficiency. The trade-offs between accuracy and computational workload during deployment will also be investigated, considering practical and state-of-the-art optimization techniques.

Data Availability Statement

The data presented in this study are openly available in Kaggle at https://www.kaggle.com/datasets/musreaghaseb/skeleton-based-rlvs-dataset and https://www.kaggle.com/datasets/musreaghaseb/ntu-violence-dataset.

Conflicts of Interest

The authors declare no conflicts of interest.

Funding

This work was funded by the University of Jeddah, Saudi Arabia, under grant no. UJ-23-DR-156. Therefore, the authors thank the University of Jeddah for its technical and financial support.

Acknowledgments

This work was funded by the University of Jeddah, Saudi Arabia, under grant no. UJ-23-DR-156. Therefore, the authors thank the University of Jeddah for its technical and financial support.

References

[1] O. Elharrouss, N. Almaadeed, and S. Al-Maadeed, *A Review of Video Surveillance Systems* (Amsterdam, Netherlands: Elsevier, 2021).

[2] N. Janbi, I. Katib, A. Albeshri, and R. Mehmood, "Distributed Artificial Intelligence-As-A-Service (DAIaaS) for Smarter IoE and 6G Environments," *Sensors* 20, no. 20 (2020): 5796–5828, https://www.mdpi.com/1424-8220/20/20/5796, https://doi.org/10.3390/s20205796.

[3] N. Janbi, R. Mehmood, I. Katib, A. Albeshri, J. M. Corchado, and T. Yigitcanlar, "Imtidad: A Reference Architecture and a Case Study on Developing Distributed AI Services for Skin Disease Diagnosis Over Cloud, Fog and Edge," *Sensors* 22, no. 5 (2022): 1854, https://doi.org/10.3390/s22051854.

[4] N. F. Janbi and N. Almuaythir, "BowlingDL: A Deep Learning-Based Bowling Players Pose Estimation and Classification," in *2023 1st International Conference on Advanced Innovations in Smart Cities (ICAISC)* (Jeddah, Saudi Arabia, April 2023), https://ieeexplore.ieee.org/document/10085434/.

[5] F. U. Ullah, M. S. Obaidat, A. Ullah, K. Muhammad, M. Hijji, and S. W. Baik, "A Comprehensive Review on Vision-Based Violence Detection in Surveillance Videos," *ACM Computing Surveys* 55, no. 10 (2023): 1–44, https://dl.acm.org/doi/10.1145/3561971.

[6] V. D. Huszar, V. K. Adhikarla, I. Negyesi, and C. Krasznay, "Toward Fast and Accurate Violence Detection for Automated Video Surveillance Applications," *IEEE Access* 11 (2023): 18772–18793, https://ieeexplore.ieee.org/document/10044675/, https://doi.org/10.1109/access.2023.3245521.

[7] M. M. Ali, "Real-time Video Anomaly Detection for Smart Surveillance," *IET Image Processing* 17, no. 5 (2023): 1375–1388, https://doi.org/10.1049/ipr2.12720.

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

[8] D. Choqueluque-Roman and G. Camara-Chavez, "Weakly Supervised Violence Detection in Surveillance Video," *Sensors* 22, no. 12 (2022): 4502, https://www.mdpi.com/1424-8220/22/12/4502, https://doi.org/10.3390/s22124502.

[9] H. A. Huillcen Baca, F. de Luz Palomino Valdivia, I. S. Solis, M. A. Cruz, and J. C. G. Caceres, "Human Violence Recognition in Video Surveillance in Real-Time," *Lecture Notes in Networks and Systems* (2023): 783–795, https://link.springer.com/chapter/10.1007/978-3-031-28073-3_52, https://doi.org/10.1007/978-3-031-28073-3_52.

[10] M. Sharma and R. Baghel, "Video Surveillance for Violence Detection Using Deep Learning," *Lecture Notes on Data Engineering and Communications Technologies* 37 (2020): 411–420, https://link.springer.com/chapter/10.1007/978-981-15-0978-0_40, https://doi.org/10.1007/978-981-15-0978-0_40.

[11] F. J. Rendón-Segador, J. A. Álvarez-García, J. L. Salazar-González, and T. Tommasi, "CrimeNet: Neural Structured Learning Using Vision Transformer for Violence Detection," *Neural Networks* 161 (April 2023): 318–329, https://doi.org/10.1016/j.neunet.2023.01.048.

[12] "GitHub-CMU-Perceptual-Computing-Lab/openpose: OpenPose: Real-Time Multi-Person Keypoint Detection Library for Body, Face, Hands, and Foot Estimation," (2023), https://github.com/CMU-Perceptual-Computing-Lab/openpose.

[13] P. K. Mishra, A. Mihailidis, and S. S. Khan, "Skeletal Video Anomaly Detection Using Deep Learning: Survey, Challenges and Future Directions," *Computers, Materials & Continua* 12 (2022): https://arxiv.org/abs/2301.00114v2.

[14] Y. Su, G. Lin, J. Zhu, and Q. Wu, "Human Interaction Learning on 3D Skeleton Point Clouds for Video Violence Recognition," in *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Springer Science and Business Media Deutschland GmbH, 2020), 74–90, https://link.springer.com/chapter/10.1007/978-3-030-58548-8_5.

[15] B. Omarov, S. Narynov, Z. Zhumanov, A. Gumar, and M. Khassanova, "A Skeleton-Based Approach for Campus Violence Detection," *Computers, Materials & Continua* 72, no. 1 (2022): 315–331, https://www.techscience.com/cmc/v72n1/46897, https://doi.org/10.32604/cmc.2022.024566.

[16] G. Garcia-Cobo and J. C. SanMiguel, "Human Skeletons and Change Detection for Efficient Violence Detection in Surveillance Videos," *Computer Vision and Image Understanding* 233 (2023): 103739, https://linkinghub.elsevier.com/retrieve/pii/S1077314223001194, https://doi.org/10.1016/j.cviu.2023.103739.

[17] H. Yang, Y. Gu, J. Zhu, K. Hu, and X. Zhang, "PGCN-TCA: Pseudo Graph Convolutional Network With Temporal and Channel-Wise Attention for Skeleton-Based Action Recognition," *IEEE Access* 8 (2020): 040–047, https://doi.org/10.1109/access.2020.2964115.

[18] S. Yan, Y. Xiong, and D. Lin, "Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition," in *32nd AAAI Conference on Artificial Intelligence* (March 2018), 7444–7452, https://arxiv.org/abs/1801.07455v2.

[19] J. Lee, M. Lee, D. Lee, and S. Lee, "Hierarchically Decomposed Graph Convolutional Networks for Skeleton-Based Action Recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (October 2023), 444–453.

[20] A. Datta, M. Shah, and N. Da Vitoria Lobo, "Person-on-Person Violence Detection in Video Data," *Proceedings-International Conference on Pattern Recognition* 16, no. 1 (2002): 433–438.

[21] M. M. Soliman, M. H. Kamal, M. A. El-Massih Nashed, Y. M. Mostafa, B. S. Chawky, and D. Khattab, "Violence Recognition From Videos Using Deep Learning Techniques," in *Proceedings-2019 IEEE 9th International Conference on Intelligent Computing and Information Systems, ICICIS* (April 2019), 80–85.

[22] F. J. Rendón-Segador, J. A. Álvarez-García, F. Enríquez, and O. Deniz, "ViolenceNet: Dense Multi-Head Self-Attention With Bidirectional Convolutional LSTM for Detecting Violence," *Electronics* 10, no. 13 (2021): 1601, https://doi.org/10.3390/electronics10131601.

[23] M. S. Kang, R. H. Park, and H. M. Park, "Efficient Spatio-Temporal Modeling Methods for Real-Time Violence Recognition," *IEEE Access* 9 (2021): 76 270–276 285, https://doi.org/10.1109/access.2021.3083273.

[24] A. R. Abdali and A. A. Aggar, "DEVTrV2: Enhanced Data-Efficient Video Transformer For Violence Detection," in *2022 7th International Conference on Image, Vision and Computing, ICIVC 2022* (May 2022), 69–74.

[25] in *Pose Estimation-TensorFlow Lite* (2023), https://www.tensorflow.org/lite/examples/pose_estimation/overview.

[26] "MoveNet: Ultra Fast and Accurate Pose Detection Model *TensorFlow Hub*," (2023), https://www.tensorflow.org/hub/tutorials/movenet.

[27] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A Comprehensive Survey on Graph Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems* 32, no. 1 (1 2021): 4–24, https://doi.org/10.1109/tnnls.2020.2978386.

[28] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-Structural Graph Convolutional Networks for Skeleton-Based Action Recognition," in *2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)* (March 2019), 474–480.

[29] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-Stream Adaptive Graph Convolutional Networks for Skeleton-Based Action Recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (June 2019).

[30] J. Xie, Y. Meng, Y. Zhao, A. Nguyen, X. Yang, and Y. Zheng, "Dynamic Semantic-Based Spatial Graph Convolution Network for Skeleton-Based Human Action Recognition," *Proceedings of the AAAI Conference on Artificial Intelligence* 38, no. 6 (March 2024): 6225–6233, https://ojs.aaai.org/index.php/AAAI/article/view/28440, https://doi.org/10.1609/aaai.v38i6.28440.

[31] F. Ye, S. Pu, Q. Zhong, C. Li, D. Xie, and H. Tang, "Dynamic GCN: Context-Enriched Topology Learning for Skeleton-Based Action Recognition," in *MM 2020 - Proceedings of the 28th ACM International Conference on Multimedia*, 10 (June 2020), 55–63.

[32] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, and W. Hu, "Channel-Wise Topology Refinement Graph Convolution for Skeleton-Based Action Recognition," in *Proceedings of the IEEE International Conference on Computer Vision* (April 2021), 339–348.

[33] C. Pang, X. Gao, Z. Chen, and L. Lyu, "Self-Adaptive Graph With Nonlocal Attention Network for Skeleton-Based Action Recognition," *IEEE Transactions on Neural Networks and Learning Systems* 35 (2023): 1–13, https://doi.org/10.1109/tnnls.2023.3298950.

[34] P. Geng, X. Lu, C. Hu, H. Liu, and L. Lyu, "Focusing Fine-Grained Action by Self-Attention-Enhanced Graph Neural

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Networks With Contrastive Learning," *IEEE Transactions on Circuits and Systems for Video Technology* 33, no. 9 (September 2023): 4754–4768, https://doi.org/10.1109/tcsvt.2023.3248782.

[35] S. Jang, H. Lee, W. J. Kim, J. Lee, S. Woo, and S. Lee, "Multi-Scale Structural Graph Convolutional Network for Skeleton-Based Action Recognition," *IEEE Transactions on Circuits and Systems for Video Technology* 34, no. 8 (2024): 7244–7258, https://doi.org/10.1109/tcsvt.2024.3375512.

[36] C. Plizzari, M. Cannici, and M. Matteucci, "Skeleton-Based Action Recognition via Spatial and Temporal Transformer Networks," *Computer Vision and Image Understanding* 208-209 (2021): https://doi.org/10.1016/j.cviu.2021.103219.

[37] T. Alsarhan, O. Harfoushi, A. Y. Shdefat, N. Mostafa, M. Alshinwan, and A. Ali, "Improved Graph Convolutional Network With Enriched Graph Topology Representation for Skeleton-Based Action Recognition," *Electronics* 12, no. 4 (2023): 879, https://doi.org/10.3390/electronics12040879.

[38] Z. Sun, T. Wang, and M. Dai, "Combining Channel-Wise Joint Attention and Temporal Attention in Graph Convolutional Networks for Skeleton-Based Action Recognition," *Signal, Image and Video Processing* 17, no. 5 (2023): 2481–2488, https://link.springer.com/article/10.1007/s11760-022-02465-z, https://doi.org/10.1007/s11760-022-02465-z.

[39] L. Hu, S. Liu, and W. Feng, "Spatial Temporal Graph Attention Network for Skeleton-Based Action Recognition," 8 (2022), http://arxiv.org/abs/2208.08599.

[40] M. Rahevar, A. Ganatra, T. Saba, A. Rehman, and S. A. Bahaj, "Spatial-Temporal Dynamic Graph Attention Network for Skeleton-Based Action Recognition," *IEEE Access* 11 (2023): 21 546–621 553, https://doi.org/10.1109/access.2023.3247820.

[41] J. Huo, H. Cai, and Q. Meng, "Independent Dual Graph Attention Convolutional Network for Skeleton-Based Action Recognition," *Neurocomputing* 583 (May 2024): 127496, https://doi.org/10.1016/j.neucom.2024.127496.

[42] C. Pang, X. Lu, and L. Lyu, "Skeleton-Based Action Recognition Through Contrasting Two-Stream Spatial-Temporal Networks," *IEEE Transactions on Multimedia* 25 (2023): 8699–8711, https://doi.org/10.1109/tmm.2023.3239751.

[43] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Decoupled Spatial-Temporal Attention Network for Skeleton-Based Action-Gesture Recognition," *Lecture Notes in Computer Science* 12626, no. LNCS (2021): 38–53, https://link.springer.com/chapter/10.1007/978-3-030-69541-5_3, https://doi.org/10.1007/978-3-030-69541-5_3.

[44] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention Is All You Need," *Advances in Neural Information Processing Systems* (2017): 5999–6009.

[45] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, "Disentangling and Unifying Graph Convolutions for Skeleton-Based Action Recognition," *Proceedings - IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 6 (2020): 140–149.

[46] T. Hassner, Y. Itcher, and O. Kliper-Gross, "Violent Flows: Real-Time Detection of Violent Crowd Behavior," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* (April 2012), 1–6.

[47] E. Bermejo Nievas, O. Deniz Suarez, G. Bueno García, and R. Sukthankar, "Violence Detection in Video Using Computer Vision Techniques," in *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6855 (Berlin, Germany: Springer, 2011), 332–339, https://link.springer.com/chapter/10.1007/978-3-642-23678-5_39, https://doi.org/10.1007/978-3-642-23678-5_39.

[48] M. Perez, A. C. Kot, and A. Rocha, "Detection of Real-World Fights in Surveillance Videos," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 5 (May 2019), 2662–2666.

[49] J. Liu, A. Shahroudy, M. Perez, G. Wang, L. Y. Duan, and A. C. Kot, "NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, no. 10 (2020): 2684–2701, https://doi.org/10.1109/tpami.2019.2916873.

[50] "Skeleton-Based RLVS Dataset," (2023), https://www.kaggle.com/datasets/musreaghaseb/skeleton-based-rlvs-dataset/data.

[51] "NTU-violence Dataset," (2023), https://www.kaggle.com/datasets/musreaghaseb/ntu-violence-dataset/data.

[52] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, "Skeleton-Based Action Recognition With Shift Graph Convolutional Network," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (March 2020), 180–189.

[53] N. Aldahoul, H. A. Karim, R. Datta, S. Gupta, K. Agrawal, and A. Albunni, "Convolutional Neural Network-Long Short Term Memory Based IOT Node for Violence Detection," in *3rd IEEE International Conference on Artificial Intelligence in Engineering and Technology, IICAIET 2021* (September 2021).

[54] R. Vijeikis, V. Raudonis, and G. Dervinis, "Efficient Violence Detection in Surveillance," *Sensors* 22, no. 6 (March 2022): 2216, https://www.mdpi.com/1424-8220/22/6/2216, https://doi.org/10.3390/s22062216.

[55] K. Cheng, Y. Zhang, C. Cao, L. Shi, J. Cheng, and H. Lu, "Decoupling GCN With DropGraph Module for Skeleton-Based Action Recognition," in *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2020), 536–553, https://link.springer.com/chapter/10.1007/978-3-030-58586-0_32.

ijis, 2024, 1, Downloaded from https://onlinelibrary.wiley.com/doi/10.1155/2024/2323337 by Readcube (Labtiva Inc.), Wiley Online Library on [12/12/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License