Gesture Scoring Based on Gaussian Distance-Improved DTW

A. Xiwen Chen[1], B. Weiwei Yang[2], Bing Lu[1,*], D. Gaoning Nie[1], E. Miao Jin[1], F. Xu Wang[1]

[1]China Electric Power Research Institute Ltd.,
No. 143, Luoyu Road, 430074, Wuhan, Hubei Province, China
[2]State Grid Corporation Limited Technical College Branch,
No. 500, Second Ring South Road, 250002, Jinan, Shandong Province, China
*1219279442@qq.com; cug12989@163.com; *cugwhly@163.com; cug12989@sina.com;*
cug20211@sohu.com;cug12989@yeah.net

Abstract—The power industry has been dedicated to applying virtual reality (VR) technology to build training systems in virtual environments, enabling personnel to complete skill training in real simulated environments while ensuring their safety. Conventional action scoring systems struggle to provide accurate scores for fine movements. Accurate scoring of fine movements can help workers identify their shortcomings during power operations, thus improving learning efficiency. This is of great significance for training on virtual environment-based power operation. This paper proposes a power operation-orientated VR action evaluation method based on the Gaussian distance-improved dynamic time warping (DTW) algorithm and the temporal convolutional network (TCN) model. First, the adaptive adapter is used to extract one-dimensional features from the three-dimensional data of the data gloves. Then, based on the TCN model, action data with significant discrepancies are filtered out. Finally, the obtained data are input into the Gaussian distance-improved DTW algorithm, where the path size is calculated. Corresponding scoring criteria are established on the basis of the path size to evaluate the actions. The results demonstrate that the VR action evaluation method based on the Gaussian distance-improved DTW algorithm and the TCN model significantly improves the accuracy of evaluating fine movements compared to traditional evaluation algorithms.

***Index Terms*—VR; TCN; DTW algorithm; Power grid operations.**

I. INTRODUCTION

The normal operation of power systems plays a crucial role in ensuring social stability. Achieving technological breakthroughs in the maintenance and calibration of electrical equipment under live conditions is particularly important. However, there are inherent safety risks during live operations. To provide workers with safe training and learning opportunities, virtual reality (VR) technology has been introduced to build training systems in virtual environments. The conduct of various hazardous training activities in virtual environments has become a major trend, as demonstrated by the use of virtual environments in the medical field for the teaching of surgeries [1]. Combining virtual environments with training has significant implications for the safety of training in hazardous environments.

To achieve faster and more efficient scoring of fine hand movements, it is necessary to amplify the features of small hand movements and eliminate significantly erroneous actions. VR technology and data gloves can accurately capture the coordinate displacement of various hand nodes. The Gaussian distance-improved dynamic time warping (DTW) algorithm can amplify the data features of small hand movements, while the temporal convolutional network (TCN) algorithm can quickly filter out significantly erroneous actions. This paper combines VR technology with the theoretical foundations of the Gaussian distance-improved DTW algorithm to evaluate actions by extracting coordinate displacement from data gloves and applying feature extraction through an adaptive adapter.

Power grid operations involve numerous delicate operations, and improper execution of fine and subtle movements may lead to a series of safety accidents. Traditional action evaluation methods often have limitations in evaluating fine movements and distinguishing between correct and incorrect execution. Therefore, it is crucial to have a reasonable evaluation of fine hand movements in the virtual reality safety training system for power grids. This aspect is of great significance for training of the power grid.

The main contributions of this paper are as follows.

1. Proposing the use of data gloves to capture the coordinate displacement of the hands in a virtual environment and employing an autoencoder for feature extraction, reducing the three-dimensional data to one-dimensional data that meets the input requirements of the entire system.
2. Introducing the TCN algorithm for data classification, which assigns a zero score to actions with significant deviations, thus improving scoring efficiency.
3. Presenting a power grid operation-related action evaluation method based on the Gaussian distance-improved DTW algorithm, which amplifies the features of fine hand movements, thereby enhancing the efficiency and accuracy of evaluating fine hand movements in virtual reality environments.

Manuscript received 5 January, 2024; accepted 25 March, 2024.

This work was supported by the State Grid Corporation Headquarters Science and Technology project under Grant. No. 5700-202217206A-1-1-ZN.

The structure of this paper is as follows. Section I provides background information on power grid training projects, the application of VR technology in power grid training systems, and discusses how to improve the efficiency and accuracy of scoring fine hand movements. The contributions and organisation of the study are also introduced. Section II presents research findings in the fields of VR technology, action evaluation, the TCN algorithm, and the DTW algorithm. It summarises the unresolved issues and proposes the solutions presented in this paper. Section III describes the data processing methods for extracting data from data gloves, introduces the TCN algorithm and the DTW algorithm, and constructs the TCN-Gaussian Practice Temporal Convolutional Network (GPDTW) system model based on the TCN algorithm and the Gaussian distance-improved DTW algorithm. Section IV explains the hand movement data acquisition method using data gloves and analysing the experimental results. Section V summarises the entire paper, presenting the conclusion that the VR action evaluation method based on the TCN-GPDTW system model can be applied to evaluate fine hand movements in virtual environments. The conclusion validates the efficiency and accuracy improvements achieved by the TCN-GPDTW system model based on the TCN algorithm and Gaussian distance-improved DTW algorithm in virtual reality-based action evaluation.

II. LITERATURE REVIEW

The application of temporal convolutional neural networks (TCN) for prediction has been widely used in various fields. By combining time-series data with the characteristics of convolutional neural networks (CNN), TCN models are built to forecast future events or trends. These models emphasise the modelling of uncertainty and provide more comprehensive information for decision making through probabilistic predictions [2]. In [3], TCN is used to classify satellite image time series, improve the accuracy and efficiency of satellite image classification, and offer a new solution for satellite remote sensing applications. Real-time speech enhancement using TCN is explored in [4]. By applying convolutional operations to speech signals, the temporal features are captured, resulting in real-time improvement of speech quality and providing clearer and more intelligible speech output. TCN is utilised for action segmentation tasks in [5]. This approach transforms action segmentation problems into temporal data classification problems. Using TCN models, it learns the boundary and features of the action in temporal data, leading to more accurate and consistent results in action segmentation. In [6], a fatigue assessment method for drivers is proposed, introducing a spatio-temporal convolutional neural network (STCNN) based on electroencephalogram (EEG). By capturing the spatio-temporal characteristics of EEG signals, real-time detection and identification of driver fatigue states are achieved, improving the accuracy and reliability of driver fatigue assessment. A diagnostic method using TCN for laboratory tests is presented in [7]. Laboratory test data are transformed into temporal data, and TCN models are employed to extract key features and patterns. This method improves the accuracy and efficiency of diagnostics based on laboratory tests. The initialisation strategy of spatio-temporal CNNs is discussed in [8], proposing an adaptive initialisation strategy that dynamically selects the most suitable initialisation method based on the network structure and task characteristics. This strategy contributes to improving the performance and convergence efficiency of the model. In [9], a model called STCNN is introduced, with the aim of achieving long-term traffic prediction. By extracting features in the spatio-temporal dimension through convolutional operations, future traffic flow can be predicted. This research has a significant reference value in the field of traffic flow prediction. A method based on TCNs for human activity recognition is proposed in [10]. By learning temporal relationships, effective features are extracted from temporal data and used to classify different human activities. Lastly, a method based on TCNs for hourly heat load prediction is presented in [11]. Using temperature, humidity, and other features of time series data and employing convolutional operations to extract time-related features, a prediction model is established.

Action recognition has extensive applications in various fields, employing techniques such as traditional feature extraction and machine learning, as well as deep learning and neural networks [12]. In [13], a method is proposed to improve human action recognition using a score distribution and ranking. By analysing the score distribution and ranking information generated by the action recognition system, this method improves recognition accuracy. Through statistical analysis of score distribution and adjustment of ranking, different actions can be better distinguished, reducing misjudgments. In [14], a method is introduced for action recognition by learning deep multigranularity spatio-temporal video representations. By dividing the video into multiple spatio-temporal regions of granularity and performing deep learning representations on each region, the method captures action details and contextual information. By fusing and encoding the multigranularity representations, this approach better captures action features in the video, enabling accurate action recognition. A deep local video feature method for action recognition is proposed in [15]. It utilises CNNs for feature extraction from video and employs long short-term memory networks (LSTMs) for temporal sequence modelling. By combining local video patches with global context, this method captures spatio-temporal features of actions effectively. In [16], an approach is presented for skeleton-based action recognition using LSTM and CNN. This method represents human poses with skeleton data and uses CNN to extract spatial features from skeleton sequences. Then, LSTMs are applied to model the temporal aspect of the skeleton sequences. By combining CNN and LSTM, this method captures the spatio-temporal relationships in the skeleton action sequences, enabling accurate action recognition. In [17], a method for recognition of actions is proposed using dynamic image networks. This method represents actions by transforming video sequences into dynamic images and utilises CNNs to extract features from dynamic images. Dynamic images are generated by modelling the differences between adjacent frames in the temporal domain. Subsequently, the dynamic images are fed into the CNN for processing, extracting feature representations with temporal and spatial information. An algorithm for human action recognition is proposed in [18].

This algorithm is based on machine learning and pattern recognition techniques, aiming to automatically recognise human actions by analysing video or image sequences. The article may also involve performance evaluation and experimental results to validate the accuracy and effectiveness of the algorithm in action recognition tasks. In [19], a pose-based human action recognition method is presented using the extreme gradient boosting (XGBoost) algorithm. Human pose data are utilised to represent and analyse different actions. This pose-based approach combined with the XGBoost algorithm achieves good performance and accuracy on various action data sets. In [20], a YOLO-based framework is proposed to simultaneously recognise and localise human actions in videos. The method first employs the YOLO algorithm for object detection in video frames, which recognises object-bounding boxes that contain human actions. Then, action classification is performed on the detected bounding boxes, enabling recognition of different actions. By combining YOLO's object detection and action classification, this method achieves accurate human action recognition and localisation. In [21], a method is presented for real-time multiview human action recognition using a wireless camera network. A network of wireless cameras deployed at different locations captures video data from different perspectives. By fusing and synchronising the video streams from multiple cameras, multiview video data with rich perspective information can be obtained. Machine learning and pattern recognition techniques are then applied to analyse and process these multiview video data, enabling real-time recognition of human actions.

Dynamic time warping (DTW) is a technique used for comparing and matching time-series data. It compares two time series of different lengths or partial alignment by warping (aligning) the time axis to find the best match between them [22]. Adaptive constrained dynamic time warping (ACDTW) [23] introduces adaptive constraints to overcome the problem of fixed constraints in computing the similarity of time series with DTW. ACDTW adapts the constraints based on the characteristics of the data, providing a more flexible and accurate comparison of time series.

In [24], a time-weighted DTW method is proposed for time series data mining. It improves the comparison and matching accuracy of time series data by introducing time weighting to the DTW algorithm. DTW was used to analyse the correlation between the COVID-19 pandemic and the prices of energy commodities [25]. In [25], the authors provide evidence of the association between the pandemic and the prices of energy products by applying the DTW method. Discriminative differentiable dynamic time warping (D3tw) [26] is a method for weakly supervised action alignment and segmentation. It utilises DTW to precisely align and segment weakly supervised action data in a differentiable manner. In [27], an intelligent healthcare system is developed using speech recognition, support vector machine (SVM), and DTW. The system aims to achieve automation and intelligence in the medical field through speech recognition technology. In [28], a classification method is proposed based on object and time constraints using DTW for crop classification using Sentinel-2 satellite data. The method considers both object boundaries and time constraints in crop classification. In [29], an improved DTW algorithm called EventDTW is developed to align biomedical signals with nonuniform sampling frequencies. It addresses the issue of traditional DTW algorithms when dealing with signals with nonuniform sampling frequencies. A method for online signature verification [30] combines time series averaging and a locally stable weighted DTW algorithm. The objective is to improve the accuracy and robustness of online signature verification systems. In [31], a phenology-time weighted dynamic time warping (PTWDTW) method is proposed for the classification and mapping of winter wheat in northern China using Sentinel-2A/B data. The method aims to improve the accuracy of winter wheat monitoring and identification.

In summary, for grid-related training, a method is needed to evaluate fine-grained actions. This paper proposes an evaluation method for actions related to the grid based on the TCN-GPDTW model. First, hand coordinate displacements are extracted using data gloves. Then adaptive dimension reduction is applied to the data, followed by inputting the data into TCN and Gaussian DTW algorithms for scoring. The TCN algorithm significantly improves scoring efficiency, while the Gaussian DTW algorithm greatly enhances scoring accuracy.

III. System Model

Action scoring of human body movements is an active research area in computer vision with wide applications across various domains. However, traditional action scoring systems often evaluate movements based on skeletal data, which typically involve larger and more expansive actions. In the context of power grid operation training, we need to assess subtle hand movements of power workers, which often involve smaller amplitudes and minimal data variations. As a result, more precise algorithms are required to handle such data.

Therefore, this paper proposes a joint algorithm combining temporal convolutional neural networks (TCNN) and dynamic time warping (DTW) to address certain limitations of traditional action scoring methods and improve the accuracy of scoring fine-grained movements. First, hand motion data during the operation process are obtained using data gloves. The data are preprocessed and fed into a trained TCN model for the initial evaluation. The Gaussian distance is then used instead of the Euclidean distance in the DTW algorithm, and the selected data are passed through the Gaussian DTW algorithm for comparison with standard actions to complete the scoring process.

A. Algorithm Flow Refinement

This paper constructs a data set using data gloves to capture hand motion data and evaluates long-duration fine-grained hand movements based on the data set, allowing power grid operators to assess the compliance of their own actions. Unreal engine (UE) is an open and advanced real-time 3D creation tool that provides realistic visual effects and immersive experiences. The entire intelligent power grid training system is built using UE4. Through collaboration with industry professionals, UE4 is used along with three basic static functions and blueprints to collect professional hand operation motion data. The collected data are preprocessed and fed into a trained TCN model to determine

whether the actions performed are necessary for normal workflow, as opposed to significantly deviating actions. Actions with significant deviations are assigned a score of 0. For actions with minor deviations, the data are further preprocessed to match the format required for DTW. The processed data are then fed into the improved DTW algorithm, which compares it with the data of standard actions, measuring the distance between the generated paths. The improvement in DTW involves replacing the Euclidean distance with the Gaussian distance formula, which considers the weight of neighbourhood distances of data points, effectively handling noisy data and outliers, thus enhancing the accuracy of similarity measurement and noise resistance. Finally, the distance is converted into the corresponding scores. The refined flow chart of the joint algorithm for segmenting long-duration fine-grained hand movements is illustrated in Fig. 1.

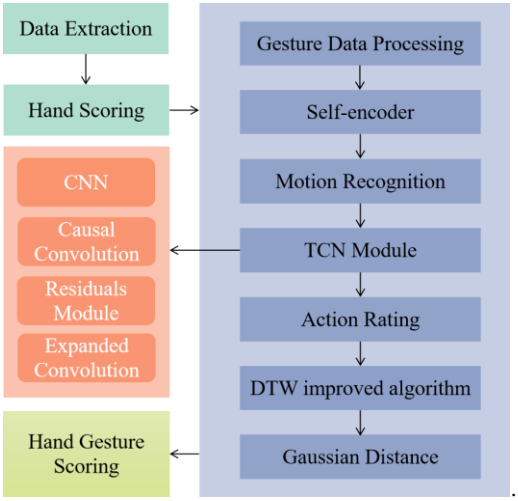

Fig. 1. Refined flow chart of the algorithm.

In VR-based power grid training using data gloves as the primary tool for human-computer interaction and data extraction, this paper aims to collect data through data gloves in a virtual reality environment and preprocess them in the format required by TCN. The data obtained from the data gloves are three-dimensional data representing the coordinates (X, Y, Z) of the hand joints. However, the TCN requires one-dimensional time series as input. Therefore, feature extraction is performed on the extracted data to achieve dimensionality reduction.

The data extracted from the data gloves are nonlinear, and deep learning neural networks are more advantageous in extracting features from nonlinear data. To effectively extract features from the data, this paper proposes using an autoencoder for feature extraction to optimise input vectors.

An autoencoder consists of an encoder and a decoder. The encoder transforms the input data into a low-dimensional encoding, while the decoder reconstructs the encoding back into the original data. Its structure is similar to that of a neural network, but with the unique property that the number of inputs and outputs are the same. In simple terms, it is a neural network model that trains itself with the objective of reproducing the input itself. The structure of an autoencoder is shown in Fig. 2.

A simple autoencoder consists of three layers: an input layer, a hidden layer, and an output layer. The input layer receives the original data, which are then processed by the hidden layer for feature extraction, and finally reconstructed by the output layer to restore the input data as the output data.

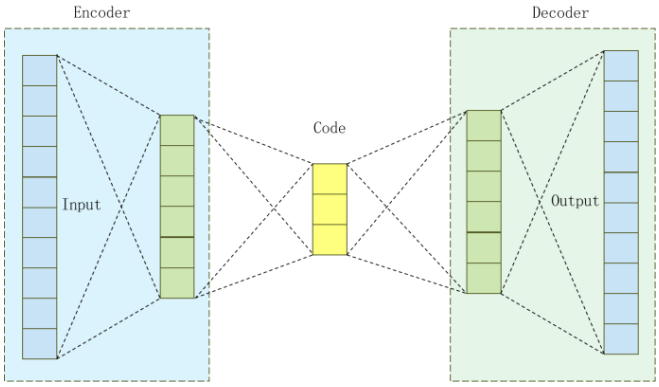

Fig. 2. Structure of autoencoder.

Between the input layer and the hidden layer, an encoder is introduced in the autoencoder. Its purpose is to compress the input data and reduce the dimensionality. Each node in the encoder corresponds to a node in the hidden layer. Through learning, the encoder compresses the input data into a set of feature vectors that represent the main characteristics of the original data. Between the hidden layer and the output layer, a decoder is introduced into the autoencoder. Its role is to decompress the feature vectors from the hidden layer and reconstruct them into output data. Each node in the decoder corresponds to a node in the hidden layer. Through learning, the decoder reconstructs the feature vectors from the hidden layer into output data while adjusting the weights to make the reconstructed output data as similar as possible to the original data. Now, let us provide a specific description of the operation process of the internal structure of the encoder.

First, a set of data are input into the input layer. The encoding process from the input layer to the hidden layer can be formulated as follows

$$y = s_x\left(w_x x + b_x\right). \tag{1}$$

In the equations provided, y represents the output results of the encoder, given by $y = [y_1, y_2, y_3, ..., y_m]$, x represents the input data $x = [x_1, x_2, x_3, ..., x_m]$, w_x represents the weight values from the input layer to the hidden layer, and b_x represents the bias values in the encoding process. Typically, the sigmoid function is used as the activation function, and the formula is as follows

$$s = \frac{1}{1+e^{-x}}. \tag{2}$$

The formula for the decoding process from the hidden layer to the output layer is as follows

$$x' = s_y\left(w_y y + b_y\right). \tag{3}$$

In the equation provided, s_y generally represents the activation function used in the output layer, which can be either the sigmoid function or the identity function, w_y represents the weight values from the hidden layer to the output layer, and b_y represents the bias values in the decoding

process.

The ideal situation for an autoencoder is to have the output results identical to the input data. To move closer to this goal, the auto-encoder needs to undergo multiple training iterations. The parameters that require training are weights and biases. The termination condition for training is determined by observing the magnitude of the error between the input and output, typically represented by the error function L(x, x'). The formula is as follows

$$L(x,x') = -\frac{1}{n}\sum_{i=1}^{n}[x_i \ln y_i + (1-x_i)\ln(1-y_i)] + \\ +\frac{\lambda}{2}\sum_{l=1}^{n_l}\sum_{j=1}^{s^{(l)}}\sum_{i=1}^{s^{(l)}+1}(W_{ij}^{(l)})^2. \quad (4)$$

In the equation, the first term represents the cross term, and the second term represents the weight decay term, n_l represents the number of hidden layers, $s^{(l)}$ represents the number of parameters in the l^{th} layer, λ represents the weight decay parameter within the neural network, and $W_{ij}^{(l)}$ represents the weight connection matrix between the i^{th} neuron in the l^{th} layer and the j^{th} neuron in the next layer.

The main purpose of the second term in (4) is to control the weight fluctuations between layers in the network to prevent overfitting of the data. When the error function L(x, x´) reaches its minimum value, it indicates the end of training, and the auto-encoder has completed its learning task.

B. Temporal Convolutional Networks

When workers perform electrical operations in a virtual environment, inexperienced workers may take incorrect actions. In this paper, TCN is introduced for classification to distinguish between erroneous actions and correct but nonstandard actions, thereby accelerating the scoring speed of the entire system. Compared to other neural networks, TCN employs techniques such as residual connections and batch normalisation, making the model more stable and avoiding the issues of vanishing and exploding gradients. TCN has achieved comparable or even better performance than recurrent neural networks (RNNs) in many time-series tasks, while avoiding the temporal dependencies present in RNNs, thus enabling more efficient parallel computation. The core of TCN is the one-dimensional convolutional layer, which is used for feature extraction and modelling of the input time series. Unlike traditional RNNs and LSTMs, TCN adopts a convolution-based approach to capture long-term dependencies in time series. Specifically, TCN utilises a series of one-dimensional convolutional layers to perform convolution operations on the time series and captures features at different time scales through different kernel sizes. These convolutional layers do not have any recurrent connections, eliminating the need to maintain complex state information and leading to faster training and prediction speeds.

In addition to the one-dimensional convolutional layer, TCN introduces two important techniques: residual connections and dilated convolutions. Residual connections are a highly effective technique used to address vanishing and exploding gradient problems in deep neural networks. In TCN, residual connections are used to connect adjacent convolutional layers, allowing the model to better capture long-term dependencies. Specifically, residual connections apply convolutional operations to the input time series through a convolutional layer and an identity mapping, and then sum their outputs as the input to the next layer. The computation formula for a residual block is as follows

$$y = f(x) + x. \quad (5)$$

In the TCN, the residual block aims to learn the difference between the input feature vector, denoted x, and the output feature vector, denoted y. This allows for better capture of feature information and reduces the loss of information.

Dilated convolution is a technique that allows the expansion of the size of the convolutional kernel, thereby increasing the receptive field of the model to capture longer temporal dependencies. In TCN, dilated convolution is applied in the final layer of the convolutional stack by increasing the stride of the convolutional kernel, effectively enlarging its receptive field. The computation formula for dilated convolution is as follows

$$y_i = \sum_{j=1}^{k} \omega_j x_{i-d\cdot(j-1)} + b. \quad (6)$$

In the equation, y_i represents the output value of the convolution operation, $x_{i-d\times(j-1)}$ represents the position of the input data in the convolutional kernel, ω_j represents the weights of the convolutional kernel, and b represents the bias. The parameter d is known as the dilation rate, which controls the receptive field size of the convolutional kernel. When d = 1, the dilated convolution is reduced to a standard convolution operation. The TCN structure is illustrated in Fig. 3.

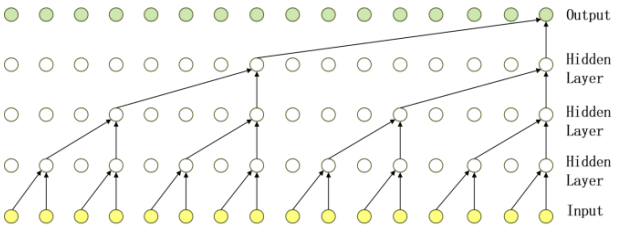

Fig. 3. The structure of the time series convolutional model.

The kernel size is set to 2, which means that the input of each layer consists of the output of the previous layer in two consecutive time steps. The dilations are set as [1, 2, 4, 8], indicating the interval between the input time steps for each layer. With a dilation of 4, the output from the previous layer at every fourth time step is taken as the input for the current layer, until enough input is collected to match the kernel size.

C. Improving DTW-Based Action Evaluation with Gaussian Distance

In this study, DTW is chosen to measure the similarity between two actions, as it is a suitable algorithm for handling time series data. In action evaluation, an action sequence can be viewed as a time series, where the state at each time step (such as body posture, joint angles, etc.) serves as an element of the sequence. When DTW matching is performed on the two action sequences, their similarity can be measured, resulting in a score.

In the context of power operations, it is necessary to effectively rate the hand movements of workers. Traditional DTW algorithms only consider the shape information of time series, while neglecting the differences between different features within the time series. In power operations, hand movement data tend to have small variations and indistinct features. Additionally, noise and outliers may exist in the hand movement sequences due to environmental and equipment factors. In the gesture scoring system for the power grid, traditional DTW fails to handle these issues effectively, leading to inaccurate scoring.

To address these challenges, this study proposes an improved DTW algorithm based on Gaussian distance. In the DTW algorithm, a distance metric must be defined to compute the distance between two action sequences. The Euclidean distance, also known as the Euclidean metric, measures the straight-line distance between two points in two- or three-dimensional space. Its formula is

$$d(x,y)=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2+...+(x_n-y_n)^2}, \quad (7)$$

where $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., x_n)$ represent two n-dimensional vectors, where n denotes the dimensionality of the vectors.

Manhattan distance, also known as the city block distance or taxicab distance, refers to the distance between two points in two- or three-dimensional space, measured by the sum of the absolute differences of their coordinates. Its formula is

$$d(x,y)=\sum_{i=1}^{n}|x_1-y_1|, \quad (8)$$

where $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ represent two n-dimensional vectors, where n denotes the dimensionality of the vectors. In practical applications, the Euclidean distance is commonly used to compute the distance for continuous data, while the Manhattan distance is typically used to compute the distance for categorical data.

In this study, we propose the use of Gaussian distance as the distance metric function in DTW. Gaussian distance is a nonlinear distance metric approach that considers the relationships between different features within a time series, allowing for a better capture of similarity in the time series. Its computation formula is

$$d(x,y)=\sqrt{\sum_{i=1}^{n}\omega_i(x_i-y_i)^2}, \quad (9)$$

where $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ represent two n-dimensional vectors, where n denotes the dimensionality of the vectors, q is a weight parameter used to adjust the importance of different features. When q takes a smaller value, it indicates that the corresponding feature contributes less to the distance. On the contrary, when q takes a larger value, it indicates that the corresponding feature has a greater impact on the distance.

Compared to the traditional DTW algorithm, the improved DTW algorithm based on Gaussian distance takes into account the differences between different features in the time series. Furthermore, it allows one to adjust the weight parameter to control the relative importance of different features. This enables a more accurate assessment of similarity between time series and reduces the impact of noise and outliers on the evaluation results.

D. Grid Gesture Action Scoring Based on TCN-GPDTW

The improved DTW algorithm is applied to the gesture scoring system for the power grid, addressing the challenges of small variations and indistinct features in hand movements during power operations. It also helps mitigate the impact of noise and outliers in the hand movement sequences.

After the classification using TCN, zero scores are assigned to incorrect operation flows, but there may still be issues such as nonstandard actions and inadequate details in the selected actions. Power operations involve significant risks and even minor issues can pose a serious threat to worker safety. It is crucial for workers to follow proper procedures when learning power operation flows in virtual environments. Therefore, this study uses GPDTW to score worker operation flow, and a TCN-GPDTW-based grid gesture action scoring system is designed to evaluate virtual actions in VR environments.

The TCN-GPDTW system model for action scoring involves the following steps.

1. Data extraction using data gloves, removing unnecessary data.
2. Applying the extracted 3D data from VR gloves to an autoencoder for feature extraction and dimensionality reduction.
3. Feeding the resulting one-dimensional time series into TCN for classification, assigning zero scores to actions with low recognition rates.
4. Applying the remaining action data to the GPDTW model for scoring.
5. Summarising all action scores and obtaining an overall score based on a certain weighting.

IV. APPLICATION EXAMPLES

There are numerous actions involved in power operations. In this study, we selected five actions from the power operation process for scoring experiments. The power operation process consists of 24 scenarios, including high-risk actions such as grounding, discharge, insulation mat laying, wire wrapping with insulation tape, and high-altitude wire connection. Accurate scoring is crucial for evaluating such actions, enabling workers to clearly identify the correctness of their own behaviour. An accurate scoring system is of the utmost importance for workers to learn power operation procedures effectively.

A. Action Data Acquisition

The acquisition of action data is performed by professional power grid workers who operate in the training system. A set of relatively standardised actions is selected as the evaluation criterion. Safety operation training consists of 24 scenarios with multiple actions. In this example, five actions were chosen for evaluation. To obtain a standardised multiscale hand gesture data set, 20 skilled operators were assigned to simulate a series of power grid operations, including wearing safety helmets, wearing insulated gloves, electrical testing, opening electrical box doors, and screwing screws, within a virtual environment. OpenCV technology was used to

analyse and process the action frames captured in the UE4 engine, extracting hand gesture information. Samples were preprocessed to extract action features. The distribution of each action in the data set is shown in Table I.

TABLE I. DISTRIBUTION OF ACTIONS IN THE MULTISCALE HAND DATA SET.

Action Name	Quantities
Wear a safety helmet	90
Wear insulated gloves	70
A power test	67
Open the door to the electrical box	98
Screw	67

The virtual glove consists of 27 nodes, including 24 finger joints, one wrist joint, one hand centre node, and one node representing the entire hand model. Each data frame contains the coordinate offsets, rotation angles, and scaling factors for each hand joint in the virtual space. In this study, the data were preprocessed and only coordinate offset data were used in the action scoring system. Therefore, the rotation angles and scaling factors were excluded during the data preprocessing stage. The coordinate offset data are represented by three-dimensional (x, y, z) values. However, the temporal convolutional network (TCN) used for action evaluation requires one-dimensional data for scoring. To address this, an autoencoder was used to perform dimensionality reduction on the three-dimensional coordinate offset data.

B. Example of Action Evaluation for Standard Electrical Testing

The operations studied in this paper for electrical grid work include wearing a safety helmet, wearing insulated gloves, electrical testing, opening the electrical cabinet door, and screwing screws. The operation of opening the cabinet door in a VR environment is illustrated in Fig. 4.

Fig. 4. Opening cabinet door operation in VR environment.

Based on the analysis and summary of power operation training combined with the operation manual, experts believe that when performing manual movements, the characteristics of operational motion can be considered static or within a certain range of motion. For example, when the hand maintains horizontal movement, the coordinates of all the joints of the hand will change. If there is a simultaneous change in the coordinates of all joints along one axis while the changes along the other axes are minimal, it can still be considered as a motion that conforms to the operational variation. Using 70 % of the data as the training set and 30 % of the data as the prediction set, TCN and CNN were used for the prediction. The results showed that TCN significantly outperformed CNN. Figure 5 illustrates the comparison between TCN and CNN in terms of training loss and prediction accuracy.

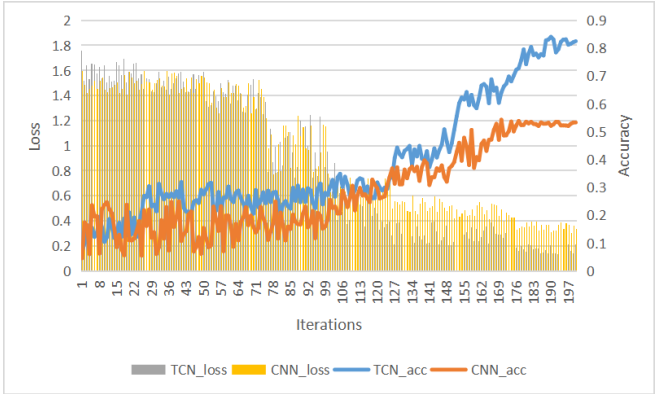

Fig. 5. Comparison of TCN and CNN loss and accuracy.

In Fig. 5, the horizontal axis represents the number of training iterations, which refers to the number of times the model is trained on the training data set. The vertical axis represents the training accuracy, which indicates the accuracy of predictions made on the validation data set.

A TCN model was constructed based on actual actions. Actions to be classified were extracted and dimensionality reduced to generate one-dimensional vectors, which were then input into the TCN model for classification. Data with accuracy greater than 50 % were selected and fed into an improved dynamic time warping (DTW) algorithm for scoring. The data for standard actions and the selected action data were compared using the improved DTW algorithm.

Figure 6 displays a line graph of Gaussian DTW between nonstandard actions and standard actions. Using (5), the distance between each data point and point can be calculated, followed by path optimisation to obtain the final distance between the two sequences.The horizontal axis represents the moments in time at which the actions occur, in seconds. The vertical axis corresponds to the values obtained by reducing the dimensionality of the collected action data.

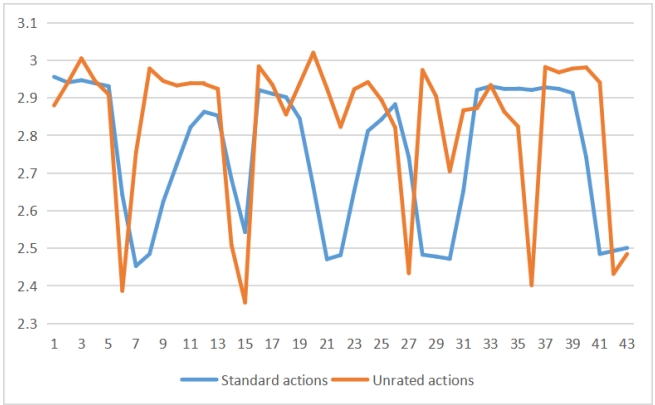

Fig. 6. Standard actions and actions to be scored.

C. Construction of an Action Evaluation Formula

After obtaining the distance between the standard action

and the action to be evaluated, it is necessary to construct a suitable action evaluation formula to score the standardness of the action. The allocation of influence factors affects the rationality of the action evaluation formula. Different influence factors should be set for different key joints based on the impact of different actions. For example, when performing a screwing action, a larger influence factor should be assigned to the thumb joint, while for a hat-wearing action, a larger influence factor should be assigned to the wrist joint.

After determining the influence factors, the scoring formula was established as follows:

$$\mathrm{Esk} = \left(1 - \mathrm{dist} * \mathrm{Fk}\right) * \mathrm{s} / 27, \left(\mathrm{k} = 1, 2, 3 \ldots 14\right), \quad (9)$$

$$\mathrm{score} = \mathrm{Es1} + \mathrm{Es2} + \mathrm{Es3} + \ldots + \mathrm{Es27}, \quad (10)$$

where k represents the number of joints, ranging from 1 to 27. s represents the maximum score achievable for an action, which is 100. Since there are 27 joints, the maximum score for each joint is s/27. dist denotes the accumulated distance value obtained from the calculation of the shortest path of the DTW algorithm, Fk represents the influence factor of each joint, Esk represents the score of each joint, and the score indicates the final score of the test action, which is the sum of the scores for the 27 joint angles.

D. Experimental

To verify the reliability and feasibility of the TCN-GPDTW system model proposed in this paper, the evaluation scores of experts and the action evaluation scores based on dynamic time warping (DTW) were set as control experiments. The entire grid virtual reality safety training system was used as the experimental object, and the actions performed by the experimental participants were evaluated in a laboratory setting. By recording the evaluation scores of three methods, the feasibility of the proposed method was determined.

The conventional DTW algorithm, based on the dynamic programming concept, is commonly used to assess the similarity between temporal data and has been widely applied in audio processing. It has also been utilised for action evaluation. By reducing the coordinates of the wrist joint of the standard action and the test action, the algorithm calculates the shortest distance between the two sequences to determine the similarity between them, thus completing the evaluation of the action.

Expert scores, DTW-based action evaluation, and the grid-related action evaluation method of the TCN-GPDTW model were used to evaluate five types of actions: wearing a safety helmet, wearing insulated gloves, conducting electricity testing, opening an electrical box door, and screwing a screw. Each type of action was evaluated three times and the evaluation results are shown in Table II.

TABLE II. ACTION DISTRIBUTION IN MULTISCALE HAND DATA SET.

Evaluation Method / Action	Expert Rating	DTW-Based Action Evaluation	TCN-GPDTW
Wearing Safety Helmet 2	95.00	85.22	91.39
Wearing Safety Helmet 3	81.00	81.54	83.63
Wearing Insulated Gloves 1	97.00	96.28	98.75
Wearing Insulated Gloves 2	95.32	95.26	91.96
Wearing Insulated Gloves 3	96.00	98.35	95.87
Voltage Testing 1	97.00	97.85	90.25
Voltage Testing 2	53.00	97.26	73.25
Voltage Testing 3	92.00	97.61	86.46
Opening Electrical Box Door 1	100.00	98.58	99.68
Opening Electrical Box Door 2	99.00	99.65	98.55
Opening Electrical Box Door 3	99.00	99.24	98.94
Screwing Screw 1	93.00	98.32	92.23
Screwing Screw 2	94.00	97.33	92.33
Screwing Screw 3	55.00	98.56	78.24

As shown in Table II, expert ratings exhibit a certain subjectivity, with more consistent scores for simpler and nonhazardous actions, all scoring above 90 points. Expert ratings are more detailed and stringent to assess risky actions. In cases of errors or improper execution during hazardous actions, scores can fall below 60 points, rendering them "unqualified". However, relying solely on one-on-one expert instruction and evaluation incurs substantial human and material resources, significantly hampering training efficiency.

DTW-based action evaluation proves effective for assessing large-scale actions, but it does not accurately represent the correctness of intricate actions. When confronted with finer actions such as voltage testing and screwing, minor errors often go unnoticed, resulting in roughly similar scores. Moreover, DTW-based evaluation is susceptible to irrelevant actions and fails to assess ordered sequences, such as the insulated glove procedure, where a low score is assigned if the right glove is donned first, unlike the standard left glove first.

The approach of the TCN-GPDTW model to evaluating actions related to the power grid significantly addresses the inefficiencies of expert ratings. It remains unaffected by irrelevant manoeuvres, enabling comprehensive evaluations of actions on any scale. It excels in detecting errors in delicate actions. This highlights the applicability of the TCN-GPDTW model's power grid-related action assessment methodology to power grid virtual reality safety training systems. Furthermore, the evaluation scores for this approach show validity and objectivity.

V. DISCUSSION

Currently, most of the research focusses on evaluating the stability and communication efficiency of smart grid systems, with limited research on the precise, long-term assessment of fine hand movements within smart grid training systems. Research that specifically targets evaluations of actions related to power grid operations remains scarce. In this paper, a methodology for assessing power grid-related actions based on the TCN-GPDTW system model is proposed to assess intricate actions during power grid operations. By using the Gaussian DTW algorithm for action scoring, it becomes more pronounced in evaluating differences between fine action characteristics. This system model significantly enhances the accuracy of evaluating delicate actions compared to traditional assessment models.

Most existing action evaluation models primarily rely on action recognition accuracy and loss functions as their main criteria, often falling short in accurately describing fine action features and neglecting minor differences among similar actions. This paper introduces a Gaussian DTW algorithm that considers differences between distinct features in time sequences. It allows the adjustment of weight parameters to control the relative importance of different features. Feature extraction is performed using an autoencoder to optimise input vectors. The data obtained are then compared within the TCN-GPDTW system model to derive an evaluation grade for the hand action, and the feasibility of this method is verified through experimental validation. In contrast to the existing literature, this paper offers specific scoring for fine operations in power grid tasks, surpassing the performance of traditional evaluation systems.

The TCN-GPDTW model proposed in this paper theoretically applies to a wide range of action evaluations, but requires one-dimensional vector inputs for action data. In the process of extraction of feature from action data, issues might arise where action characteristics are not distinct or where dimensionality reduction fails to adequately represent original data features, occasionally resulting in inaccuracies in evaluation. Enabling the input of high-dimensional data would allow the system to more accurately describe action features and enhance the accuracy of the evaluation. Our team will continue to work towards addressing this aspect.

VI. CONCLUSIONS

To evaluate actions related to the power grid, this study introduces a methodology to assess actions related to the power grid based on the TCN-GPDTW model. Initially, an autoencoder is employed to extract features from glove-mounted data, transforming three-dimensional coordinate data into one-dimensional sequences. These sequences are input into a pre-trained TCN model. Data with significant deviations are assigned a zero score, whereas the remaining data are subjected to the Gaussian DTW algorithm for comparison with standard action data. This process derives an evaluation grade for the action of the hand, yielding results characterised by objectivity and effectiveness.

In application instances, this paper evaluates five categories of actions related to the power grid in live power grid operations: wearing safety helmets, donning insulated gloves, voltage testing, opening electrical box doors, and screwing screws. Comparative analysis with expert ratings and DTW-based action evaluations reveal that the TCN-GPDTW model-based methodology yields more objective results than expert ratings and effectively highlights fine action characteristics across different scales compared to DTW-based evaluations. Therefore, the TCN-GPDTW model-based power grid-related action assessment methodology can be applied to the assessment of fine actions within power grid operations.

For future improvement of the DTW algorithm, deeper exploration can be conducted in its data processing section, allowing the Gaussian-enhanced DTW algorithm to handle multidimensional data, thereby avoiding feature degradation during dimensionality reduction. This approach will be applied experimentally to the motion scoring within intelligent power grid training systems, contributing to the further enhancement of such systems.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] J. Falah *et al.*, "Virtual Reality medical training system for anatomy education", in *Proc. of 2014 Science and Information Conference*, 2014, pp. 752–758. DOI: 10.1109/SAI.2014.6918271.

[2] Y. Chen, Y. Kang, Y. Chen, and Z. Wang, "Probabilistic forecasting with temporal convolutional neural network", *Neurocomputing*, vol. 399, pp. 491–501, 2020. DOI: 10.1016/j.neucom.2020.03.011.

[3] C. Pelletier, G. I. Webb, and F. Petitjean, "Temporal Convolutional Neural Network for the classification of Satellite Image Time series", *Remote Sensing*, vol. 11, no. 5, p. 523, 2019. DOI: 10.3390/rs11050523.

[4] A. Pandey and D. Wang, "TCNN: Temporal convolutional neural network for real-time speech enhancement in the time domain", in *Proc. of ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6875–6879. DOI: 10.1109/ICASSP.2019.8683634.

[5] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal Convolutional Networks: A unified approach to action segmentation", in *Computer Vision – ECCV 2016 Workshops. ECCV 2016. Lecture Notes in Computer Science()*, vol. 9915. Springer, Cham, 2016, pp. 47–54. DOI: 10.1007/978-3-319-49409-8_7.

[6] Z. Gao *et al.*, "EEG-based spatio-temporal convolutional neural network for driver fatigue evaluation", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2755–2763, 2019. DOI: 10.1109/TNNLS.2018.2886414.

[7] N. Razavian and D. Sontag, "Temporal convolutional neural networks for diagnosis from lab tests", 2015. arXiv: 1511.07938.

[8] E. Mansimov, N. Srivastava, and R. Salakhutdinov, "Initialization strategies of Spatio-Temporal Convolutional Neural Networks", 2015. arXiv: 1503.07274.

[9] Z. He, C.-Y. Chow, and J.-D. Zhang, "STCNN: A Spatio-Temporal Convolutional Neural Network for long-term traffic prediction", in *Proc. of 2019 20th IEEE International Conference on Mobile Data Management (MDM)*, 2019, pp. 226–233. DOI: 10.1109/MDM.2019.00-53.

[10] Y. A. Andrade-Ambriz, S. Ledesma, M. A. Ibarra-Manzano, M. I. Oros-Flores, and D.-L. Almanza-Ojeda, "Human activity recognition using temporal convolutional neural network architecture", *Expert Systems with Applications*, vol. 191, art. 116287, 2022. DOI: 10.1016/j.eswa.2021.116287.

[11] J. Song, G. Xue, X. Pan, Y. Ma, and H. Li, "Hourly heat load prediction model based on temporal convolutional neural network", *IEEE Access*, vol. 8, pp. 16726–16741, 2020. DOI: 10.1109/ACCESS.2020.2968536.

[12] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey", *Image and Vision Computing*, vol. 60, pp. 4–21, 2017. DOI: 10.1016/j.imavis.2017.01.010.

[13] M. Hoai and A. Zisserman, "Improving human action recognition using score distribution and ranking", in *Computer Vision – ACCV 2014. ACCV 2014. Lecture Notes in Computer Science()*, vol. 9007. Springer, Cham, 2015. DOI: 10.1007/978-3-319-16814-2_1.

[14] Q. Li, Z. Qiu, T. Yao, T. Mei, Y. Rui, and J. Luo, "Action recognition by learning deep multi-granular spatio-temporal video representation", in *Proc. of the 2016 ACM on International Conference on Multimedia Retrieval*, 2016, pp. 159–166. DOI: 10.1145/2911996.2912001.

[15] Z. Lan, Y. Zhu, A. G. Hauptmann, and S. Newsam, "Deep local video feature for action recognition", in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 1219–1225. DOI: 10.1109/CVPRW.2017.161.

[16] C. Li, P. Wang, S. Wang, Y. Hou, and W. Li, "Skeleton-based action recognition using LSTM and CNN", in *Proc. of 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2017, pp. 585–590. DOI: 10.1109/ICMEW.2017.8026287.

[17] H. Bilen, B. Fernando, E. Gavves, and A. Vedaldi, "Action recognition with dynamic image networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2799–2813, 2018. DOI: 10.1109/TPAMI.2017.2769085.

[18] S. P. Sahoo and S. Ari, "On an algorithm for human action recognition", *Expert Systems with Applications*, vol. 115, pp. 524–534, 2019. DOI: 10.1016/j.eswa.2018.08.014.

[19] V. Ayumi, "Pose-based human action recognition with Extreme Gradient Boosting", in *Proc. of 2016 IEEE Student Conference on*

Research and Development (SCOReD), 2016, pp. 1–5. DOI: 10.1109/SCORED.2016.7810099.

[20] S. Shinde, A. Kothari, and V. Gupta, "YOLO based human action recognition and localization", *Procedia Computer Science*, vol. 133, pp. 831–838, 2018. DOI: 10.1016/j.procs.2018.07.112.

[21] S. Ramagiri, R. Kavi, and V. Kulathumani, "Real-time multi-view human action recognition using a wireless camera network", in *Proc. of 2011 Fifth ACM/IEEE International Conference on Distributed Smart Cameras*, 2011, pp. 1–6. DOI: 10.1109/ICDSC.2011.6042901.

[22] M. Müller, "Dynamic time warping", in *Information Retrieval for Music and Motion*. Springer, Berlin, Heidelberg, 2007, pp. 69–84. DOI: 10.1007/978-3-540-74048-3_4.

[23] H. Li, J. Liu, Z. Yang, R. W. Liu, K. Wu, and Y. Wan, "Adaptively constrained dynamic time warping for time series classification and clustering", *Information Sciences*, vol. 534, pp. 97–116, 2020. DOI: 10.1016/j.ins.2020.04.009.

[24] H. Li, "Time works well: Dynamic time warping based on time weighting for time series data mining", *Information Sciences*, vol. 547, pp. 592–608, 2021. DOI: 10.1016/j.ins.2020.08.089.

[25] K. Dmytrów, J. Landmesser, and B. Bieszk-Stolorz, "The connections between COVID-19 and the energy commodities prices: Evidence through the Dynamic Time Warping method", *Energies*, vol. 14, no. 13, p. 4024, 2021. DOI: 10.3390/en14134024.

[26] C.-Y. Chang, D.-A. Huang, Y. Sui, L. Fei-Fei, and J. C. Niebles, "D3tw: Discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation", in *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3546–3550. DOI: 10.1109/CVPR.2019.00366.

[27] A. Ismail, S. Abdlerazek, and I. M. El-Henawy, "Development of smart healthcare system based on speech recognition using support vector machine and dynamic time warping", *Sustainability*, vol. 12, no. 6, p. 2403, 2020. DOI: 10.3390/su12062403.

[28] O. Csillik, M. Belgiu, G. P. Asner, and M. Kelly, "Object-based time-constrained dynamic time warping classification of crops using Sentinel-2", *Remote sensing*, vol. 11, no. 10, p. 1257, 2019. DOI: 10.3390/rs11101257.

[29] Y. Jiang *et al.*, "EventDTW: An improved dynamic time warping algorithm for aligning biomedical signals of nonuniform sampling frequencies", *Sensors*, vol. 20, no. 9, p. 2700, 2020. DOI: 10.3390/s20092700.

[30] M. Okawa, "Time-series averaging and local stability-weighted dynamic time warping for online signature verification", *Pattern Recognition*, vol. 112, art. 107699, 2021. DOI: 10.1016/j.patcog.2020.107699.

[31] Q. Dong *et al.*, "Mapping winter wheat in North China using Sentinel 2A/B data: A method based on phenology-time weighted dynamic time warping", *Remote Sensing*, vol. 12, no. 8, p. 1274, 2020. DOI: 10.3390/rs12081274.