electronics

MDPI

Article

Human Action Recognition of Spatiotemporal Parameters for Skeleton Sequences Using MTLN Feature Learning Framework

Faisal Mehmood [1], Enqing Chen [1,2,*], Muhammad Azeem Akbar [3] and Abeer Abdulaziz Alsanad [4]

[1] School of Information Engineering, Zhengzhou University, No. 100 Science Avenue, Zhengzhou 450001, China; faisalmehmood685@uaf.edu.pk
[2] Henan Xintong Intelligent IOT Co., Ltd., No. 1-303 Intersection of Ruyun Road and Meihe Road, Zhengzhou 450007, China
[3] Department of Software Engineering, Lappeenranta-Lahti University of Technology, 53851 Lappeenranta, Finland; azeem.akbar@lut.fi
[4] College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh 11623, Saudi Arabia; aaasand@imamu.edu.sa
* Correspondence: ieeqchen@zzu.edu.cn

Abstract: Human action recognition (HAR) by skeleton data is considered a potential research aspect in computer vision. Three-dimensional HAR with skeleton data has been used commonly because of its effective and efficient results. Several models have been developed for learning spatiotemporal parameters from skeleton sequences. However, two critical problems exist: (1) previous skeleton sequences were created by connecting different joints with a static order; (2) earlier methods were not efficient enough to focus on valuable joints. Specifically, this study aimed to (1) demonstrate the ability of convolutional neural networks to learn spatiotemporal parameters of skeleton sequences from different frames of human action, and (2) to combine the process of all frames created by different human actions and fit in the spatial structure information necessary for action recognition, using multi-task learning networks (MTLNs). The results were significantly improved compared with existing models by executing the proposed model on an NTU RGB+D dataset, an SYSU dataset, and an SBU Kinetic Interaction dataset. We further implemented our model on noisy expected poses from subgroups of the Kinetics dataset and the UCF101 dataset. The experimental results also showed significant improvement using our proposed model.

Keywords: human action recognition (HAR); skeleton data; spatiotemporal; multi-task learning network (MTLN); convolutional neural network (CNN)

Citation: Mehmood, F.; Chen, E.; Akbar, M.A.; Alsanad, A.A. Human Action Recognition of Spatio-Temporal for Skeleton Sequences Using MTLN Feature Learning Framework. *Electronics* **2021**, *10*, 2708. https://doi.org/10.3390/electronics10212708

Academic Editors: Giovanni Dimauro and George A. Papakostas

Received: 20 August 2021
Accepted: 1 November 2021
Published: 5 November 2021

1. Introduction

At present, human action recognition (HAR) is an excellent area for research in computer vision. Three-dimensional skeleton data analysis of the path of human skeleton joints is less susceptible to brightness variations and never changes camera views [1]. Action recognition (AR) based on 3D skeleton sequences is gaining increased interest due to the popularity of more accurate and cheap devices [2–4]. The main-focus of our research is skeleton-based 3D pose recognition.

Success has been achieved in image classification [5,6]. However, in video action recognition, understanding the human poses to obtain temporal information on the sequence requests is broken into points [7–11], whereas the 3D shape of the human skeleton is also a crucial signal for action recognition [12]. The skeleton sequence is used to manage the joints of the human skeleton. To investigate the spatial and temporal characteristics of human interaction, studies have utilized recurrent neural networks (RNNs) with long short-term memory (LSTM) [13,14] neurons for joints of the skeleton sequence [2,15–17].

With respect to the long-term temporal dependency issues, LSTM networks are implemented to remember the information of the entire sequences across various periods; however, they still face some difficulties [18]. Moreover, it is tough to develop a deep LSTM for meeting unexpected features [19]. Currently, CNNs [20] have the capability to perfect long-term temporal dependency of a complete video [21]. In this research, we first directly determine the long-term sequential data from the skeleton structures as a frame. The CNN edges are used to efficiently access the long-term temporal shape from the skeleton sequence. Additionally, the human skeleton is viewed as an entire joint of pictures.

All proposed approaches show better performance in HAR. On the other hand, these approaches ignore the shape of humans and the connection between skeleton joints. Hence, we discuss the two main issues with this technique. Firstly, in the proposed model, the spatial relationship between distant joints can be established; in the convolutional layer, the relationship between two joints is treated equally according to the operational tool of networks [22], whereas the reality of the situation is that joint actions occur individually. The skeleton joints represent standard structural information. Each part of the human body performs a different activity. Thus, we should complete a study of humans' internal and external poses in the process of recognition. This study assists practitioners and academic researchers by executing the proposed model on the NTU RGB+D dataset, SYSU dataset, and SBU Kinetic Interaction dataset.

We propose an action recognition method using human skeleton information. Implementing this method, we extract the critical information when inputting the skeleton data for action recognition to minimize the extra loss of illumination. Hence, human skeleton information is robust, with no lighting effect. Subsequently, our spatiotemporal parameter-based model is implemented to perform action recognition.

The fundamental contributions of this work are summed up as follows:

1. We generate points for frames from various human actions as input that originate from datasets.
2. We develop a new deep CNN network model to transform each skeleton sequence. By learning from the hierarchical structure of deep CNNs, we enable long-term temporal modeling of the skeleton structure for frame images.
3. We create an MTLN to pick up the skeleton structure's spatial configuration and material factors to produce the CNN frames.
4. Our experimental results prove that MTLN achieves improvement compared to concatenating or pooling the features of the frames.
5. For better and efficient results, we implement AlexNet in the proposed network model. The final results also show the significance of our methodology.
6. Standard datasets are used to establish the presence of the proposed model. Some classic procedures are implemented for comparison.

The remainder of the article is arranged as follows: in Section 2, we introduce related work; in Section 3, we describe our network model in detail; Section 4 represents the datasets used in our approach, their implementation, and a discussion of the results; lastly, the conclusion is presented in Section 5.

2. Related Work

Aspects of computer vision focus on the skeleton base sequencing of data. Three-dimensional skeleton-based data sequences are more compact and robust than traditional image-based data in extracting the more essential 3D elements and removing noise. Many studies have recently focused on skeleton bases and action recognition approaches using either hand-crafted or deep learning techniques. These algorithms are schemes for extracting important features and sequences from raw data [23–25].

Deep learning techniques learn features in an end-to-end manner from data in the form of an array using machines. There are two aspects of the learning function: intra-

frame and inter-frame. Intra-frames present mutual coincidences, whereas inter-frames indicate the skeleton's temporal estimation.

Ke et al. [26] presented a technique for obtaining skeleton data and elementary knowledge to isolate 3D space records and sort grayscale images according to their dimensions. As mentioned, all implemented methods transfer the joint orders of data to create a collective representation. This technique extracts global coincidences from skeleton data, which enables a better evaluation of local coincidences.

For converting the frame-level feature, ConvNet can be used to learn from temporal data, while adopting LSTM; these approaches are practical and well-proven [27,28]. For expressing temporal evolution [29,30], optical flow represents another approach. Moreover, C3D [31] allows sequentially extracting features from spatial and temporal data using 3D convolutional layers.

Recurrent neural networks (RNNs) [32] were commonly implemented in previous studies on skeleton-based human action recognition for the temporal arrangement of skeleton sequences. Hierarchical construction [2] signifies spatial associations between body parts. On the other hand, the authors of [16,33,34] recommended a system for simultaneously learning spatial and temporal information, i.e., 2D LSTM. A two-stream RNN structure for spatiotemporal evolutions was proposed in [35]. Neural networks enable the user to determine the outlook position and obtain significant outcomes. Other approaches include neighbor searches [36] and Lie groups [37]; graphical neural networks [38] were also found to perform well with skeleton-based recognition data. Table 1 summarizes the performance of the above methods on two commonly utilized datasets: NTU RGB+D [38] and SBU Kinect Interaction [39].

Ke et al. [26] suggested modifying human skeleton joint images into grayscale images. In the proposed research, we further develop skeleton images using the complexity of the MTLN network. Table 1 shows that CNN-based methods have displayed excellent progress in learning skeleton images associated with RNN techniques.

Table 1. Performance of skeleton-based action recognition models in the literature.

Methods	Approach	NTU RGB+D Cross Subject	SBU Kinect Interaction
Two-stream RNN [35]	RNN	71.3%	94.8%
Ensemble TS-LSTM [32]	RNN	76.0%	-
Clips + CNN + MTLN [26]	CNN	79.6%	93.6%
GLAN [40]	CNN	80.1%	95.6%
A2GNN [37]	Graphic NN	72.7%	-
ST-GCN [38]	Graphic NN	81.5%	-

In skeleton-based action recognition, features created on the movements of joints are primarily used in the experiment. The combined program is divided into two elements: spatial movement and temporal movement [1]. The spatial actions which code the construction information of the human frame are designed between all couples of joints [1,36] or between some position joints and the new joints [3,41]. The temporal displacements that describe the actions of joints are recorded between motion frames [42]. Coordination is another generally used feature to describe the spatial position. The authors of [35] proposed a two-stream RNN to obtain both temporal dynamics and spatial structures. The RNN model is appropriate for material data; however, it lacks the ability to extract features from spatial data.

No specific model exists to address spatial and temporal dependency problems; however, scholars have extracted features from skeleton information using the CNN model and obtained excellent results [43]. CNN is successfully used for both images and videos [44]. Previous studies on skeleton-based action recognition have shown that CNNs can extract features from the temporal relationships of the combined sequences. Moreover,

CNN-based models perform better in terms of intra-frame relationships than RNN-based models. However, CNNs cannot extract features from all joints. Accordingly, we proposed determining the spatial connection through significant action recognition based on skeleton points.

3. Proposed Methodology

This study had three aims: (1) to arrange the joints in every frame to interconnect body parts; (2) to create parallel vectors from joints and skeleton sequences; (3) to use all these data as input for training the neural network. MTLN was used for action recognition in the proposed model architecture.

3.1. Implementation Details

We used an eighth Generation Core i7 Quad-Core Processor with 8 GB RAM, 8 GB NVIDIA, and a 520 GB SSD hard drive for our research. Clips were created from all frames of the original skeleton sequence without preprocessing (i.e., normalization, temporal down sampling, or noise filtering) for all datasets. As an initial step, the first layer's remote units were set to 512. The number of teams in the second layer (i.e., the output layer) was 256. Using the RMSprop algorithm with a momentum value of 0.92, the network was trained. The learning rate was set to 0.001 for 1000 steps, then to 0.0001 after 1000 steps, with a batch size of 16 degrees. The training stopped after 32 epochs. In the activity of our model, 32,000 steps were taken. The performance of the suggested method on each dataset was compared to existing approaches using the same testing procedure. Training took 12 h.

3.2. Frame Process

As shown in Figure 1, every frame in the skeleton sequence is related to all other structures. The first step was to establish a chain of joints from every frame to interconnect each joint. Thus, two-dimensional arrays were created by comparing all edges of the skeleton sequence. In this article, we converted the two-dimensional network structures of the four matches into a CNN-based model, generating four frames based on the four distant positions of the skeleton.

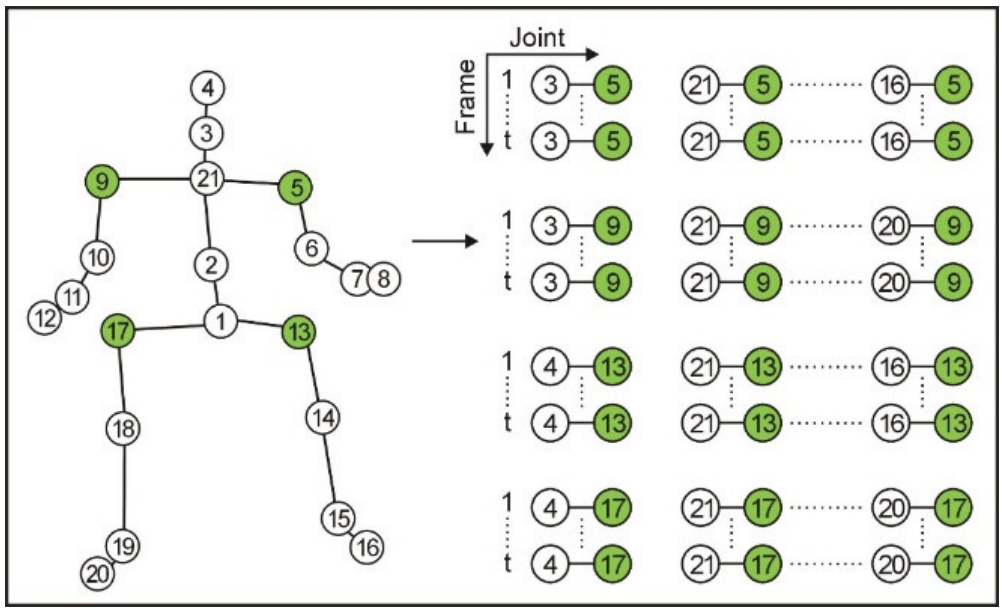

Figure 1. The skeletal sequence created in frames. To begin, the joints of each body part are linked to form a chain of skeleton joints for each frame. (i.e., 1–2–3–…–21) Green points represent the four reference points (i.e., right shoulder 5, left shoulder 9, right hip 13, and left hip 17), which are used to calculate the relative positions of other joints to combine the dissimilar spatial relations between

joints. Therefore, all frames of the skeleton are joined at common positions to construct the four 2D arrays according to a cylindrical vertex. The points on the 2D array are transformed into parallel gray images.

In the initial deep CNN data extraction stage, the skeleton sequence of long-term temporal knowledge pulls solid symbols from each frame to represent a distant spatial connection. Using multi-task learning, the CNN properties of all frames are handled in a coordinated manner, allowing them to determine 3D spatiotemporal information for action detection.

3.3. CNN Training for Feature Extraction

Each frame defines the temporal dynamics of the skeleton sequence, and the spatial-constant CNN characteristic of each frame provides strong knowledge about the skeleton sequence. Expert CNN models help predict attributes since they are executed with models trained using Image Net [19]. Using a pretrained CNN feature, the CNN features of each frame can be removed using AlexNet [45]. They can be helpful in several cross-domain applications [46,47]. The frame creation and execution process for skeleton sequence is presented in Figure 2.

Moreover, recent skeleton datasets were too small or too noisy to allow effective evaluation of a deep system. Even though the frames were not original images, the edges were inserted into the pretrained CNN model using AlexNet [45]. Realistic photos and the generated frames shared the same structure because they were matrices with patterns. CNN models efficiently extract feature information from enormous image datasets, which enables the user to identify the elements of designs in a group.

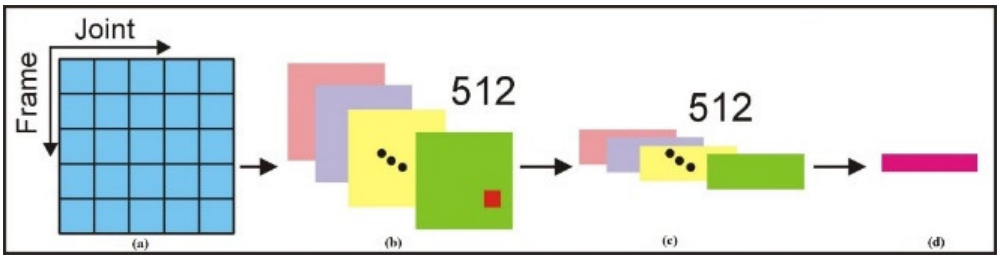

Figure 2. (**a**) An input frame created from a skeleton sequence, with columns parallel to the different vectors created from the joints and the rows, similar to the various structures of the skeleton sequence, for which the columns are parallel to the multiple vectors created from the joints and the rows are parallel to the different frames of the skeleton sequence. (**b**) Output feature maps of the conv8_1 layer (size: 28 × 28 × 512). (**c**) All joints in the skeleton sequence have temporal features, which are obtained by applying mean pooling to each feature map in the row. (**d**) The output feature is obtained by integrating all feature maps in the dataset (**c**).

In the proposed pretrained AlexNet [45], there were eight groups of convolutional layers from conv1 to conv8. Every set of convolutional layers contained four stacks of a layer, each of which had a similar kernel size. There were two fully connected (FC) layers and 32 convolutional layers in the model. In this way, deep neural networks can extract powerful features from different frames and use them in another field. Each layer has its way of pulling elements from another frame. The feature extraction process primarily depends on the new classes. Early layers have greater potential to be transferred to another domain [48]. As a result, this study, which was based on convolutional layer stimulations, aimed to establish the material factors of skeleton sequences.

The convolutional layer feature map enables the user to achieve action recognition and picture recognition [49,50]. In this study, we discarded the FC layers and the last three convolutional layers of the network. Each frame image of the four frames was adjusted to 224 × 224 and embedded into the model. The CNN outcomes were inserted into the temporal mean pooling (TMP) layer as the input frame. The frame's dimensions were 28 × 28

× 512, i.e., 512 feature maps of size 28 × 28; the created frame's rows were parallel to alternative edges in a skeleton sequence. The skeleton sequence was represented by the movements of the row features of the resulting image. Meanwhile, the stimulations of each feature map on the conv8_1 layer corresponded to the regional locations in the original input image [49].

Feature maps were used to extract the temporal information of the skeleton sequence from the row features. In detail, the feature maps were combined with TMP using a kernel size of 28 × 1, i.e., the temporal function was performed under pooling. The activation of the *k*-th feature map's eighth row and *j*-th column can be represented as *xki*, *j*.

Then, the output of the *k*-th feature map after TMP is as follows:

$$Y^{k} = \left(y^{k}_{1}, \ldots, y^{k}_{j}, \ldots, y^{k}_{28} \right)$$
$$y^{k}_{j} = \frac{1}{28} \sum_{i=1}^{28} \max\left(0, x^{k}_{i,j}\right) \quad (1)$$

All feature maps (512) are combined to form a 14,336-D (28 × 512 = 14,336) feature vector, which signifies the temporal dynamics of the skeleton sequence.

3.4. Multi-Task Learning Network

Individually, temporal dynamic information is proposed from skeleton sequence vectors and contains one specific spatial association between the joints. Four-component vectors are intrinsically related to one another. Meanwhile, the information obtained from the skeleton sequence is fed into frames that provide temporal information. Then, the features can be extracted from frames to determine the long-term temporal structure of the skeleton sequence. Each frame generated from different images is inserted into a deep CNN to obtain the CNN features. In Figure 3e, the three 14,336-D properties of the four frames in an equal period are combined to procedure a feature vector; overall, four feature vectors were generated. Then, the four CNN features of the four frames (See Figure 3) were combined into one feature vector at a time. Thus, we extracted four feature vectors from all feature vectors indicating the skeleton sequence's temporal information and one specific joint spatial connection. The feature vectors describe distant spatial relationships with core connections between joints. This research recommends applying intrinsic connections among different feature vectors for action recognition with MTLN. The simplification of multiple tasks demonstrates an equal computation of numerous correlated studies utilizing the core connections.

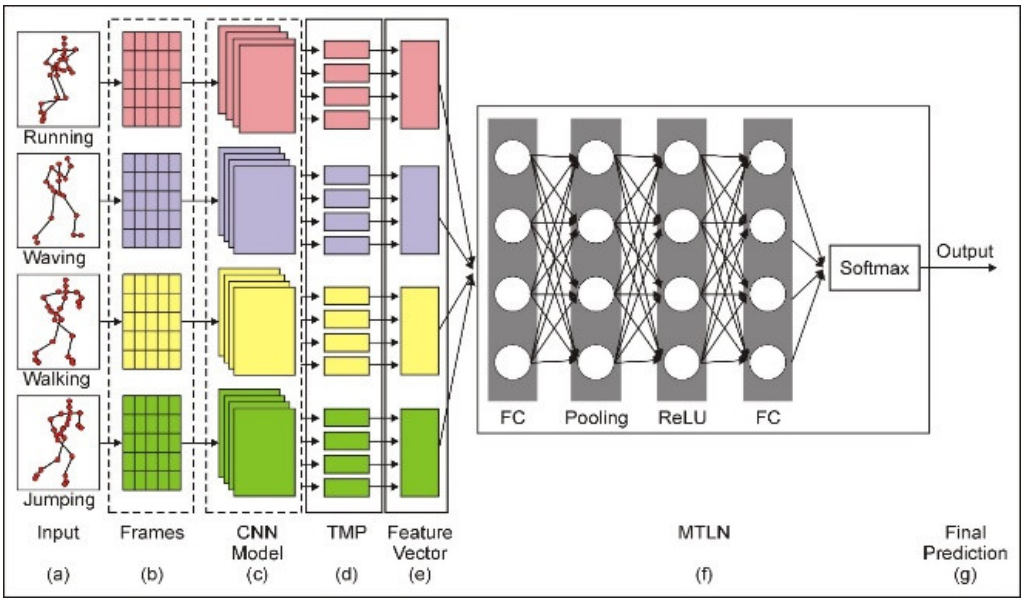

Figure 3. Developed model architecture. (**a**) Skeleton sequence; (**b**) frame generated from the given sequence; (**c**) framing joint points inserted into a deep CNN model; (**d**) TMP (temporal mean pooling) layer used to access the information from frames; (**e**) CNN output of four frames at equal time phases, where feature vectors are used to get the results, and each vector represents the skeleton sequence's timing and spatial relation; (**f**) the developed MTLN containing a ReLU (rectified linear unit), a max-pooling layer, an FC (fully connected) layer, a Softmax layer, and another FC layer; (**g**) final output for testing and training data.

Using MTLN, each feature vector's arrangement was managed as a distinct task. The MTLN was trained from various inputs from one feature vector and obtained multiple outcomes as the final prediction. Then, four features were extracted from a sequence of feature vectors corresponding to the temporal information in the skeleton and one specific joint's spatial connection. The features describe distant spatial relationships with core connections between joints. MTLN explains various tasks as a function of weight, which enhances the presentation of specific tasks [22]. MTLN provided a standard process through which four-component vectors could be used for action recognition utilizing their intrinsic associations. Each component vector's organization was handled as another task with a similar grouping mark of the skeleton sequences.

3.5. Architecture of Network

The construction of our developed network is presented in Figure 3f. The deep CNN network model contained four frames as different input joints of additional skeleton images, one max-pooling layer, one rectified linear unit (ReLU) [44] to present extra nonlinearity between two fully connected (FC) layers, and the output using a Softmax layer. Using the four features as information sources, the MTLN produces four edge-level forecasts compared to one assignment.

In detail, for each skeleton sequence, we created four parallel frames for the CNN layers. Frames were derived from the relative joints of four discussion points. Each frame contained the historical elements of the entire skeleton sequence and a specific spatial link between the joints. Each skeleton point was associated with multiple frames of data with different spatial associations, thereby providing critical knowledge at different spatial angles.

From one point of view, the impact of information on skeleton sequences is crucial because of the disappearance of various sources of noise such as the background and the naturally structured 3D joint position data. As a second point of view, image-based datasets are more significant than point-based sequences, allowing deeper examination of information, particularly for preparing deep learning models. Furthermore, because of the

development of Kinect and robust depth sensors, as well as the increasing number of methods to approximate joint positions, it is essential to access skeleton-based data.

The four-loss values were summed to calculate the system's loss value, used to update the system factors. A final class prediction was generated by averaging results from the four task classes. While training, to calculate the loss value of each task, the class scores were utilized. Thus, all functions were entirely lost such that the last loss of the structure was used. During the testing process, the scores of all tasks were averaged to determine the previous forecast of the action class. Equation (2) indicates the loss of the *k*-th task (*k* = 1, …, 4).

$$L_k(x_k, y) = \sum_{i=1}^{n} y_i \left[-\log \left(\frac{Exp.x_{ki}}{\sum_{j=1}^{n} Exp.x_{kj}} \right) \right], \quad L_k(x_k, y) = \sum_{i=1}^{n} y_i \left[\log \left(\sum_{j=1}^{n} Exp.x_{kj} \right) - x_{ki} \right] \tag{2}$$

where x_k is the vector inserted into the Softma$x$ layer generated by the *k*-th input feature, *n* is the number of action classes, and y_i is the ground-truth label for class I. Equation (3) can be used to get the network's final loss value as the sum of four specific losses.

$$L(X, y) = \sum_{k=1}^{4} l_k(x_k, y), \tag{3}$$

where $X = [x1, \ldots, x4]$.

4. Experiments

Datasets

To evaluate the performance of our developed model, our model was implemented on five different well-known datasets: NTU RGB+D (NTU) [16], SBU Kinect Interaction [39], Kinetics [45], SYSU-3D (SYSU) [51], and UCF101 [24].

1. The NTU RGB+D dataset [27] is currently the largest dataset for action recognition with more than 56,000 sequences and four million frames. The applied dataset has 50 different human poses and 70 class actions with 40 regular action classes. Cross-subject (CS) and cross-view (CV) are the two suggested elevation protocols. In general, at the beginning of our research, we followed the settings of [27]. In the CS evaluation, 40,500 samples from 40 subjects were used for training a planned model, using the other 18,540 examples for analysis. For the cross-view evaluation, 38,700 samples were taken from the second camera, while the third camera was used for training the model and analyzing the other 18,600 samples from camera 1.
2. The SBU Kinect Interaction dataset [39] contains 280 skeleton sequences and 6810 frames. We followed the regular research protocol of fivefold cross-validation with delivered splits, yielding eight classes. In each skeleton, frames had two people, and 15 joints were labeled for each person. While training, two samples were used for two skeleton sequences. While testing, the average forecast score was calculated. During the training process, random collection was used to augment data. Five recent crops were taken for prediction scores, and four corners were averaged for the testing calculation.
3. Kinetics-Motion [45], the most significant RGB action recognition dataset, has 400 action classes, including three lac video clips. Videos were downloaded from YouTube, and each clip has a 10 s duration. Yan et al. [38] provided estimated poses for action recognition based on joints. In the first step, videos were resized to 340,256 pixel resolution at 30 frames per second. Furthermore, an OpenPose toolbox was used Hu, et

al., 2016 which cannot separate the particular classes of RGB action recognition datasets because it has no background context and image presence. Yan et al. [38] proposed a Kinetics-Motion dataset, a 30-class subset of Kinetics with action labels associated with body movement. We followed the suggested process on the Kinetics-Motion dataset. The 30 classes were skateboarding, tai chi hopscotch, pull-ups, capoeira, punching bag, squat, deadlifting, clean and jerk, push up, punching bag, belly dancing, country line dancing, surfing crowd, swimming backstroke, front raises, crawling baby, windsurfing, skipping rope, throwing discus, snatch weight lifting tobogganing, hitting a baseball, roller skating, arm wrestling, riding a mechanical bull, salsa dancing, hurling (sport), lunge, hammer throw, and juggling balls.

4. The SYSU-3D dataset [51] contains 480 sequences and 12 distant actions executed by 40 people. Twenty joints from each frame of the series were connected with 3D coordinates. We set a 0.2 ration of training and validation datasets for the random split of the data, the same ration is used by existing studies [52,53].
5. UCF101-Motion [24] contains 13,300 videos from 100 action classes fixed at 320 × 240 pixel resolution at 25 FPS. At the same time, as input RGB videos, approximately 16 joint actions were taken using the AlphaPose toolbox. Similarly to Kinetics-Motion, in UCF101, predefined actions such as "cutting in the kitchen" are more closely related to items and actions. To verify this, we followed the method in ST-GCN [38] and established a subset of UCF-101 named "UCF-Motion". UCF-Motion has 24 classes associated with the poses in a total of 3170 videos: jump rope, playing the piano, crawling baby, playing the flute, playing the cello, punch, tai chi, boxing speed bag, pushups, juggling balls, golf swing, clean and jerk, playing the guitar, bowling, ice dancing, playing soccer juggling, playing dhol, the tabla, boxing punching bag, salsa spins, hammer throw, rafting, and writing on board.

5. Experiment Results and Analysis

5.1. Data Preprocessing

1. All frames were created from the original skeleton sequence with preprocessing steps such as temporal downsampling, noise filtering, and normalization.
2. In the first FC, the layer had 512 hidden units. In the second FC layer (i.e., the output layer), the action classes in each dataset and the number of the units were identical. The network was trained using the stochastic gradient descent algorithm at a training rate of 0.002 with 200 batch sizes and 50 epochs used in preparing the model.
3. The performance of the developed model on each dataset was compared with existing methods.

5.2. Classification Report

The datasets used in this article included five different types of human actions NTU RGB+D, SYSU, SBU Kinetic Interaction, Kinetics, and UCF101. We applied MTLN to these datasets and obtained the accuracy, recall, and precision, yielding an average of 90% in each case (Table 2).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}.$$

$$\text{Precision} = \frac{TP}{TP+FP}.$$

$$\text{Recall} = \frac{TP}{TP+FN}.$$

$$\text{F1 Score} = \frac{2*Precision*Recall}{Precision+Recall}.$$

Table 2. Precision, recall, and F1 score for all classes, labeled 0–3.

Class	Class Level	Precision	Recall	F1-Score	Accuracy (%)
Running	0	0.88	0.91	0.89	90
Waving	1	0.92	0.85	0.88	90
Walking	2	0.88	0.88	0.88	90
Jumping	3	0.93	0.95	0.95	90

5.3. Confusion Matrix

We established a confusion matrix as shown in Figure 4. We trained our model with 100 different videos of running, waving, walking, and jumping. It is clear from the matrix that the running class had 21 accurate predictions for running, no prediction for waving, two wrong predictions for walking because the model considered running and walking identical, and no wrong prediction for jumping. The waving class had 22 accurate predictions with one wrong prediction each for running and walking and two wrong predictions for jumping. The walking class had 21 accurate predictions, with one and two wrong predictions for waving and running, respectively, and no wrong prediction for jumping. The jumping class was the most accurate with 26 correct predictions and only one wrong prediction for waving. It is clear from the confusion matrix that our model was very accurate for the most part.

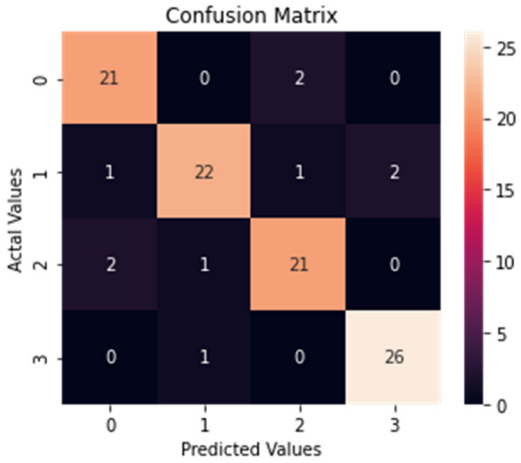

Figure 4. Confusion matrix of our model.

5.4. Comparison with Different Models for Action Recognition

Table 3 shows the accuracy rate of our compared models, showing gradual improvement in final accuracy for our model. We can conclude that the proposed technique better determines the correlation information between different joints in HAR.

Table 3. NTU RGB+D dataset showing an accuracy (%) comparison with previously proposed methods for cross-view (CV) and cross-subject (CS) evaluation.

Methods	Year	CV	CS
Trust Date ST-LSTM [15]	2016	77.7	69.2
Visualization CNN [54]	2017	82.6	76.0
VA-LSTM [55]	2017	87.7	79.2
STA-LSTM [56]	2018	81.2	73.4
ST-GCN [38]	2018	88.3	81.5
FDNet [57]	2018	89.8	83.5
DTIs [57]	2019	-	85.4
HCN [26]	2018	91.1	86.5

DPI + att-DTIs [58]	2019	-	90.2
GL LSTM + Diff [59]	2020	90.2	84.4
Our model	-	91.4	91.3

The accuracy of cross-view (CV) and cross-subject (CS) evaluation is clearly shown in Figure 5.

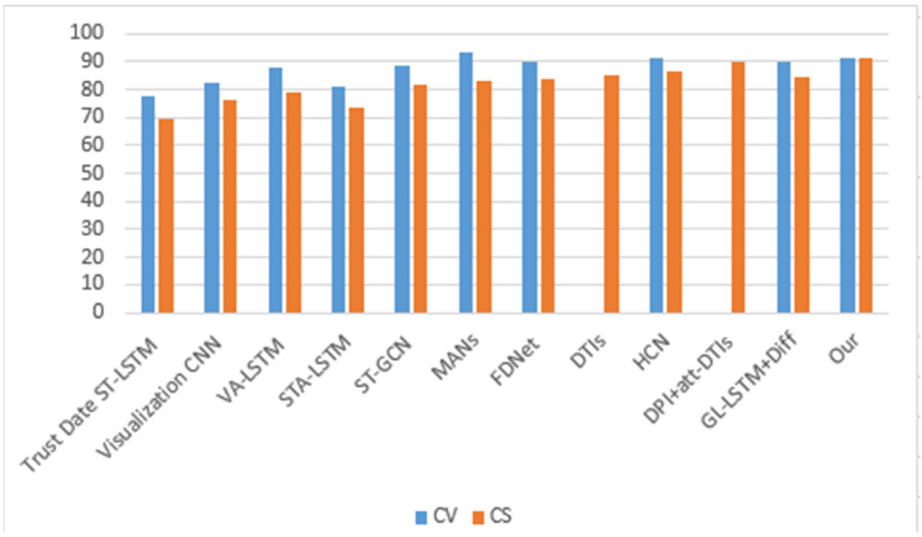

Figure 5. Accuracy comparisons for the cross-view (CV) and cross-subject (CS) evaluation for different models using the NTU RGB + D dataset.

The accuracy of cross-subject (CS) evaluation for models proposed in the last 5 years is shown in Figure 6.

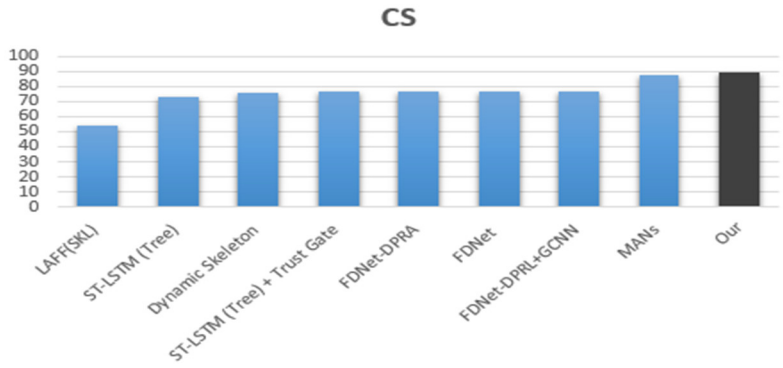

Figure 6. Accuracy comparisons for cross-subject (CS) evaluation for different methods using the SYSU dataset.

5.5. Comparison with State-of-the-Art Methods

We conducted a systematic evaluation of the various models using the datasets mentioned above. The proposed approach showed improved accuracy (Tables 3–7).

1. The NTU RGB+D dataset (Table 3) is currently the most extensive dataset; the proposed model achieved greater recognition accuracy than previous approaches, showing a 0.3% enhancement compared to previous CNN-based models for cross-view evaluation. We compared our model-generated accuracy with the latest DPI + att-DTIs-based model, showing a 1.1% improvement (Table 3).
2. Using the SUSU-3D dataset (Table 4), our proposed model improved the recognition accuracy compared to previous models by a considerable margin, e.g., 1.6%

compared with the well-known MANs [55] model. This demonstrates that our suggested model can thoroughly explore the characteristics of skeleton-based sequences to complete the action recognition test at a high level.

3. For the SBU Kinect Interaction, the MTLN approach exceeded the performance of other studies in the literature in terms of recognition accuracy, similarly to the NTU RGB+D dataset. Our model achieved 95.4% ± 1.7% accuracy across the five splits in the case of SBU Kinect Interaction presented in Table 5 and Figure 7.
4. For Kinetics-Motion (Table 6), the previously developed models showed inferior performance to our proposed model using MTLN [60]. The accuracy of recognition was also comparable to systems using other modalities such as RGB and optical flow. The proposed model was robust toward noise, whereas deficiencies were commonly produced due to missing or improper pose estimates (Figure 8).
5. For UCF101-Motion (Table 7), the proposed approach outperformed existing algorithms that only used one modality [28,31] or both the presence feature and optical flow [61]. This experiment demonstrates that joints are a natural modality for identifying motion associated with actions, but that joints alone cannot distinguish all unique action classes (Figure 9). Knowing specific categories requires an object and part appearances, whereas incorrect pose estimation decreases the limit of recognition accuracy (Figure 9). Performance can be improved using our model.

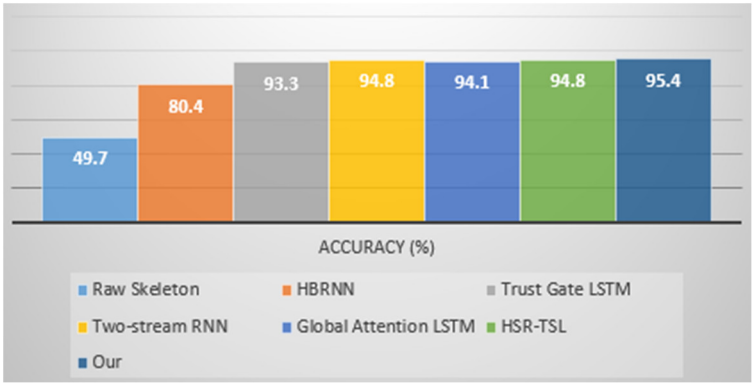

Figure 7. Accuracy comparison with other methods using the SUB Kinetic Interaction dataset.

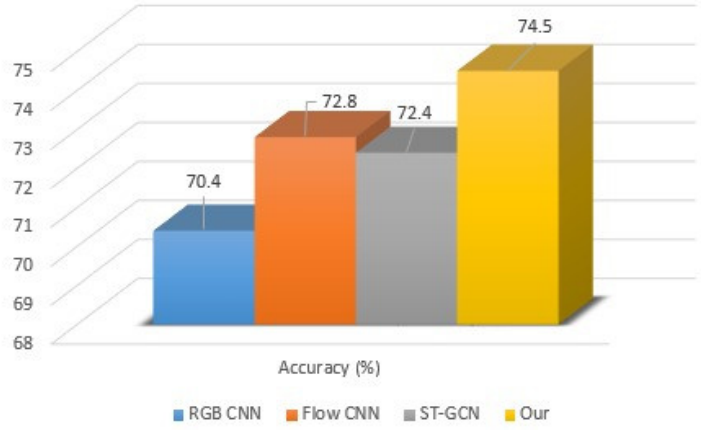

Figure 8. Accuracy comparison with other methods using the Kinetics-Motion dataset.

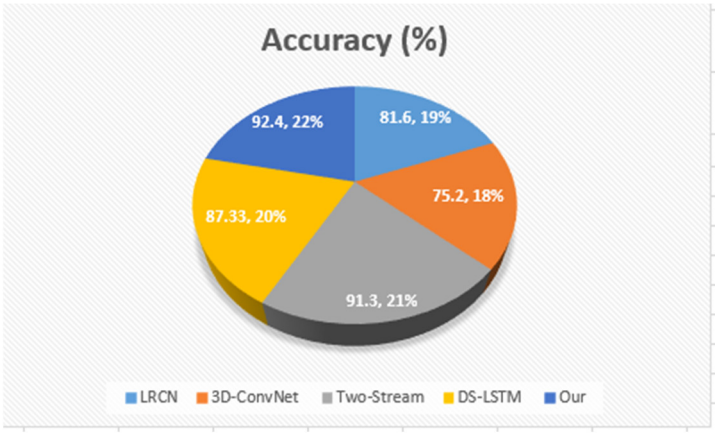

Figure 9. Accuracy comparison with other methods using the UCF-Motion dataset.

Table 4. Accuracy (%) achieved with previous models for cross-subject (CS) evaluation using SYSU dataset.

Methods	Year	CS
LAFF(SKL) [61]	2016	54.2
ST-LSTM (Tree) [62]	2017	73.4
Dynamic Skeleton [50]	2015	75.5
ST-LSTM (Tree) + Trust Gate [62]	2017	76.5
FDNet-DPRA [56]	2018	76.7
FDNet [56]	2018	76.9
FDNet DPRL + GCNN [63]	2018	76.9
MANs [55]	2018	87.6
Our model	-	89.2

Table 5. Recognition accuracy (%) achieved with previous methods using SUB Kinetic Interaction dataset.

Methods	Accuracy (%)
Raw Skeleton [39]	49.7
HBRNN [2]	80.4
Trust Gate LSTM [5]	93.3
Two-stream RNN [35]	94.8
Global Attention LSTM [33]	94.1
HSR-TSL [64]	94.8
Our model	95.4

Table 6. Comparison of accuracy (%) with previous methods using Kinetics-Motion dataset.

Methods	Accuracy (%)
RGB CNN [45]	70.4
Flow CNN [45]	72.8
ST-GCN [38]	72.4
Our model	74.5

Table 7. Comparison of action recognition accuracy (%) with previous methods using UCF-Motion dataset.

Methods	Accuracy (%)	RGB	Key Points
LRCN [28]	81.6	✓	-
3D-ConvNet [31]	75.2	✓	-

Two-Stream [65]	91.3	✓	-
DS-LSTM [66]	87.33	✓	✓
Our model	92.4	✓	✓

6. Conclusions

This article developed a CNN network for feature learning and action recognition to insert a skeleton sequence into the model. The experiment showed that joints are a valid modality for detecting motion associated with actions, but that joints alone cannot distinguish all unique action classes. Understanding a specific type requires knowledge of the object and the parts. Four-frame CNNs were interconnected into a single matrix, while the skeleton sequence was described temporally using a particular spatial connection between joints. Furthermore, we implemented MTLN to mutually learn the feature vectors at a similar phase in parallel, which improved the performance in HAR via these critical connections. We analyzed our model using five different datasets: NTU RGB+D dataset, SBU Kinect Interaction, Kinetics-Motion, SYSU-3D dataset, and UCF101-Motion. The experimental results showed the effectiveness of our newly developed model and the feature learning method.

Furthermore, we contribute to the field by improving the generalizability of our model with respect to previous studies. This feature needs to be tested on a more significant number of sequences to ensure its robustness. It currently recognizes only one action in a series. We intend to address this issue in the next phase of our implementation and expand our system to identify more complex activities and recognize sequences involving combinations of actions.

Author Contributions: Methodology and Formal analysis, F.M.; Project administration, E.C.; Data curation, A.A.A. and Software, M.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grants U1804152,62101503 and 61806180.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Han, F.; Reily, B.; Hoff, W.; Zhang, H. Space-time representation of people based on 3D skeletal data: A review. *Comput. Vis. Image Underst.* **2017**, *158*, 85–105.
2. Du, Y.; Wang, W.; Wang, L. Hierarchical recurrent neural network for skeleton-based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1110–1118.
3. Ke, Q.; An, S.; Bennamoun, M.; Sohel, F.; Boussaid, F. Skeletonnet: Mining deep part features for 3-d action recognition. *IEEE Signal Process. Lett.* **2017**, *24*, 731–735.
4. Wang, P.; Li, Z.; Hou, Y.; Li, W. Action recognition based on joint trajectory maps using convolutional neural networks. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 102–106.
5. Cregan, D.; Meier, U.; Schmidhuber, J. Multi-column deep neural networks for image classification. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3642–3649.
6. Ke, Q.; Li, Y. Is rotation a nuisance in shape recognition? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 4146–4153.
7. Fernando, B.; Gavves, E.; Oramas, J.M.; Ghodrati, A.; Tuytelaars, T. Modeling video evolution for action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5378–5387.
8. Gaidon, A.; Harchaoui, Z.; Schmid, C. Temporal localization of actions with actions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2782–2795.
9. Ke, Q.; Bennamoun, M.; An, S.; Boussaid, F.; Sohel, F. Human interaction prediction using deep temporal features. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 403–414.
10. Niebles, J.C.; Chen, C.-W.; Fei-Fei, L. Modeling temporal structure of decomposable motion segments for activity classification. In Proceedings of the European Conference on Computer Vision, Crete, Greece, 5–11 September 2010; pp. 392–405.

11. Wang, L.; Qiao, Y.; Tang, X. Latent hierarchical model of temporal structure for complex activity classification. *IEEE Trans. Image Process.* **2013**, *23*, 810–822.
12. Zhu, W.; Lan, C.; Xing, J.; Zeng, W.; Li, Y.; Xie, X. Co-occurrence feature learning for skeleton-based action recognition using regularized deep LSTM networks. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February, 2016.
13. Graves, A. Supervised sequence labeling. In *Supervised Sequence Labeling with Recurrent Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 5–13.
14. Graves, A.; Mohamed, A.-r.; Hinton, G. Speech recognition with deep recurrent neural networks. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 6645–6649.
15. Liu, J.; Shahroudy, A.; Xu, D.; Wang, G. Spatio-temporal lstm with trust gates for 3d human action recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016, pp. 816-833.
16. Shahroudy, A.; Liu, J.; Ng, T.-T.; Wang, G. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1010–1019.
17. Veeriah, V.; Zhuang, N.; Qi, G.-J. Differential recurrent neural networks for action recognition. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4041–4049.
18. Gu, J.; Wang, G.; Cai, J.; Chen, T. An empirical study of language CNN for image captioning. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1222–1231.
19. Sainath, T.N.; Vinyals, O.; Senior, A.; Sak, H. Convolutional, long short-term memory, fully connected deep neural networks. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Brisbane, Australia, 19–24 April 2015; pp. 4580–4584.
20. LeCun, Y.; Bengio, Y. Convolutional networks for images, speech, and time series. In *The handbook of Brain Theory and Neural Networks*; MIT Press: Cambridge UK, 1995; Volume 3361, p. 1995.
21. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal segment networks: Towards good practices for deep action recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–12 October 2016; pp. 20–36.
22. Caruana, R. Multitask learning. In *Learning to Learn*; Springer: Berlin/Heidelberg, Germany, 1998.
23. Gowayyed, M.A.; Torki, M.; Hussein, M.E.; El-Saban, M. Histogram of oriented displacements (HOD): Describing trajectories of human joints for action recognition. In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, Beijing, China, 3–9 August 2013.
24. Li, W.; Zhang, Z.; Liu, Z. Action recognition based on a bag of 3d points. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, San Francisco, CA, USA, 13–18 June 2010; pp. 9–14.
25. Vemulapalli, R.; Arrate, F.; Chellappa, R. Human action recognition by representing 3d skeletons as points in a lie group. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 588–595.
26. Ke, Q.; Bennamoun, M.; An, S.; Sohel, F.; Boussaid, F. A new representation of skeleton sequences for 3d action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3288–3297.
27. Ng, J.Yu.; Hausknecht, M.; Vijayanarasimhan, S.; Vinyals, O.; Monga, R.; Toderici, G. Beyond short snippets: Deep networks for video classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4694–4702.
28. Donahue, J.; Hendricks, L.A.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Guadarrama, S.; Saenko, K.; Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2625–2634.
29. Feichtenhofer, C.; Pinz, A.; Zisserman, A. Convolutional two-stream network fusion for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1933–1941.
30. Carreira, J.; Zisserman, A. Quo Vadis, action recognition? a new model and the kinetics dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
31. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
32. Lee, I.; Kim, D.; Kang, S.; Lee, S. Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1012–1020.
33. Liu, J.; Wang, G.; Hu, P.; Duan, L.-Y.; Kot, A.C. Global context-aware attention LSTM networks for 3D action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1647–1656.
34. Song, S.; Lan, C.; Xing, J.; Zeng, W.; Liu, J. An end-to-end Spatio-temporal attention model for human action recognition from skeleton data. In Proceedings of the Thirty-first AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.

35. Wang, H.; Wang, L. Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 499–508.
36. Weng, J.; Weng, C.; Yuan, J. Spatio-temporal naive-Bayes nearest-neighbor (st-nbnn) for skeleton-based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4171–4180.
37. Li, C.; Cui, Z.; Zheng, W.; Xu, C.; Ji, R.; Yang, J. Action-attending graphic neural network. *IEEE Trans. Image Process.* **2018**, *27*, 3657–3670.
38. Yan, S.; Xiong, Y.; Lin, D. Spatial-temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
39. Yun, K.; Honorio, J.; Chattopadhyay, D.; Berg, T.L.; Samaras, D. Two-person interaction detection using body-pose features and multiple instance learning. In Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012; pp. 28–35.
40. Yang, Z.; Li, Y.; Yang, J.; Luo, J. Action recognition with visual attention on skeleton images. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 3309–3314.
41. Wang, J.; Liu, Z.; Wu, Y.; Yuan, J. Learning action ensemble for 3D human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *36*, 914–927.
42. Yang, X.; Tian, Y. Effective 3d action recognition using eigenjoints. *J. Vis. Commun. Image Represent.* **2014**, *25*, 2–11.
43. Li, C.; Wang, P.; Wang, S.; Hou, Y.; Li, W. Skeleton-based action recognition using LSTM and CNN. In Proceedings of the 2017 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), Hong Kong, China, 10–14 July 2017; pp. 585–590.
44. Nair, V.; Hinton, G.E. Rectified linear units improve restricted Boltzmann machines. In Proceedings of the 27th International Conference on Machine Learning (ICML-10), Haifa, Israel, 21–24 June 2010; pp. 807–814.
45. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012, Lake Tahoe, NV, USA, 3–6 December 2012; p. 9.
46. Han, X.; Leung, T.; Jia, Y.; Sukthankar, R.; Berg, A.C. Matchett: Unifying feature and metric learning for patch-based matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3279–3286.
47. Razavian, A.S.; Azizpour, H.; Sullivan, J.; Carlsson, S. CNN features off-the-shelf: An astounding baseline for recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Columbus, OH, USA, 23–28 June 2014; pp. 806-813.
48. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 3320–3328.
49. Peng, X.; Schmid, C. Encoding Feature Maps of CNN for Action recognition. Hal-Inria, Faranc. Available online: https://hal.inria.fr/hal-01236843/ (accessed on 29 April 2017).
50. Radenović, F.; Tolias, G.; Chum, O. CNN image retrieval learn from BoW: Unsupervised fine-tuning with hard examples. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 3–20.
51. Hu, J.-F.; Zheng, W.-S.; Lai, J.; Zhang, J. Jointly learning heterogeneous features for RGB-D activity recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5344–5352.
52. Cao, Z.; Simon, T.; Wei, S.-E.; Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7291–7299.
53. Fang, H.-S.; Xie, S.; Tai, Y.-W.; Lu, C. Rmpe: Regional multi-person pose estimation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017, pp. 2334–2343.
54. Liu, M.; Liu, H.; Chen, C. Enhanced skeleton visualization for view-invariant human action recognition. *Pattern Recognit.* **2017**, *68*, 346–362.
55. Zhang, P.; Lan, C.; Xing, J.; Zeng, W.; Xue, J.; Zheng, N. View adaptive recurrent neural networks for high-performance human action recognition from skeleton data. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2117–2126.
56. Song, S.; Lan, C.; Xing, J.; Zeng, W.; Liu, J. Spatio-Temporal Attention-Based LSTM Networks for 3D Action Recognition and Detection. *IEEE Trans. Image Process.* **2018**, *27*, 3459–3471.
57. Tokuda, K.; Jadoulle, J.; Lambot, N.; Youssef, A.; Koji, Y.; Tadokoro, S. Dynamic robot programming by FDNet: Design of FDNet programming environment. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566), Sendai, Japan, 28 September–2 October 2004; pp. 780–785.
58. Liu, M.; Meng, F.; Chen, C.; Wu, S. Joint dynamic pose image and space-time reversal for human action recognition from videos. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 2–9 February 2019; pp. 8762–8769.
59. Han, Y.; Chung, S.L.; Xiao, Q.; Lin, W.Y.; Su, S.F. Global Spatio-Temporal Attention for Action Recognition Based on 3D Human Skeleton Data. *IEEE Access* **2020**, *8*, 88604–88616.

60. Zhang, P.; Lan, C.; Xing, J.; Zeng, W.; Xue, J.; Zheng, N. View Adaptive Neural Networks for High-Performance Skeleton-Based Human Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 1963–1978.
61. Hu, J.-F.; Zheng, W.-S.; Ma, L.; Wang, G.; Lai, J. Real-time RGB-D activity prediction by soft regression. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 280–296.
62. Liu, J.; Shahroudy, A.; Xu, D.; Kot, A.C.; Wang, G. Skeleton-based action recognition using Spatio-temporal lstm network with trust gates. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 3007–3021.
63. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 568–576.
64. Si, C.; Jing, Y.; Wang, W.; Wang, L.; Tan, T. Skeleton-based action recognition with hierarchical spatial reasoning and temporal stack learning network. *Pattern Recognit.* **2020**, *107*, 107511.
65. Zhou, K.; Wu, T.; Wang, C.; Wang, J.; Li, C. Skeleton Based Abnormal Behavior Recognition Using Spatio-Temporal Convolution and Attention-Based LSTM. *Procedia Comput. Sci.* **2020**, *174*, 424–432.
66. Jiang, X.; Xu, K.; Sun, T. Action Recognition Scheme Based on Skeleton Representation With DS-LSTM Network. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 2129–2140.