energies

MDPI

Article

# Manual Operation Evaluation Based on Vectorized Spatio-Temporal Graph Convolutional for Virtual Reality Training in Smart Grid

**Fangqiuzi He [1,2], Yong Liu [2], Weiwen Zhan [2], Qingjie Xu [2] and Xiaoling Chen [3,*]**

[1] School of Art and Design, Wuhan Polytechnic University, Wuhan 430023, China; fangqiuzihe@163.com
[2] School of Mechanical Engineering and Electronic Information, China University of Geosciences, Wuhan 430074, China; yongliu@cug.edu.cn (Y.L.); wwz645351492@gmail.com (W.Z.); cugwhly@163.com (Q.X.)
[3] School of Art and Media, China University of Geosciences, Wuhan 430074, China
* Correspondence: babyvslee1009@cug.edu.cn

**Abstract:** The standard of manual operation in smart grid, which require accurate manipulation, is high, especially in experimental, practice, and training systems based on virtual reality (VR). In the VR training system, data gloves are often used to obtain the accurate dataset of hand movements. Previous works rarely considered the multi-sensor datasets, which collected from the data gloves, to complete the action evaluation of VR training systems. In this paper, a vectorized graph convolutional deep learning model is proposed to evaluate the accuracy of test actions. First, the kernel of vectorized spatio-temporal graph convolutional of the data glove is constructed with different weights for different finger joints, and the data dimensionality reduction is also achieved. Then, different evaluation strategies are proposed for different actions. Finally, a convolution deep learning network for vectorized spatio-temporal graph is built to obtain the similarity between test actions and standard ones. The evaluation results of the proposed algorithm are compared with the subjective ones labeled by experts. The experimental results verify that the proposed action evaluation method based on the vectorized spatio-temporal graph convolutional is efficient for the manual operation accuracy evaluation in VR training systems of smart grids.

**Keywords:** virtual reality; manual operation accuracy evaluation; graph convolutional neural network

check for updates

**Citation:** He, F.; Liu, Y.; Zhan, W.; Xu, Q.; Chen, X. Manual Operation Evaluation Based on Vectorized Spatio-Temporal Graph Convolutional for Virtual Reality Training in Smart Grid. *Energies* **2022**, *15*, 2071. https://doi.org/10.3390/en15062071

Academic Editor: Pablo Cortés

Received: 15 December 2021
Accepted: 6 March 2022
Published: 11 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## 1. Introduction

With the rapid development of artificial intelligence and computer vision, the virtual reality (VR) technology has been widely used in vocational training, medical care, entertainment, criminal investigation, and other fields. The rapid development of smart grid brings more challenges to the security and stable operation of the power system, which makes the operation training of the power engineers very important. VR training resulted in better retention, task performance, learning speed, and engagement than the video training counterpart, maintaining system usability [1]. This paper proposes an interactive VR training system as an efficient and low-cost solution for training systems in smart grids, which have entered the era of large power grids with high voltage, long distance, and high parameters.

A simplified virtual reality training system for radiation shielding and measurement in nuclear engineering is proposed, which enables beginners and non-experts to experience the environment of radiation sources and radiation shielding walls [2]. In the VR training of power grid, the evaluation, recognition, and analysis of hand actions are very important. The evaluation results are obtained based on the data gloves, which is used to obtain the coordinates, angular velocity, motion direction, and other data information of the wrist and finger joints. These action data have high dimensionality and high difference, which makes them hard to deal with. How to efficiently evaluate the actions based on the movement dataset collected by data gloves is the critical problem for the application of the VR training system in smart grids.

A prerequisite for highly accurate movement evaluation is the accurate acquisition of relevant data on hand movements. VR technology can detect data on hand movements in real time, which is highly compatible with the need to obtain highly accurate data. Three-dimensional pick-and-place tasks (10Cubes) have been developed in a virtual world, 10Cubes could calculate the hand shake level for the cure of Parkinson's disease (PD) [3]. This paper combines artificial intelligence technology with VR technology, the target elements such as actions and objects in the virtual environment can be serialized and the required features can be extracted for analysis, and the relevant models can be used for adaptive learning to improve the accuracy of action evaluation by using VR technology.

The main contributions of this paper as follows: 1. The application of VR technology to the operation training of the power grid is proposed, and a method to process the data obtained from the data glove to datatize the action features is also proposed; 2. a deep learning model is proposed with high recognition accuracy for the action evaluation in the operation training of the power grid; 3. A graph convolution kernel with a spatio-temporal mechanism is constructed, adding an attention mechanism and applying a division strategy for hand skeletal joints to improve the input of vector graph convolution deep learning model and improve the accuracy of action evaluation.

The organization of this paper is presented as follows: 1. The Introduction presents the training of power grid program using VR technology and the problem of how to make the accuracy of action evaluation improved in a virtual environment and, finally, introduces the solution proposed in this paper; 2. The Literature Review introduces the current research results and solutions to related problems in the world and summarizes the problems found to be unsolved, introduces the data glove and proposes a solution; 3. The Vectorized Spatio-temporal Graph Convolutional for VR action evaluation used in this paper is introduced, and the method of constructing the graph convolution kernel, attention mechanism, and division strategy are also introduced; 4. In the Application section, the action data acquisition method is introduced, and the obtained experimental results are analyzed; 5. In the Conclusion, to summarize the whole paper, it is demonstrated that the Vectorized Spatio-temporal Graph Convolutional deep learning model can be applied to hand recognition in a virtual environment, and it is proven that the accuracy can be improved by applying spatio-temporal graph convolution, attention mechanism, and a hand skeletal joint division strategy to the Vectorized Spatio-temporal Graph Convolutional deep learning model.

## 2. Literature Review

The similarity function of humanoid robot imitating human motion based on motion rhythm is proposed, and the solution method of motion trajectory when it is highly similar is given [4]. The pose word bag method is proposed to align the action sequences, which can calculate the similarity between the two action sequences more accurately [5]. Based on the benchmark analysis method of Kinect technology, the differences of action characteristics of people of different ages are analyzed and the action characteristics are obtained [6]. A general combination of action features is proposed, and a multidimensional dynamic time adjustment method is used to evaluate the similarity of actions [7]. On the basis of the Kinect SDK obtaining the position information of bone joints and describing quaternion rotation, a human pose recognition algorithm that simulates human bone motion is proposed in [8]. A human body structure model combined with spatio-temporal information is proposed to describe action features and achieved good results in the evaluation of action similarity [9]. Space-time trajectory is used to represent the coordinates of limb joints and their changes with time, so that the action similarity evaluation is assigned to the similarity calculated for trajectory shape on Riemannian manifold [10]. A fast, simple, and more robust moving pose feature MP (Moving Pose) is proposed to evaluate the similarity of human motion with low delay [11]. On the basis of proposing a function similar to mime motion and establishing kinematic constraints including ground contact conditions, a high-precision motion algorithm satisfying kinematic constraints is proposed in [12]. A joint angle sequence model for human motion recognition is proposed, which uses Kinect sensors to obtain depth images and improves the matching between motion capture and images [13]. A full-connection depth Long-Short Memory Network (Long Short-Term Memory, LSTM) model is proposed to identify human action [14]. A human action recognition model based on the combination of LSTM and CNN is proposed in [15]. A construction worker's motion recognition model using the Long Short-Term Memory (LSTM) network based on an evaluation of the effectiveness of motion sensors' numbers and locations to maximize motion recognition performance is proposed in [16].

For human-specific action recognition problems based on extremely unbalanced datasets, a specific action recognition method suitable for extremely unbalanced datasets is proposed in [17]. Based on the characteristics of human skeleton sequence data to fully integrate data modeling and graph structure, the use of graph convolutional networks is proposed in [18]. A novel skeleton transformer module is designed based on the attention mechanism, which achieves automatic rearrangement and selection of important skeleton joints by linear transformation, and classifies actions using CNN models. This class of methods mainly uses the advantages of CNN model in dealing with image recognition problems to achieve action classification of skeleton sequences is proposed in [19]. Use a convolutional neural network (CNN) to estimate human poses by analyzing the projection of the depth and ridge data that use a convolutional neural network (CNN) to estimate human pose by analyzing the projection of the depth and ridge data, this method can eliminate the 3D information loss and drift of joint positions that can occur during the estimation of human poses [20]. The coordinates of the skeleton joints can be transformed into a tensor and fed into the designed Neural Network learning action features is proposed in [21]. A hierarchical RNN combines features from different body parts in layers. In the shallow layer, each sub-network extracts features on individual joints, and then fuses these feature representations in the layers. After all, the node information is fused, the network performing the final action is proposed in [22].

After the GCN was proposed, the graph attention network was designed according to the attention mechanism by superimposing the hidden self-attention layer, assigning different weights to different nodes in the neighborhood, and, finally, classifying nodes on the graph structure data is proposed in [23]. A GCN-based model for human action recognition was first proposed in 2018, called the ST-GCN (spatio-temporal graph convolutional network), in [24]. Subsequently, an Action-Structured GCN is proposed for human action recognition and prediction based on this model, which uses the attentional idea to design A-links to evaluate the importance of linkage between any nodes and uses S-link to obtain higher order dependencies of the graph and learns the spatio-temporal characteristics of the action by the combination of the two types of links. The GCN-based approach uses graph structure to model the skeleton and extracts features by convolutional operations on the graph to recognize and predict actions is proposed in [25].

Based on BP neural network, an intelligent optimization method of motion management system has carried out experiments and analysis from feature extraction, feature selection, and principal component analysis to the selection of support vector machine model functions [26]. A novel multiview video-based markerless system is proposed, that uses 2D joint detections per view (from OpenPose) to estimate their corresponding 3D positions while tackling the people association problem in the process to allow the tracking of multiple persons at the same time [27]. A novel algorithm called RSC-Net is proposed, which consists of a Resolution-aware network, a Self-supervision loss, and a Contrastive learning scheme, and this network is able to learn the 3D body shape and pose across different resolutions with a single model [28]. Rigorous robust benchmarks, termed COCO-C, MPII-C, and OCHuman-C, are built to evaluate the weaknesses of current advanced pose estimators, and a new algorithm termed AdvMix is proposed to improve their robustness in different corruptions [29]. To improve the practicality of the lower extremity exoskeleton robot, a wavelet packet transform (WPT)-based sliding window difference average filtering feature extract algorithm and the unscented Kalman neural network (UKFNN) recognition algorithm is proposed in [30]. For observing the state information between the object and the hand by using customized ten kinds of HIM manipulation in order to recognize the complex HIMs, a human in-hand motion (HIM) recognition system based on multi-modal perception information fusion is proposed in [31]. A hybrid feature selection method by combining filter and wrapper methods (FESCOM) was proposed to eliminate irrelevant features for motion recognition of upper-limb exercises [32].

An AR-based Training System for Piano Performance is proposed, two user studies conducted by us show that the system requires relatively less cognitive load and may increase learning efficiency and quality [33]. AR technology is another novel research direction, but it may require more devices for support.

In conclusion, many existing methods can extract action features from color and deep images, while multi-sensor vector datasets for data gloves have few processes for such data to complete action evaluation of VR systems. In the VR technology-based power grid training, human–computer interaction and data acquisition are inseparable from the data gloves, which are used in this paper as shown in the Figure 1.

**Figure 1.** VR data glove.

In view of the problems existing in the appellate research, this paper collects the data through the data gloves in the virtual reality environment, first preprocesses the data, and provides different evaluation strategies for different actions, so as to establish a Vectorized Spatio-temporal Graph Convolutional deep learning model to complete the action evaluation in the VR environment.

## 3. Vectorized Spatio-Temporal Graph Convolutional for VR Action Evaluation Method

Human motion recognition is an active research field in computer vision and has been widely used in various fields. However, traditional motion recognition and evaluation often rely on manual feature extraction. These methods will miss many high-level features contained in hand structure. Dynamic hand movement contains important operation information. In the operation training of power grid, the evaluation of hand movement needs higher accuracy. Therefore, a learning model that can improve the accuracy of action recognition is urgently needed.

The accuracy of action recognition for a certain action is usually expressed by the following equation:

$$G_{accuracy}(x) = \frac{N_{correct}(x)}{N_{total}(x)}, \tag{1}$$

where $x$ is the action type, $N_{correct}(x)$ is the number of times that the action sequences belonging to type $x$ in the test action dataset are correctly classified, and $N_{total}(x)$ is the total number of action sequences belonging to type $x$ in the test action dataset:

$$P(x) = max\{G_{accuracy}(x)\}, \tag{2}$$

a learning model to maximizes the value of the $G_{accuracy}(x)$ is needed.

In this paper, a vector graph convolution depth learning model is proposed to solve some problems existing in traditional motion recognition and evaluation and to improve the accuracy of motion recognition. Firstly, the hand motion data are obtained through the data glove, and the hand structure sequence is used as the input to construct the hand vector graph. Then, the depth features in the vector graph are extracted through the graph convolution network and adding attention mechanism to maximize the value of $P(x)$, in order to complete the comparison and evaluation of the tester's motion and standard motion.

### 3.1. Algorithm Refinement Process Design

This paper constructs a dataset by using the data obtained from the data glove to assist the power grid operator to correct the non-standard operation during his own training. Through cooperation with relevant professional staff in the industry, UE4 is used and three basic static functions are written in combination with the blueprint to collect the hand operation actions, record the hand bone key points in NTU RGB + D dataset format, and finally change the convolution core of ST-GCN network and train the dataset according to the hand actions during operation. The obtained training model can complete the hand movement recognition, and has strong generalization performance and robustness. The detailed flow chart of hand motion recognition algorithm is shown in Figure 2.

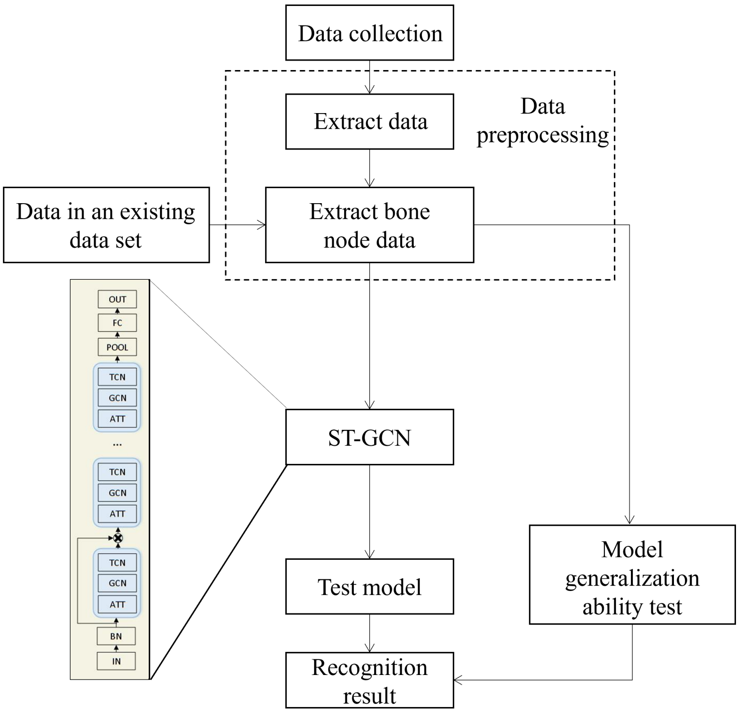


**Figure 2.** Algorithm refinement flow chart.

### 3.2. Algorithm Refinement Process Design

The article maximizes the value of $P(x)$ by learning the model using a Vectorized Spatio-temporal Graph Convolutional deep learning model. Different from 2D or 3D convolution neural networks, the implementation details of graph volume are different.

In spatio-temporal convolution graphs, the spatial graph in a single frame is represented by adjacency matrix A, and the self-connection in the graph is represented by identity matrix I. Therefore, in the volume product of a single frame spatial graph, the following equation can be used:

$$f_{out} = \Lambda^{\frac{1}{2}}(A+I)\Lambda^{\frac{1}{2}} f_{in} W. \tag{3}$$

For the spatio-temporal graph, $f_{in}$ in the equation is an input characteristic graph, which is represented by tensors $(C, V, T)$. The convolution is the result of standard two-dimensional convolution multiplied by normalized adjacency matrix.

Because the division of receptive fields in the process of graph volume always involves the division of subsets of adjacent nodes, different division methods have different implementation details. For non-unitary subset partition, the original adjacent matrix needs to be decomposed into several matrices for operation:

$$A + I = \sum_j A_j. \tag{4}$$

In the above equation, the adjacency matrix is disassembled into the sum of adjacency matrices in different subsets. For example, in the strategy mode divided according to the center of gravity of the hand structure, the $A_0 = I$, $A_1 = A_{centriptal}$, and $A_2 = A_{centrifual}$, the above equation is changed to the following:

$$f_{out} = \sum_j \Lambda^{-\frac{1}{2}} A_j \Lambda^{-\frac{1}{2}} f_{in} W_j. \tag{5}$$

The weight vectors of the plurality of output channels are stacked to form a weight matrix W. Actually, the feature map input in space-time is taken as the tensor $(C, V, T)$ dimension.

The model of the whole network is composed of 9 different spatio-temporal convolution layers, as shown in the Figure 3. Because different nodes need to share weights in the whole graph convolution process, the coherence between different input data is very important. When the data are sent into the whole model, it needs to go through a batch normalization layer to normalize all hand information. In order to prevent over fitting of data, this paper uses the Dropout method to randomly discard some data units, and sets the fourth and seventh layers as pooling layers. Finally, the feature vector is generated through full connection and classified by SoftMax function to explain what is the action input hand data sequence.

### *3.3. Construction of Hand Vector Graph*

After determining to use the Vectorized Spatio-temporal Graph Convolutional deep learning model, the hand vector map needs to be constructed as the input to this learning model. The data obtained by the data glove are a series of data for a period of time, including the information of hand joint points at different time points. In order to extract the motion features of the whole hand movement, it is necessary to construct a hand structure vector diagram, which is an undirected graph, as shown in Figure 3. $G = (V, E)$ is composed of T frames with N joint points in each frame. In this vector diagram, all nodes from different frames form the set $V = \{v_{ti} | t = 1, ..., T; i = 1, ..., N\}$, in which $t$ represents the serial numbers of all frames, and $i$ represents the serial numbers of key points in a certain frame. Therefore, the feature vector $F(v_{ti})$ of the i-th node in the t-th frame is the three-dimensional coordinate of the point. The eigenvectors of all nodes are used as the inputs of convolution neural network, and convolution calculation is carried out. Constructing the whole hand motion convolution graph can be divided into two steps. In the first step, in each frame sequence, all the hand joint points are connected in a natural connection mode to form a spatial map of the hand structure. Secondly, for different joint points of the hand, as shown in Figure 4, the same joint points in continuous frames are connected to form a time chart of the hand structure.

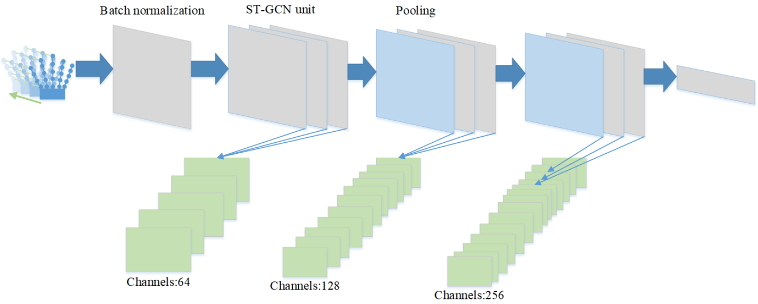


**Figure 3.** Figure convolutional network model structure.

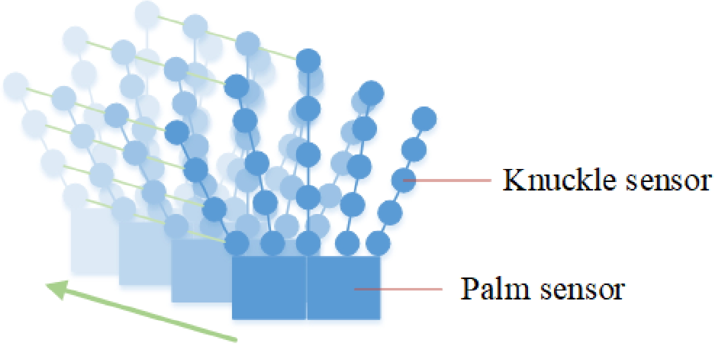


**Figure 4.** Hand actions vector diagram.

Then, the space map and time map are connected to form a Shi Kongtu based on the hand structure. Comparing graph convolution with CNN convolution network, the coordinates of each hand node can be regarded as a channel, and this model is applied to two-dimensional coordinates or three-dimensional coordinates. Similarly, the edge set E is composed of two molecular sets. The first subset is the connecting line of hand joint points in each frame, expressed as $E_S = \{v_{ti}v_{tj}|(i,j) \in H\}$, where $H$ is all joint points in the hand structure. The second subset is between frames, the same joint point connection, and this interframe connection is represented as $E_F = \{v_{ti}v_{(t+1)i}\}$. Therefore, for a particular joint point, connecting all its edges $E_S$ can be regarded as the spatial motion track of the joint point. All point sets and edge sets are hand vector graphs.

The connected hand motion vector graph can be compared with the input picture in CNN. The picture in CNN is equivalent to the pixel intensity vector arranged in the 2D image grid, while the skeleton motion space-time graph is the joint coordinate vector arranged in the 3D coordinate system. The classification method is roughly the same as CNN. The high-level feature map in the original skeleton space-time map is extracted by multi-layer convolution and finally classified by the SoftMax classifier.

### 3.4. Construction of Hand Vector Graph

After processing the input, the next step is to construct the graph convolution kernel using the learning model. At the $\tau$ moment, there are point set $V_t$ of N joint points in the spatial plot and the edge set is obtained from these joints $E_S = \{v_{ti}v_{tj}|(i,j) \in H\}$. Using a graph convolution kernel of a $K \times K$ size and an input convolution graph with c channels, the output at graph node $x$ on a certain channel can be shown in the following equation:

$$f_{out}(x) = \sum_{h=1}^{K} \sum_{W=1}^{K} f_{in}(P(x,h,w))W(h,w), \tag{6}$$

where $P$ represents the sampling function, which specifically means the set of adjacent nodes within the field of view of node $x$ in a graph. It is equivalent to $Z^2 \times Z^2 \to Z^2$.

In the traditional image convolution, this sampling process can be represented as the $P(x,h,w) = x + P^{'}(h,w)$, centered on the sampling function will obtain the adjacent surrounding pixels, plus the $x$ pixel itself, which is the visible receptive field.

In a graph convolution, if a graph node $v_{ti}$ is specified, its receptive field will be defined as the set of points adjacent to the point.

In graph volume, if a graph node is specified: $B(v_{ti}) = \{v_{tj}|d(v_{tj}, v_{ti} \leq D)\}$, where $d(v_{ti}, v_{tj})$ is represented as the shortest path from point $v_{ti}$ to point $v_{tj}$. In the graph convolution model, $D$ is set to 1, that is, the adjacent points of a node are all points with a distance of 1. Thus, it can be deduced that the sampling function $P : B(v_{ti}) \rightarrow V$ in the graph convolution can be expressed in the following equation:

$$P(v_{ti}, v_{tj}) = v_{tj}. \tag{7}$$

In Equation (6), $W$ is the weight function, which means that a weight vector with dimension C is used as the inner product with the input eigenvector of the sampling function. In the process of constructing the weight function of graph convolution, the point set $B(v_{ti})$ of a node $v_{ti}$ can be divided into k subsets, and each subset has corresponding labels which can simplify the heavy task of adding different labels to each adjacent node. Thus, the mapping of each point in the starting point set $B(v_{ti})$ to its corresponding subset can be established, $l_{ti} : B(v_{ti}) \rightarrow \{0, ..., K-1\}$, where $l_{ti}$ represents the mapping rules for the adjacent set of adjacent points. From this rule, the tensor $(C, K)$ can be used to construct the weight function as follows: $W(v_{ti}, v_{tj}) : B(v_{ti}) \rightarrow R^{C}$. Then:

$$W(v_{ti}, v_{tj}) = W^{'}(l_{ti}(v_{tj})), \tag{8}$$

the main convolutional kernel of the spacetime map can thus be established.

As the redefinition adopts the function with the weight function, Equation (6) can be applied to the graph convolution:

$$f_{out}(x) = \sum_{v_{ti} \in B(v_{ti})} \frac{1}{Z_{ti}(v_{ti})} f_{in}(P(v_{ti}, v_{tj})W(v_{ti}, v_{tj})). \tag{9}$$

Here, the regularization term $Z_{ti}(v_{tj}) = |\{v_{tk}|l_{ti}(v_{tj})\}|$ equal to the cardinality of the corresponding subset. This term increases the contribution of the different subsets.

From Equations (7)–(9), it can be obtained:

$$f_{out}(x) = \sum_{v_{ti} \in B(v_{ti})} \frac{1}{Z_{ti}(v_{ti})} * w^{'}(l_{ti}(v_{tj})). \tag{10}$$

After defining the spatial graph CNN, now begin modeling the spatio-temporal dynamics in the skeletal sequence. Recall that in the construction of the graph, the temporal aspect of the graph was constructed by connecting the same joints between consecutive frames. This allows us to define a very simple strategy to extend the spatial graph CNN to the spatio-temporal domain. This paper extends the notion of a neighborhood to include joints with also temporal connections. Extending the model of the spatial domain into the time domain, the resulting adoption function is the following Equation (11):

$$B(v_{ti}) = \{v_{qj}|d(v_{tj}, v_{ti}) \leq K, |q-t| \leq \lfloor \frac{\Gamma}{2} \rfloor\}, \tag{11}$$

where $\Gamma$ is the convolutional kernel size of the control time domain.

To complete the convolution operation on the ST graph, the weight function is needed too, which is the same as the unique case of the spatial graph. Because the timeline is ordered, directly modified the label mapping $l_{ST}$to be according to $v_{ti}$ generates a spacetime neighborhood, resulting in the weight function Equation (12), where $l_{ti}(v_{tj})$ is the label mapping for single frame case at $v_{ti}$:

$$l_{ST}(v_{qj}) = l_{ti}(v_{tj}) + (q - t + \lfloor \Gamma/2 \rfloor) \times K. \tag{12}$$

In this way, convolution operations are defined for the constructed spatio-time graph.

### 3.5. Attention Mechanism

In the process of exercise, the importance of different trunk is different. For example, leg movements may be more important than the neck. Through the leg movements, running, walking, and jumping can be judged, but the neck movements may not contain much effective information. Therefore, different torsos can be weighted.

By transforming Equation (9) the equation of graph convolution in spatial dimension is:

$$f_{out}(x) = \sum_{k}^{K_v} w_k(f_{in}(\widetilde{A_k} \odot M_k)). \tag{13}$$

The spatial graph convolution layer is modified, and the graph attention module is introduced so that the model can not only learn the parameters of the network, but also optimize the connected graph to obtain a graph structure more suitable for describing actions, so as to better predict actions. Specifically, after adding the graph attention module, the spatial graph convolution equation can be expressed as:

$$f_{out}(x) = \sum_{k}^{K_v} w_k(f_{in}(A_k^{'} + B_k)). \tag{14}$$

Compared with Equations (13) and (14) can be seen that the graph attention module includes two parts, in which $A^{'}$ is a data-driven graph matrix. Firstly, it initializes the parameters through the graph volume kernel constructed in Section 3.4 and then updates the parameters in the process of network propagation. Different from Equation (13), $\widetilde{A}$ is a numerically fixed adjacency matrix, and M learns the strength of the connection. No new connection structure can be generated in the whole training process. The use of a new matrix $A^{'}$ can completely replace the effect of interaction with M. Matrix $A^{'}$ can not only make full use of the initial physical connection relationship but can also optimize the topology of the connected graph in the training process and update the weight of the edge, so as to realize the effect of replacing two matrices with one matrix. In addition, $A^{'}$ is unique in different convolution layers, so it is personalized in each layer and contains different semantics.

The second part of the module is the graph attention matrix $B$, which can help the model better model the action for each sample and increase the personalization of the model. Specifically, for an input feature, $f(v_{ti})$(the feature of a node $v_{ti}$), two convolution layers are first used to map $f(v_{ti})$ into vectors $K$ and $Q$, namely:

$$K_{ti} = W_K f(v_{ti}), \tag{15}$$

$$Q_{ti} = W_Q f(v_{ti}), \tag{16}$$

where $W_K$ and $W_Q$ are the weight matrices corresponding to the two convolution layers, respectively. Next, calculate the inner product $u_{(t,i)\to(t,j)} = <Q_{ti}, K_{ti}>$ of $Q_{ti}$ ($Q$ vector of node $v_{ti}$) and $K_{ti}$ (k vector of node $v_{ti}$). Where nodes $v_{ti}$ and $v_{tj}$ are in the same time step; $\langle , \rangle$ stands for inner product symbol. Inner product $u_{(t,i)\to(t,j)}$ is called the similarity of node $v_{ti}$ and node $v_{tj}$. Then, in order to limit the range of u to 0∼1, it is normalized by SoftMax function, i.e.,:

$$\alpha_{(t,i)\to(t,j)} = \frac{exp(u_{(t,i)\to(t,j)})}{\sum_{n=1}^{N} exp(u_{(t,i)\to(t,n)})}, \tag{17}$$

$\alpha$ is the normalized similarity of inner product u, that is, the element of matrix B in Equation (14). It can be seen that matrix B is also completely learned from different action samples. It can effectively learn the weights of any two body joint points in different actions. This data-driven approach increases the flexibility and versatility of the model and enables the model to effectively predict actions in the face of diverse data. By adding the above-mentioned graph attention module, the network can continuously optimize the graph structure in the training process, adapt to the changes of various samples, and form the topology that is most suitable for describing actions, thus finally improving the performance of the model and making the results of action prediction more accurate.

### *3.6. Partitioning Strategy*

In the process of constructing the convolution kernel of the spatio-temporal graph, it involves the subset division of the adjacent node set of a root node. Because the spatio-temporal graph is not similar to the image, it has a rigid spatial sequence structure. Therefore, the size of the receptive field of a point and the subset division of adjacent point sets play an important role in completing the whole convolution. To determine the subset partition strategy of point set, the following three partition strategies are designed in determining mapping 1.

The first is the unique partition strategy, which divides the root node and all adjacent nodes into a subset. In this strategy, the eigenvector of each node is an inner product of the same weight vector. This strategy is used to represent the inner product of the average eigenvector and weight vector of all adjacent nodes. This partition strategy is expressed as $K = 1$ and $l_{ti}(v_{ti}) = 0, \forall i, j \in V$.

The second partition strategy is based on distance. The nodes around the root node $v_{ti}$ are divided according to their distance $d(v_{tj})$ from the root node. Since the limit of distance D is set to 1 in the process of constructing convolution kernel, the subset can be divided into two parts. When $d = 0$, it means that the root node itself is a subset. When $d = 1$, it means that all points with a distance of 1 from the root node are a subset. Having two subsets means that there are two different weight vectors. This division can be expressed as $k = 2$ and $l_{ti}(v_{ti}) = d(v_{ti}, v_{tj})$.

The third division strategy is the spatial configuration division proposed for hand movement. This division strategy divides the whole point set into three subsets, in which the root node itself is a subset, the adjacent nodes closer to the center of gravity of the whole hand structure than the root node itself are divided into a subset, and the adjacent nodes farther away from the center of gravity of the whole hand structure than the root node itself are divided into a subset. The center of gravity of the hand structure is determined by the average three-dimensional coordinates of all joint points in the same frame. The theoretical basis of this division strategy is that in the process of hand motion, and motion can be simply divided into centripetal motion and centrifugal motion. This division strategy can be expressed as the following equation:

$$l_{ti}(v_{ti}) = \begin{cases} 0. & \text{if } r_j = r_l \\ 1. & \text{if } r_j < r_l \\ 2. & \text{if } r_j > r_l, \end{cases} \tag{18}$$

where $r_i$ represents the average distance from all nodes in the figure to the center of gravity of the hand structure, which is obtained from all data information in the training set.

Using different division strategies for different action types can lead to a substantial increase in the accuracy of action recognition. This paper mainly focuses on action evaluation of hand movements, so the third division strategy is used to maximize the value of $P(x)$ .

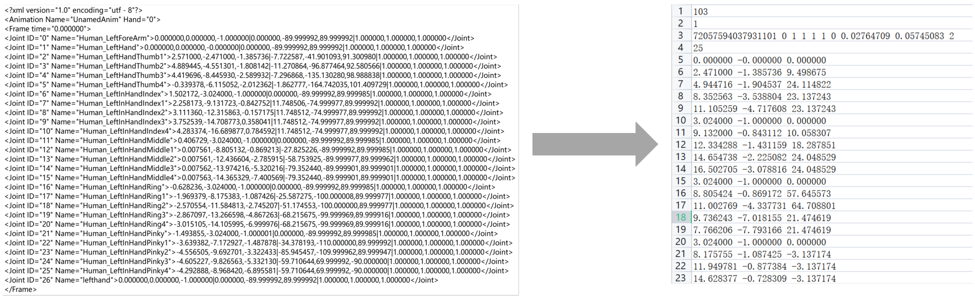
## **4. Application Instances**

There are many risk points in power operations, and these risks can expose the safety of power operations in a more comprehensive way. The virtual operation experiments in this paper are carried out in the laboratory, and the experiments are repeated for the key actions in power operations, and the corresponding data are recorded for the virtual operation action evaluation experiments.

### *4.1. Method of Action Data Acquisition*

During training, it is necessary to obtain the data of the data gloves in virtual environment in real time. After many tests, the hand model is divided into several nodes according to the finger joints, and the motion data are obtained. Meanwhile, for readability, the action data are saved in XML format , as shown in the Figure 5.

Because the NTU RGB + D action recognition dataset is a dataset trained with data features, which is consistent with the idea of extracting hand actions features and saving them with data described above, and the NTU RGB + D action recognition dataset has a huge basic database, the data of the data glove stored in the XML file above are integrated into the data of the NTU RGB + D dataset.



```
<?xml version="1.0" encoding="utf - 8"?>
<Animation Name="UnamedAnim" Hand="0">
<Frame time="0.000000">
<Joint ID="0" Name="Human_LeftForeArm">0.000000,0.000000,-1.000000|0.000000,-89.999992,89.999992|1.000000,1.000000,1.000000</Joint>
<Joint ID="1" Name="Human_LeftHand">0.000000,0.000000,-0.000000|0.000000,-89.999992,89.999992|1.000000,1.000000,1.000000</Joint>
<Joint ID="2" Name="Human_LeftHandThumb1">2.571000,-2.471000,-1.385736|-7.722587,-41.901093,91.300980|1.000000,1.000000,1.000000</Joint>
<Joint ID="3" Name="Human_LeftHandThumb2">4.889445,-4.551301,-1.808142|-11.270864,-96.877464,92.580566|1.000000,1.000000,1.000000</Joint>
<Joint ID="4" Name="Human_LeftHandThumb3">4.419696,-8.445930,-2.589932|-7.296868,-135.130280,98.988838|1.000000,1.000000,1.000000</Joint>
<Joint ID="5" Name="Human_LeftHandThumb4">-0.339378,-6.115052,-2.012362|-1.862777,-164.742035,101.409729|1.000000,1.000000,1.000000</Joint>
<Joint ID="6" Name="Human_LeftInHandIndex">1.502172,-3.024000,-1.000000|0.000000,-89.999992,89.999985|1.000000,1.000000,1.000000</Joint>
<Joint ID="7" Name="Human_LeftInHandIndex1">2.258173,-9.131723,-0.842752|11.748506,-74.999977,89.999992|1.000000,1.000000,1.000000</Joint>
<Joint ID="8" Name="Human_LeftInHandIndex2">3.111360,-12.315863,-0.157175|11.748512,-74.999977,89.999992|1.000000,1.000000,1.000000</Joint>
<Joint ID="9" Name="Human_LeftInHandIndex3">3.752539,-14.708773,0.358041|11.748512,-74.999977,89.999992|1.000000,1.000000,1.000000</Joint>
<Joint ID="10" Name="Human_LeftInHandIndex4">4.283374,-16.689877,0.784592|11.748512,-74.999977,89.999992|1.000000,1.000000,1.000000</Joint>
<Joint ID="11" Name="Human_LeftInHandMiddle">0.406729,-3.024000,-1.000000|0.000000,-89.999992,89.999985|1.000000,1.000000,1.000000</Joint>
<Joint ID="12" Name="Human_LeftInHandMiddle1">0.007561,-8.805132,-0.869213|-27.825226,-89.999992,89.999985|1.000000,1.000000,1.000000</Joint>
<Joint ID="13" Name="Human_LeftInHandMiddle2">0.007561,-12.436604,-2.785915|-58.753925,-89.999977,89.999962|1.000000,1.000000,1.000000</Joint>
<Joint ID="14" Name="Human_LeftInHandMiddle3">0.007562,-13.974216,-5.320216|-79.352440,-89.999901,89.999901|1.000000,1.000000,1.000000</Joint>
<Joint ID="15" Name="Human_LeftInHandMiddle4">0.007563,-14.365329,-7.400569|-79.352440,-89.999901,89.999901|1.000000,1.000000,1.000000</Joint>
<Joint ID="16" Name="Human_LeftInHandRing">-0.628236,-3.024000,-1.000000|0.000000,-89.999992,89.999985|1.000000,1.000000,1.000000</Joint>
<Joint ID="17" Name="Human_LeftInHandRing1">-1.969379,-8.175383,-1.087426|-25.587275,-100.000008,89.999977|1.000000,1.000000,1.000000</Joint>
<Joint ID="18" Name="Human_LeftInHandRing2">-2.570554,-11.584813,-2.745207|-51.174553,-100.000000,89.999977|1.000000,1.000000,1.000000</Joint>
<Joint ID="19" Name="Human_LeftInHandRing3">-2.867097,-13.266598,-4.867263|-68.215675,-99.999969,89.999916|1.000000,1.000000,1.000000</Joint>
<Joint ID="20" Name="Human_LeftInHandRing4">-3.015105,-14.105995,-6.999976|-68.215675,-99.999969,89.999916|1.000000,1.000000,1.000000</Joint>
<Joint ID="21" Name="Human_LeftInHandPinky">-1.493855,-3.024000,-1.000001|0.000000,-89.999992,89.999985|1.000000,1.000000,1.000000</Joint>
<Joint ID="22" Name="Human_LeftInHandPinky1">-3.639382,-7.172927,-1.487878|-34.378193,-110.000000,89.999992|1.000000,1.000000,1.000000</Joint>
<Joint ID="23" Name="Human_LeftInHandPinky2">-4.556505,-9.692701,-3.322433|-85.945457,-109.999962,89.999947|1.000000,1.000000,1.000000</Joint>
<Joint ID="24" Name="Human_LeftInHandPinky3">-4.605227,-9.826563,-5.332130|-59.710644,69.999992,-90.000000|1.000000,1.000000,1.000000</Joint>
<Joint ID="25" Name="Human_LeftInHandPinky4">-4.292888,-8.968420,-6.895581|-59.710644,69.999992,-90.000000|1.000000,1.000000,1.000000</Joint>
<Joint ID="26" Name="lefthand">0.000000,0.000000,-1.000000|0.000000,-89.999992,89.999992|1.000000,1.000000,1.000000</Joint>
</Frame>
```

XML File action data display

```
1   103
2   1
3   72057594037931101 0 1 1 1 1 0 0.02764709 0.05745083 2
4   25
5   0.000000 -0.000000 0.000000
6   2.471000 -1.385736 9.498675
7   4.944716 -1.904537 24.114822
8   8.352563 -3.538804 23.137243
9   11.105259 -4.717608 23.137243
10  3.024000 -1.000000 0.000000
11  9.132000 -0.843112 10.058307
12  12.334288 -1.431159 18.287851
13  14.654738 -2.225082 24.048529
14  16.502705 -3.078816 24.048529
15  3.024000 -1.000000 0.000000
16  8.805424 -0.869172 57.645573
17  11.002769 -4.337731 64.708801
18  9.736243 -7.018155 21.474619
19  7.766206 -7.793166 21.474619
20  3.024000 -1.000000 0.000000
21  8.175755 -1.087425 -3.137174
22  11.949781 -0.877384 -3.137174
23  14.628377 -0.728309 -3.137174
```

Selected test NTU RGB + D action data set data display

**Figure 5.** Action data conversion diagram.

The dataset contains 60 categories of actions with a total sample of 56,880, of which 40 categories are daily behavioral actions, 9 categories are health-related actions, and 11 categories are two-person mutual actions. These actions were performed by 40 individuals ranging in age from 10 to 35 years old. The dataset was acquired by the Microsoft Kinect v2 sensor and used three different camera angles, with data captured in the form of depth information, 3D skeletal information, RGB frames, and infrared sequences , as shown in the Table 1. In this paper, five important hand movements in the NTU RGB + D dataset are mainly used as data collection and training for gesture recognition.

**Table 1.** Select the test action dataset.

| | | | | |
|---|---|---|---|---|
| A1: drink water | A2: eat meal | A3: brush teeth | A4: brush hair | A5: drop |
| A6: pick up | A7: throw | A8: sit down | A9: stand up | A10: clapping |
| A11: reading | A12: writing | A13: tear up paper | A14: put on jacket | A15: take off jacket |
| A16: put on a shoe | A17: take off a shoe | A18: put on glasses | A19: take off glasses | A20: put on a hat/cap |
| A21: take off a hat/cat | A22: cheer up | A23: hand waving | A24: kicking something | A25: reach into pocket |
| A31: point to something | A32: taking a selfie | A33: check time (from watch) | A34: rub two hands | A35: nod head/bow |
| A36: shake hands | A37: wipe face | A38: salute | A39: put palms together | A40: cross hands in front |
| A41: sneeze/cough | A42: staggering | A43: falling down | A44: headache | A45: chest pain |
| A46: back pain | A47: neck pain | A48: nausea | A49: fan self | A50: punch/slap |
| A51: kicking | A52: pushing | A53: pat on back | A54: point finger | A55: hugging |
| A56: giving object | A57: touch pocket | A58: shaking hands | A59: walking towards | A60: walking apart |

### *4.2. Experiment and Result Analysis*

In order to verify the reliability and feasibility of the vector graph convolution depth learning model proposed in this paper, it is verified by the accuracy of motion recognition. Taking the whole VR interactive system based on motion recognition as the experimental object, the actions made by experimenters in the laboratory were collected, and the feasibility of the model was determined by recording the recognition accuracy of different actions. Because the interactive mode of motion recognition abandons the problems of key fixation and interactive rigidity in the traditional interactive mode, it embodies the advantages of the new VR interactive mode. According to the research of actual hand movements and the types of hand movements in specific application scenarios, and according to the characteristics of hand movements, reflect the state of hand movements as much as possible. The main flow of the ST-GCN experiment is shown in the Figure 6.

This paper estimates the posture of hand movements and construct spatio-temporal maps on bone sequences. After that, the multi-layer spatio-temporal image convolution operation (ST-GCN) is applied, and a higher-level feature map is gradually generated on the image. Then, it can be classified into corresponding operation categories by using the standard SoftMax classifier.

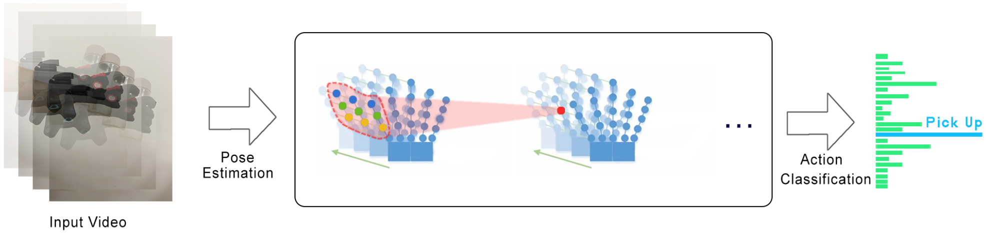


**Figure 6.** ST-GDN experiment flow chart.

Because ST-GCN shares weights on different nodes, it is very important to keep the input data ratio consistent on different nodes. In the experiment, the data are regularized and input into batch normalization. The ST-GCN model consists of nine layers of spatio-temporal map convolution. The first 3 layers output 64 channels, the middle 3 layers output 128 channels, and the last 3 layers output 256 channels. There are nine temporal convolution kernels. In each ST-GCN, residual linking is used, shedding is used for feature regularization, and half of the neurons are shed. The temporal convolution layers of the fourth and seventh layers are set as polarization layers. Finally, the outputs of 256 channels are collected and classified globally by SoftMax. It is optimized by SGD. The learning rate is set to 0.01 and reduced by 0.01 every 10 epoch iterations.

In this paper, five hand movements, A5, A16, A20, A33, and A34, commonly used in Table 1 are taken as experimental movements, because other hand movements similar to rubbing two hands, putting on a shoe, etc., are not conducive to testing the accuracy of finger actions in hand movements. The cross-validation strategy is adopted, that is, half of the data is used to train the classifier, and the other half of the data is used to verify the classification accuracy. Five hundred interactive experiments were conducted on these five hand movements, and the data were recorded through the data acquisition method in Section 4.1. The whole data sequence of each action is used to construct the hand action vector diagram. Half of the data is used to train the model, and the other half is used to predict. The experimental results of action recognition accuracy are shown in Table 2.

In the NTU RGB + D dataset, there are 948 samples for each action type. Sample number in Table 2 is the data number for each action type. In this paper, 52 self-test samples to the sample number for each action type were added into the NTU RGB + D dataset, then added our own data to each experiment.

The experimental results in Table 2 can be used to obtain the histogram of the accuracy rate in relation to Sample number, Action type, and Training number, respectively, for better observation of the experimental results and change patterns.

The histogram of the effect of sample number in Figure 7 shows that when the action type is 60 and the training number is 10, the accuracy of action recognition increases as the number of samples increases. When the sample number for each action is 20, the degree of TOP1 accuracy is 3.33%, and the degree of TOP5 accuracy is 10.28%; when the sample number for each action is 30, the degree of TOP1 accuracy is 5.84%, and the degree of TOP5 accuracy is 23.92%; when the sample number for each action is 50, the degree of TOP1 accuracy is 11.77%, and the degree of TOP5 accuracy is 39.07%; when the sample number for each action is 500, the degree of TOP1 accuracy is 60.76%, and the degree of TOP5 accuracy is 89.78%. It can be seen that when the sample number of each action is increasing, the degree of TOP1 accuracy obviously increases. It can be determined that the sample number is the key factor affecting the maximization accuracy.

**Table 2.** Accuracy test results.

| Action Type | Sample Number | Training Number | Accuracy Rate TOP1 | Accuracy Rate TOP5 |
|---|---|---|---|---|
| 60 | 10 | 10 | 2.50% | 16.67% |
| 60 | 20 | 10 | 3.33% | 10.28% |
| 60 | 30 | 10 | 5.84% | 23.92% |
| 60 | 40 | 10 | 9.77% | 35.40% |
| 60 | 50 | 10 | 11.77% | 39.07% |
| 5 | 500 | 10 | 58.55% | 100% |
| 10 | 500 | 10 | 65.84% | 97.59% |
| 20 | 500 | 10 | 59.04% | 90.23% |
| 30 | 500 | 10 | 60.30% | 90.70% |
| 60 | 500 | 10 | 60.76% | 89.78% |
| 60 | 500 | 20 | 71.30% | 93.62% |
| 60 | 500 | 40 | 72.89% | 94.21% |
| 60 | 500 | 60 | 77.05% | 95.17% |
| 60 | 500 | 80 | 74.94% | 94.60% |
| 5 | 500 | 160 | 67.04% | 100% |
| 60 | 1000 | 80 | 82.84% | 97.44% |

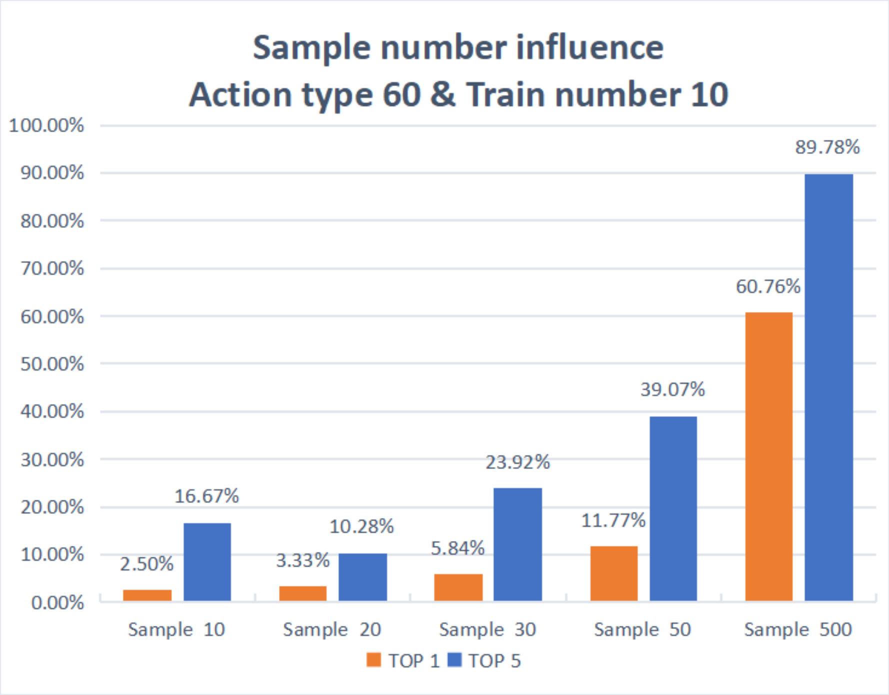


**Figure 7.** Histogram of the effect of Sample number.

The histogram of the effect of action type in Figure 8 shows that when the sample number is 500 and the train number is 10, the accuracy of action recognition increases with the increase in the action type. When the action type is 5, the degree of TOP1 accuracy is 58.55%, and the degree of TOP5 accuracy is 100%; when the action type is 10, the degree of TOP1 accuracy is 65.84%, and the degree of TOP5 accuracy is 97.59%; when the action type is 20, the degree of TOP1 accuracy is 59.04%, and the degree of TOP5 accuracy is 90.23%; when the action type is 40, the degree of TOP1 accuracy is 60.30% and the degree of TOP5 accuracy is 90.70%; when the action type is 60, the degree of TOP1 accuracy is 60.76% and the degree of TOP5 accuracy is 89.78%. It can be seen that the degree of TOP1 accuracy increases when the action type keeps increasing, but the increase is not significant, and it can be determined that the action type is not the key factor affecting the maximized accuracy.

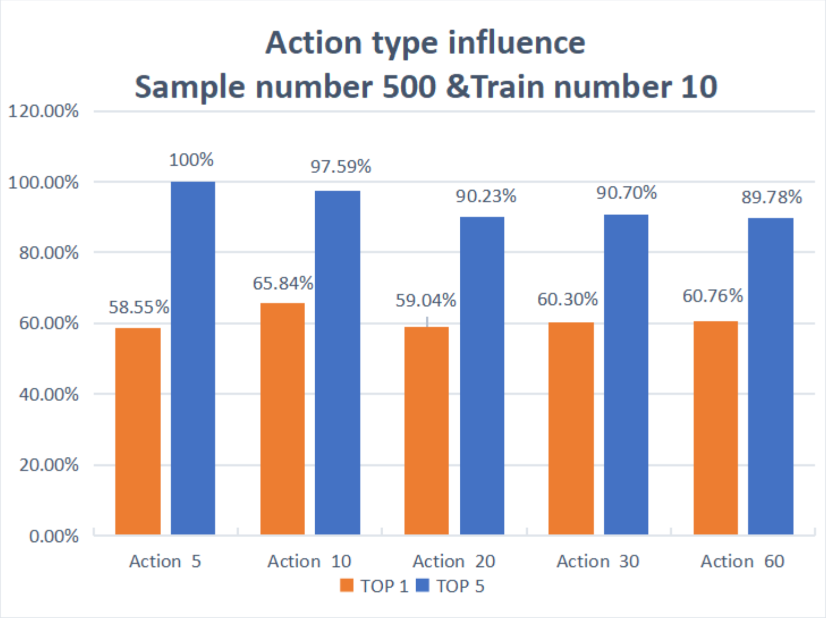


**Figure 8.** Histogram of the impact of action type.

The histogram of the impact of train number in Figure 9 shows that when the sample number is 500 and the action type is 60, the accuracy of action recognition increases with the increase in the train number. When the train number is 10, the degree of TOP1 accuracy is 60.76% and the degree of TOP5 accuracy is 89.78%; when the train number is 20, the degree of TOP1 accuracy is 71.30%, and the degree of TOP5 accuracy is 93.62%; when the train number is 40, the degree of TOP1 accuracy is 72.89%, and the degree of TOP5 accuracy is 94.21%; when the train number is 60, the degree of TOP1 accuracy is 77.05%, and the degree of TOP5 accuracy is 95.17%; when the train number is 80, the degree of TOP1 accuracy is 74.94%, and the degree of TOP5 accuracy is 94.60%. It can be seen that the degree of TOP1 accuracy increases as the train number increases, but the increase is not significant, and it can be determined that the train number is not a key factor affecting the maximization of accuracy.

In order to verify the reliability and feasibility of the whole VR interaction system, this paper conducts experimental verification by three indices: action type, sample number, and train number. Taking the whole VR interaction system based on action evaluation as the experimental object, tested the actions performed by the experimenter in the laboratory and determined the feasibility of Vectorized Spatio-temporal Graph Convolutional for VR action evaluation method by recording the recognition accuracy of VR interaction operations under different training states, and as seen by the experimental results, when the number of samples is greater than or equal to 500 and the number of training times is greater than or equal to 80, the accuracy of all the actions tested exceeds 70%, which illustrates the reliability of the hand action recognition of the model proposed in this paper. However, there is a large difference in the accuracy rate between different actions, and the recognition accuracy is better for the actions with larger amplitude and left-right expansion, while the recognition accuracy is worse for the actions with smaller amplitude and front-back expansion, and the speed and frequency are not easy to be too fast when switching between different actions; otherwise, it may easily lead to possible misoperation or reduce the accuracy of command operation.

On the whole, the most critical factor for maximizing the value of $P(x)$ is the sample number of each action type.

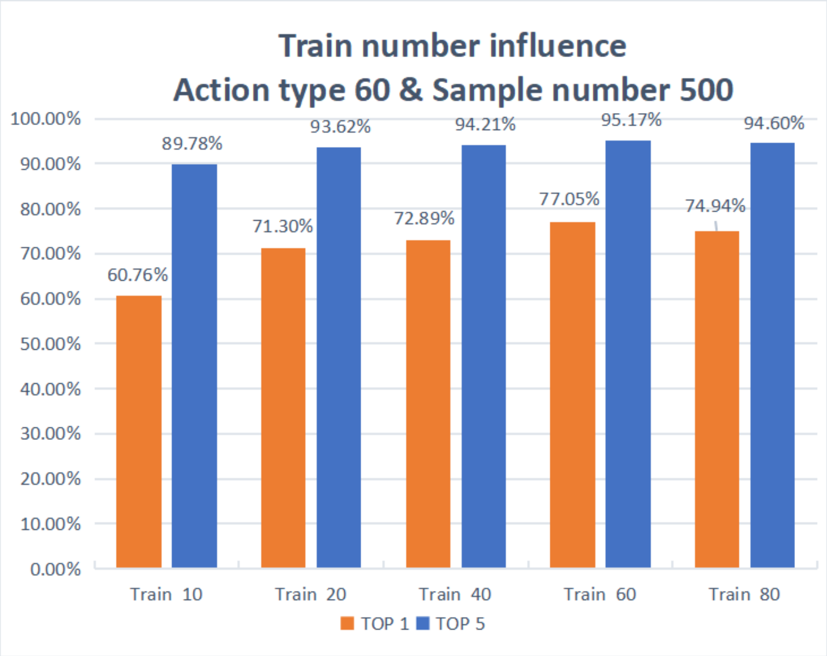


**Figure 9.** Histogram of the impact of the train number.

## 5. Conclusions

In view of the less research on hand actions evaluation using glove data in the existing power grid virtual environment, a vector graph convolution depth learning model is proposed for action evaluation. The spatio-temporal map of hand movement integrates the temporal and spatial information in the process of hand movement change and improves the integrity of features. Different partition strategies are designed for graph convolution, and experiments are carried out to verify the superiority of the model. Through the experimental contents and relevant data results mentioned in the article, the spatio-temporal convolution depth learning model can train and evaluate the data collected from the relevant actions of the data glove and integrated into the NTU RGB + D dataset and has good results in testing the accuracy. Therefore, spatio-temporal convolution map neural network can be applied to hand recognition in a virtual environment.

**Author Contributions:** F.H.: conceptualization, algorithm innovation, methodology, and writing—original draft; Y.L.: data and formal analysis, investigation, software, simulation, and writing—original draft; W.Z.: conceptualization, simulation, investigation, methodology, writing—original draft; Q.X.: investigation, and writing—review and editing; X.C.: formal analysis, and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by State Grid Corporation Headquarters Science and Technology Project Grant (Project No.: 5700-202217206A-1-1-ZN); by Science and Technology Project of State Grid Corporation of China: Research and demonstration application of key technology of full range calibration of metering low voltage current transformer under live condition; by the Humanities and social sciences research project of Hubei Provincial Education Department (No. 21q136).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data sharing not applicable.

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

## References

1. Osti, F.; de Amicis, R.; Sanchez, C.A.; Tilt, A.B.; Prather, E.; Liverani, A. A VR training system for learning and skills development for construction workers. *Virtual Real.* **2021**, *25*, 523–538. [CrossRef]
2. Hagita, K.; Kodama, Y.; Takada, M. Simplified virtual reality training system for radiation shielding and measurement in nuclear engineering. *Prog. Nucl. Energy* **2020**, *118*, 103127. [CrossRef]
3. Cikajlo, I.; Pogačnik, M. Movement analysis of pick-and-place virtual reality exergaming in patients with Parkinson's disease. *Technol. Health Care* **2020**, *28*, 391–402. [CrossRef] [PubMed]
4. Peng, Z.; Huang, Q.; Zhang, L.; Jafri, A.R.; Zhang, W.; Li, K. Humanoid on-line pattern generation based on parameters of off-line typical walk patterns. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; pp. 3758–3763.
5. Seidenari, L.; Varano, V.; Berretti, S.; Bimbo, A.; Pala, P. Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Portland, OR, USA, 23–28 June 2013; pp. 479–485.
6. Leightley, D.; Yap, M.H.; Coulson, J.; Barnouin, Y.; McPhee, J.S. Benchmarking human motion analysis using kinect one: An open source dataset. In Proceedings of the 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China, 16–19 December 2015; pp. 1–7.
7. Liu, Y.; Lu, G.; Yan, P. Exploring Multi-feature Based Action Recognition Using Multi-dimensional Dynamic Time Warping. In *Information Science and Applications (ICISA) 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 421–429.
8. Liu, X.; Feng, X.; Pan, S.; Peng, J.; Zhao, X. Skeleton tracking based on Kinect camera and the application in virtual reality system. In Proceedings of the 4th International Conference on Virtual Reality, Hong Kong, China, 24–26 February 2018; pp. 21–25.
9. Wang, C.; Wang, Y.; Yuille, A.L. An approach to pose-based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 915–922.
10. Devanne, M.; Wannous, H.; Berretti, S.; Pala, P.; Daoudi, M.; Del Bimbo, A. 3-d human action recognition by shape analysis of motion trajectories on riemannian manifold. *IEEE Trans. Cybern.* **2014**, *45*, 1340–1352. [CrossRef] [PubMed]
11. Zanfir, M.; Leordeanu, M.; Sminchisescu, C. The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 2752–2759.
12. Zhao, X.; Huang, Q.; Peng, Z.; Li, K. Kinematics mapping and similarity evaluation of humanoid motion based on human motion capture. 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; Volume 1, pp. 840–845.
13. Li, G.; Li, C. Learning skeleton information for human action analysis using Kinect. *Signal Process. Image Commun.* **2020**, *84*, 115814. [CrossRef]
14. Slama, R.; Wannous, H.; Daoudi, M.; Srivastava, A. Accurate 3D action recognition using learning on the Grassmann manifold. *Pattern Recognit.* **2015**, *48*, 556–567. [CrossRef]
15. Presti, L.L.; La Cascia, M.; Sclaroff, S.; Camps, O. Gesture modeling by hanklet-based hidden markov model. In Proceedings of the Asian Conference on Computer Vision, Singapore, 1–5 November 2014; pp. 529–546.
16. Kim, K.; Cho, Y.K. Effective inertial sensor quantity and locations on a body for deep learning-based worker's motion recognition. *Autom. Constr.* **2020**, *113*, 103126. [CrossRef]
17. Liu, Y.; You, X. Specific action recognition method based on unbalanced dataset. In Proceedings of the 2019 IEEE 2nd International Conference on Information Communication and Signal Processing (ICICSP), Weihai, China, 28–30 September 2019; pp. 454–458.
18. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
19. Li, C.; Zhong, Q.; Xie, D.; Pu, S. Skeleton-based action recognition with convolutional neural networks. In Proceedings of the 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Hong Kong, China, 10–14 July 2017; pp. 597–600.
20. Kim, Y.; Kim, D. A CNN-based 3D human pose estimation based on projection of depth and ridge data. *Pattern Recognit.* **2020**, *106*, 107462. [CrossRef]
21. Li, C.; Hou, Y.; Wang, P.; Li, W. Joint distance maps based action recognition with convolutional neural networks. *IEEE Signal Process. Lett.* **2017**, *24*, 624–628. [CrossRef]
22. 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.
23. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv* **2017**, arXiv:1710.10903.
24. 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, LI, USA, 2–7 February 2018.
25. Li, M.; Chen, S.; Chen, X.; Zhang, Y.; Wang, Y.; Tian, Q. Actional-structural graph convolutional networks for skeleton-based action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3595–3603.

26. Li, T.; Sun, J.; Wang, L. An intelligent optimization method of motion management system based on BP neural network. *Neural Comput. Appl.* **2021**, *33*, 707–722. [CrossRef]
27. Slembrouck, M.; Luong, H.; Gerlo, J.; Schütte, K.; Cauwelaert, D.V.; Clercq, D.D.; Vanwanseele, B.; Veelaert, P.; Philips, W. Multiview 3D markerless human pose estimation from openpose skeletons. In Proceedings of the International Conference on Advanced Concepts for Intelligent Vision Systems, Auckland, New Zealand, 10–14 February 2020; pp. 166–178.
28. Xu, X.; Chen, H.; Moreno-Noguer, F.; Jeni, L.A.; Torre, F.D.L. 3d human shape and pose from a single low-resolution image with self-supervised learning. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 284–300.
29. Wang, J.; Jin, S.; Liu, W.; Liu, W.; Qian, C.; Luo, P. When human pose estimation meets robustness: Adversarial algorithms and benchmarks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 11855–11864.
30. Shi, X.; Qin, P.; Zhu, J.; Xu, S.; Shi, W. Lower limb motion recognition method based on improved wavelet packet transform and unscented kalman neural network. *Math. Probl. Eng.* **2020**, *2020*, 5684812. [CrossRef] [PubMed]
31. Xue, Y.; Yu, Y.; Yin, K.; Li, P.; Xie, S.; Ju, Z. Human In-hand Motion Recognition Based on Multi-modal Perception Information Fusion. *IEEE Sens. J.* **2022**. [CrossRef]
32. Li, Q.; Liu, Y.; Zhu, J.; Chen, Z.; Liu, L.; Yang, S.; Zhu, G.; Zhu, B.; Li, J.; Jin, R.; et al. Upper-Limb Motion Recognition Based on Hybrid Feature Selection: Algorithm Development and Validation. *JMIR mHealth uHealth* **2021**, *9*, e24402. [CrossRef] [PubMed]
33. Guo, R.; Cui, J.; Zhao, W.; Li, S.; Hao, A. Hand-by-hand mentor: An AR based training system for piano performance. In Proceedings of the 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), Lisbon, Portugal, 27 March–1 April 2021; pp. 436–437.