Human-Computer Interaction Based on ASGCN Displacement Graph Neural Networks

Yiping Yang[1*], Jijun Liu[2], Liang Zhao[1], Yuchen Yin[1]
[1]China State Shipbuilding Corporation Limited No.723 Research Institute, Yangzhou, 225001, China
[2]School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, 430074, China
E-mail: yyp1982to723@163.com
[*]Corresponding author

Keywords: ASGCN algorithm, Human-computer interaction, Long and short-term memory algorithm, Joint features, Action recognition, Encoder

Received: March 29, 2024

Intelligent terminal devices have become a popular theme for research in recent years, but the development of intelligent terminals cannot be separated from high-quality human-computer interaction models. Behavioral action recognition is one of the main ways to realize human-computer interaction, but the current action recognition model still exists with obvious time delay and low recognition accuracy. In light of this, the study built an intelligent human action capture and recognition model using an action structured graph convolutional network in conjunction with an encoder-decoder architecture, long and short-term memory algorithms, and controlled experiments to assess the model's performance. The outcomes indicated that the loss of the proposed model after convergence on the test dataset was 0.56%, while the average accuracy was 95.39%, and both performances outperformed the control experiment. In the meantime, the suggested model's average F1 score was 89.79%, which was 11.13% and 3.82% higher than that of the experiment's control model. The suggested model exhibits some improvement in the accuracy and F1 score of action recognition, according to the experimental findings. Therefore, the research of the suggested behavior recognition model has practical value. Additionally, in the real scene behavior recognition detection experiments, the proposed model validates the viability of the model with higher accuracy and reduced delay.

Povzetek: Prispevek predstavi izboljšan model za prepoznavanje človeških akcij s pomočjo ASGCN in LSTM algoritmov za natančnejšo in hitrejšo interakcijo človek-računalnik.

1 Introduction

Human-computer interaction (HCI) usually relies on gesture recognition, speech recognition and action recognition (AR), etc. Speech recognition is very mature in current development, and there are quite a number of intelligent models that can realize the needs of daily HCI [1]. However, to realize more intelligent HCI, it is necessary to solve the algorithm's ability to understand the combination of action feature capture. The current mainstream motion capture algorithms include a series of machine learning algorithms such as convolutional neural network (CNN), graph convolution network (GCN), deep neural network (DNN), and so on, among which the effect of image and video processing is better than the GCN algorithm. Better is the GCN algorithm [2-3]. However, the traditional GCN algorithm still has obvious shortcomings. Shallow GCN cannot transfer labels from a limited amount of training data to the whole graph structure, and the semi-supervised performance is poor. Deep GCN will have excessive smoothing problems, and it is difficult to distinguish the features of the nodes [4]. An abstract idea of a deep learning model is the encoder-decoder (ED) architecture. An ED structure may compress a lot of data, which cuts down on processing time and space while increasing transmission and storage efficiency [5]. The advantages of the ED architecture are especially obvious when processing large files such as images, videos, and audios. In view of this, therefore, the study selects aspect-specific graph convonlutional network (ASGCN) to be optimized and used in the construction of HCI model, and the ED architecture is used in long short-term memory (LSTM) as a way to optimize the ASGCN model. The innovation of the study is that it introduces an LSTM-based encoder structure that is utilized to capture specific movements of the human body. The article is structured into four sections. Related work, the first section, concentrates on the theoretical analysis that came before the research. The second part is the methodology, which performs HCI model construction through advanced techniques. The proposed model is put through performance testing tests in the third section, known as "model testing," in order to confirm its advanced nature. The fourth part is the

conclusion, which summarizes the research results and proposes future improvement directions.

2 Related works

HCI is important for the development of smart devices, so domestic and international researchers have explored for how to realize intelligent HCI. Chowdary et al. used deep learning techniques to recognize human emotions, thus promoting the intelligence of the model and HCI. The method eliminated the original fully connected layer of ConvNets and added a new fully connected layer with weights based on the number of instructions in the task. The study's findings demonstrated that the suggested emotion identification model can identify emotions with an average accuracy (AverA) of 96% [6]. Liu et al. conducted a related study on head pose estimation and optimized the technique for application in HCI. Liu et al. solved the problem of neighboring pose information processing and mislabeling gap in head pose estimation. The model was evaluated using an open-source dataset, and the study found that the suggested model performed noticeably better than other cutting-edge techniques, leading to improved outcomes for the optimization approach [7]. Zhang et al. constructed a glove-based HCI system using friction electric nanogenerators in order to realize the intelligence of wearable devices. The system was also used to extract and friction electric nanogenerator to analyze multidimensional signal features for gesture visualization and manipulator control functions. The study applied the proposed model to five object classification and recognition tasks. According to the experimental findings, the model performed the five tasks with an AverA of 98.7% [8]. Zhang et al. proposed a gesture recognition system called WiGesID in their study as they found that gesture recognition technology can advance HCI to some extent. The system employed Wi-Fi sensing and radar sensing techniques to enhance the security of the gesture recognition system and computer vision techniques to realize the dynamic patterns of gesture recognition. The findings indicated that the proposed system exhibits superior performance in cross-domain sensing, with enhanced recognition accuracy compared to state-of-the-art models [9].

GCN is a neural network designed to process images, but as the demand for image processing increases, the traditional GCN is difficult to meet the current needs, so many researchers have improved and optimized the GCN for GCN. Bessadok et al. provided a medical image recognition method based on learning depth graph neural network (GNN) structure. The method incorporated DNN and GNN. They used the method for the recognition of a comprehensive roadmap of neuronal activity in the human brain. According to the testing data, the suggested approach performs better and can obtain recognition accuracy of above 90% [10]. Wu et al. proposed a GCN-based natural language processing model, a taxonomy that systematically organizes existing GNN research on natural language processing along three axes. In addition, the method introduced ED techniques to achieve global encoding of input data. It was experimentally concluded that the proposed model possesses high accuracy and recall in natural language processing and classification, thus the proposed model is feasible [11]. Zhu et al. presented a GCN and DNN-based picture analysis model to address the significant unsupervised graph problem. The model implemented the recovery of cluster structure by DNN improved GCN pooling method and constructed an unsupervised pooling method inspired by the modularity metric of clustering quality. After multiple sets of controlled trials, the suggested model's overall performance was shown to be superior to the mainstream state-of-the-art at the time [12]. As a result, the proposed model is considered state-of-the-art. Zhu et al. found that GCN only focuses on the homogeneity of image nodes and ignores the heterogeneity among different image nodes in practical applications. In order to solve this problem, the researchers proposed a new graph convolution framework that contains an interpretable compatibility matrix for modeling the level of anisotropy or homotropy in a graph. Experimentally, it was concluded that the new framework has a significant reduction in the dependence on the training samples, while the accuracy of the image being an Oba was improved [13]. Kiningham et al. proposed a GNN gas pedal architecture for low-latency inference design, aiming to address the shortcomings of GCN's low efficiency for image processing. The architecture combined arithmetic-intensive vertex-centered operations with memory-intensive edge-centered operations and introduced a high-performance matrix multiplication engine. Experiments concluded that the proposed framework effectively reduces the sample latency and ensemble average [14].

In summary, many researchers have explored the application of GCN algorithm in various fields, but there are still more obvious shortcomings of this method for AR tasks. Therefore, the study selects the ASGCN algorithm as the core algorithm on the basis of GCN and introduces other advanced technologies to improve it, so as to construct a more perfect AR model to realize intelligent HCI.

3 Intelligent Human-Computer interaction model based on optimized ASGCN algorithm

The construction of human AR model based on LSTM's encoder and ASGCN algorithm is firstly discussed in depth, aiming to further improve the effect of HCI in daily life. To realize intelligent HCI, the study fuses LSTM with ED and adds it to the feature fusion module (FFM) of the HCI model in order to achieve feature improvement and accurate fusion.

3.1 Construction of human motion capture and recognition model based on ASGCN

The study uses the GCN algorithm to create the human motion capture model because of the GCN model's impressive performance in a variety of domains and the quick growth of artificial intelligence technologies [15]. Given the specific needs of this research, the study chooses ASGCN algorithm as the core algorithm of human motion capture model. ASGCN is an improved algorithm of spatio-temporal graph convolutional networks (STGCN), and ASGCN adds the extraction of human body joint features (JFs) in the spatial domain on the basis of STGCN to improve the accuracy and stability of AR [16-17]. The ASGCN algorithm stacks behavior-actions together to form a fused graph convolution module, thus learning spatial and feature sequences and performing AR, the emergence of this fused module when GCN overcomes the difficulty of poor dynamic processing. Furthermore, in terms of flexibility and scalability, the enhanced ASGCN algorithm outperforms the STGCN algorithm. Figure 1 depicts the ASGCN algorithm's recognition structure.

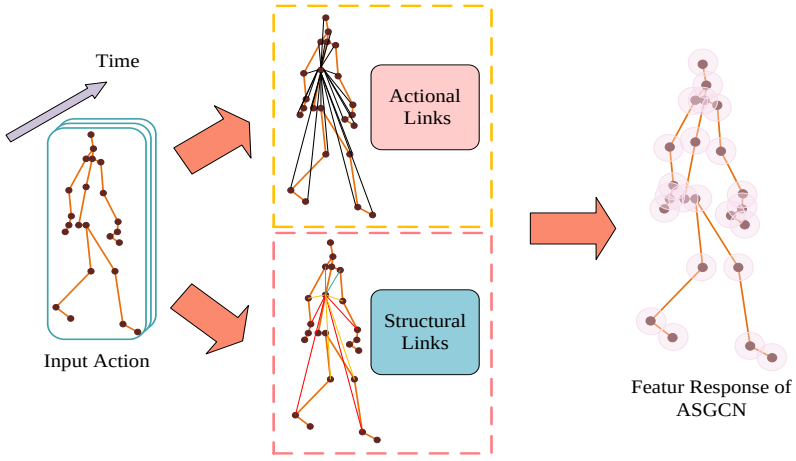


Figure 1: Schematic diagram of ASGCN algorithm recognition structure

Equation (1) displays the mathematical expression for the convolution computation used in ASGCN's JF extraction for the human body, which is based on a convolution kernel (CK).

$$f_{out}(x)=\sum_{h}^{K}\sum_{w}^{K} f_{in}\left(s(x,h,w)\cdot w(h,w)\right) \quad (1)$$

In Equation (1), K denotes the CK size and x denotes a point in the acquisition region. (h,w) denotes the height and width of the sampling region, and $W(h,w)$ denotes the weights of the sampling region. $s(x,h,w)$ is a sampling function whose computational expression is shown in Equation (2).

$$s(x,h,w)=x+\tilde{s}(h,w) \quad (2)$$

In Equation (2), $\tilde{s}(h,w)$ denotes the pixels in the neighborhood of point x. However, the skeletal model of the human body is an irregular image, so different weights need to be assigned for different skeletal joints in order to correctly analyze the behavior of the human body [18]. Therefore, after redefining the weights the convolution calculation expression of ASGCN is shown in Equation (3).

$$f_{out}(v_i)=\sum_{v_j\in N(v_i)}\frac{1}{Z_i(v_i)}f_{in}(v_j)\bullet w(l_i(v_i)) \quad (3)$$

In Equation (3), v_i denotes a point in the sampling area, v_j denotes a sampling point adjacent to v_i. $Z_i(v_i)=\left|\left\{v_k \mid l_i(v_k)=l_i(v_j)\right\}\right|$, the expression denotes normalization, which is used to balance the weights of different collection points. l_i denotes the set of mapping relationships from different collection points to neighboring collection points. The subset partition of the features to be extracted can address the issue that the model has to extract more JFs simultaneously because, as can be shown in the calculation above, a high number of samples supplied at once will likewise result in a huge amount of model computation. Figure 2 illustrates the commonly used subset division method in the ASGCN algorithm.

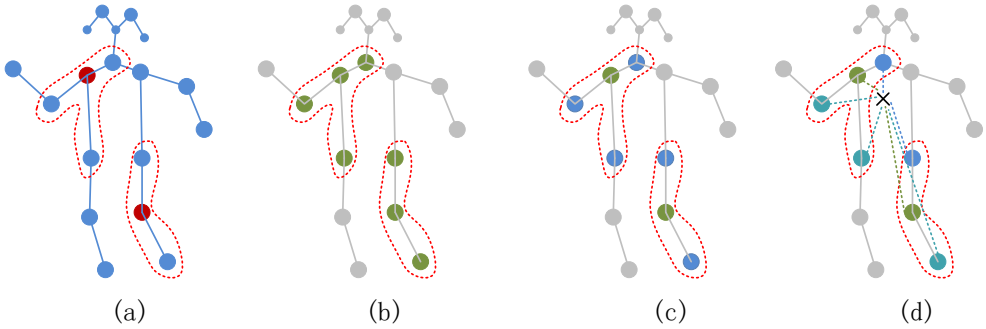


Figure 2: Schematic diagram of subset partitioning method

The subset division only solves the number of features extracted by the model at the same time, but it does not fundamentally solve the drawbacks of the ASGCN model, such as long time of accessing memory and more learning parameters. To solve the above drawbacks, the study adopts the displacement operation (DO) to simplify the learning parameters of GCN and ensure its computational efficiency and memory access efficiency. The computational expression of the DO is shown in Equation (4).

$$0_{a,b} = \sum_{i,j} K_{i,jm} I_{a+i,b+j,m} \quad (4)$$

In Equation (4), O denotes the output tensor and K denotes the size of the CK. m denotes the channel the output and input, and I denotes the size of the input tensor. j,i is the dimension index of the input tensor. In GCN, the CK is the core of extracting the human body used as a feature, which can aggregate the information in the image, the study uses the DO mainly to shrink the CK, so as to reduce the amount of computation and learning parameters. The DO method introduces unit scales at specific CK index positions, allowing the model to focus on local features rather than global information. This significantly reduces the size of the convolution kernel while maintaining the effectiveness of feature extraction. This method simplifies the model parameters, reduces the computational burden, and promotes memory access efficiency, thereby making DO an effective tool for reducing the CK size and simplifying the model structure. The size of the CK after reduction is shown in Equation (5).

$$K_{i,j,m} = \begin{cases} 1, i = i_m and j = j_m \\ 0, other \end{cases} \quad (5)$$

In Equation (5), i_m, j_m is the index, indicating that the size of the CK at (i_m, j_m) is 1 and the size of the CK at the rest of the locations is 0. The flow of the GCN convolution after the introduction of the DO is shown in Figure 3.

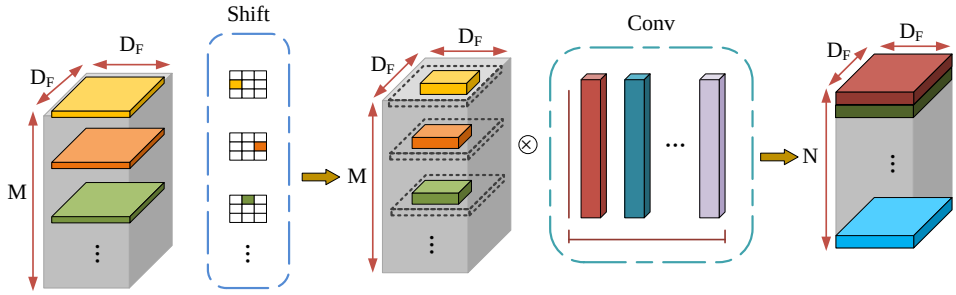


Figure 3: GCN convolution process after introducing displacement operation

In Figure 3, the left side of symbol ⊗ represents the displacement convolution module, the right side of convolution symbol represents FFM, and ⊗ is used to connect DO and convolution channels. In the displacement convolution module, the DO operation represents the primary step of the overall process. Its function is to enrich the representation of features and to capture broader neighborhood information, thereby enhancing the model's perception of multi-scale connectivity between vertices. The input of the module comprises multiple layers of features, each representing a distinct subset of features. Through the process of DO, these feature subsets are able to refine their positions in order to enhance the model's adaptability to local structures. Subsequently, FFM employs the inverse operation of subset partitioning rules to achieve the recombination of these feature subsets. This process is not merely a restoration of existing features. Rather, it

entails the acquisition of more nuanced and varied feature representations through the precise regulation of feature recombination. This approach enables the model to enhance its representation capabilities while simultaneously reducing its computational complexity [19]. To reflect the process of feature fusion, a mathematical expression for human action features is introduced in the study, as shown in Equation (6).

$$\tilde{f}_v = f_{(v,c)} \| f_{(E_v^1,c:2c)} \| f_{(E_v^2,2c:3c)} \| \cdots \| f_{(E_v^n,nc:)} \quad (6)$$

In Equation (6), c denotes the channel serial number and E denotes the set of neighboring acquisition points. n denotes the nodes of the neighboring collection points, v denotes the currently calculated collection points, and ω denotes the weight of each neighboring collection point. The weighted sum operation performed on each collection point and its neighborhood endows the model with the flexibility to identify heterogeneous connections between vertices, thereby improving its accuracy in identifying diverse human motion features and enhancing its ability to represent complex human motion patterns during the recognition process. In the context of graph convolution, the weight allocation of each adjacent collection point serves to quantify the importance of adjacent points, thereby ensuring that the heterogeneity of the graph structure is taken into account during feature aggregation. Therefore, the computational expression of ω is shown in Equation (7).

$$\omega = H\left(\eta_c D_k\left(x\right)\right) \quad (7)$$

In Equation (7), $L(.)$ denotes the activation function (AF) of the displacement convolution module, and the AF selected for the study is the ReLU function. η_C denotes the proportion of the c th channel to the total data, and $D_k\left(x\right)$ denotes the parameter mapping relationship of the input sample x. x denotes the input sample and k denotes the number of parameters. The model's speed of feature extraction and recognition can be somewhat increased by the DO and subset division, but it still lacks feature fusion and AR accuracy, therefore other sophisticated techniques must be included by the study in order to fully optimize the model.

3.2 Construction of action recognition model based on Ed improvement

The ASGCN model needs to subset the collected features during the construction process, although the model exists a FFM to splice the extracted features, it lacks a correction module. The correction module can check the features spliced by the ASGCN model degree and return the wrong features to the fusion module to be spliced again when wrong splicing is found [20]. The accuracy of the AR model can be increased by adding the correction module, which can also substantially increase the efficiency of feature fusion. In this study, the subset division rule is used as the input to the encoder, which in turn yields the corresponding coding sequence for the entire subset. The study's encoder is based on the LSTM method, which computes the positional characteristics of various subsets according to their weights at each encoding step while accounting for the subset's global location. LSTM has hidden state (HS) and memory state (MS) inside to record the historical data, so this algorithm can record the connection between different subsets better. And the HS and MS of LSTM is realized by three important structures of input gate (IG), forgetting gate (FG) and output gate (OG), which are defined as shown in Equation (8).

$$\begin{cases} f_t = \sigma(W_f[h_{t-1}; x_t]) \\ i_t = \sigma(W_i[h_{t-1}; x_t]) \\ o_t = \sigma(W_o[h_{t-1}; x_t]) \end{cases} \quad (8)$$

In Equation (8), σ denotes the standard deviation, and f_t, i_t, o_t denotes the FG, IG and OG, respectively. t denotes the moment, and W_f, W_i, W_o the three parameters denote the overall weight matrix of the FG, IG and OG, respectively. x_t denotes the input data of the t moment. In the calculation of MS and HS also need to carry out the calculation of candidate MS, the calculation expression is shown in Equation (9).

$$\tilde{C}_t = \sigma(W_c[h_{t-1}; x_t]) \quad (9)$$

In Equation (9), $\tilde{C}_t$ is the candidate MS and h_t is the HS. W_c denotes the weight matrix of the candidate MS. The candidate MS is the current moment MS, including the new candidate information and parameters added in the LSTM at this time [21]. After obtaining the candidate MS, it is also necessary to calculate the MS and HS of the LSTM, and its calculation expression is shown in Equation (10).

$$\begin{cases} h_t = o_t \otimes \tanh(C_t) \\ C_t = i_t \otimes \tilde{C}_t + f_t \otimes C_{t-1} \end{cases} \quad (10)$$

In Equation (10), $\otimes$ denotes a multiplication operation between each dimension of the same feature. C_t is the memorized state and h_t is the HS. In LSTM state computation, the AFs used are all Sigmoid (.) function. LSTM is based on FGs, IGs and OGs to form a hidden unit which is the core of the LSTM algorithm [22]. The structure of the hidden unit is shown in Figure 4.

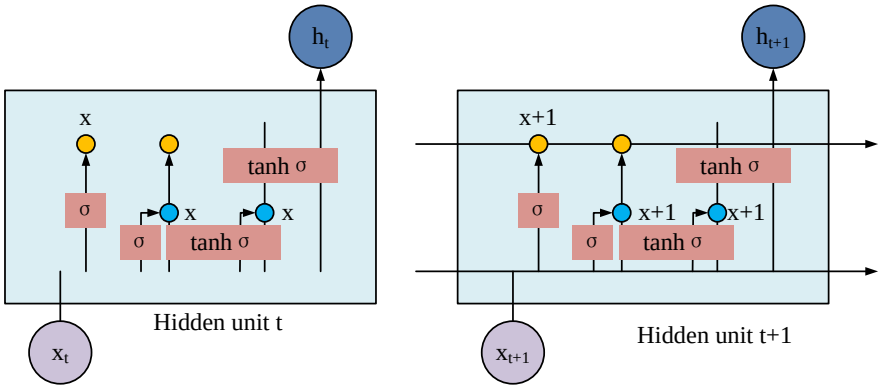


Figure 4: LSTM hidden unit structure diagram

The ED architecture with the introduction of LSTM algorithm can record the position information of the whole subset by memorizing the state. In addition, in the encoder, the expression for positional feature calculation is shown in Equation (11).

$$S_t = \sum_s \omega_{ts} h_s \tag{11}$$

In Equation (11), ω_{ts} denotes the weight of the subset at s when the time step of the encoder is t. l_s denotes the features of this subset. Where ω_{ts} is defined as shown in Equation (12).

$$\omega_{ts} = \frac{\exp(F(q_t, l_s))}{\sum_{s'} \exp(F(q_t, l_{s'}))} \tag{12}$$

In Equation (12), q_t denotes the state of the encoder in the hidden layer of the LSTM algorithm at time step t. $F(q_t, l_s)$ denotes the correlation between q_t and l_s. The calculation of correlation mainly consists of two forms: multiplication and addition, in order to avoid excessive model computation, the study adopts the method of addition for the correlation calculation of q_t and l_s, whose computational expression is shown in Equation (13).

$$F(q_t, l_s) = v_a^t \tanh(W_{a1} q_t + W_{a2} l_s) \tag{13}$$

In Equation (13), $\tanh(.)$ denotes the hyperbolic tangent function. v_a^t denotes the output sequence of the OG of the LSTM algorithm at the moment t. W denotes the weight matrix of q_t and l_s. The structure of ED is shown in Figure 5.

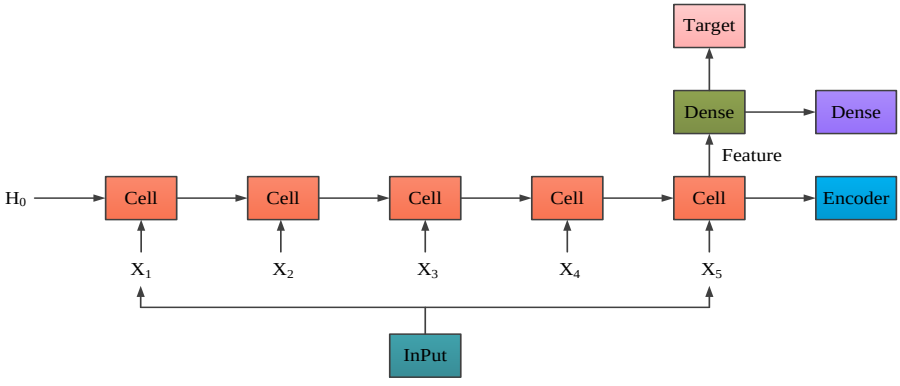


Figure 5: Schematic diagram of encoder-decoder architecture

Once the computation of the encoder is completed, the decoder can then be checked against the output of the encoder after going through a subset of the decoder feature fusion method. The study incorporates the decoder into the FFM so that the decoder becomes a submodule of the FFM, and after this operation, the FFM has the ability to check. During the encoder and decoder training process, some parameters of the hidden layer and *Soft* max classifier of the LSTM algorithm can be migrated to the decoder for training, and the encoder is obtaining the trained parameters by inverse operation. The ED model's capacity to generalize its parameters and its computing efficiency can both be enhanced via parameter migration training. The structure of the AR model fusing the encoder and decoder is shown in Figure 6.

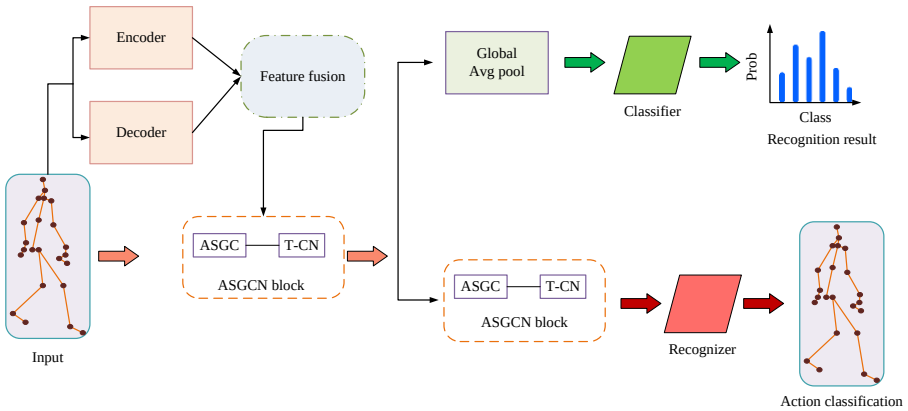
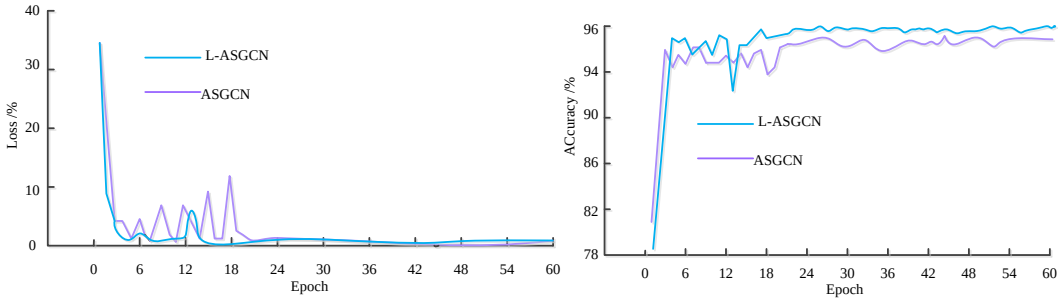


Figure 6: Schematic diagram of the action recognition model structure integrating encoder and decoder

The study's construction of an intelligent HCI model is almost complete, and for the convenience of subsequent experiments, the study replaces the proposed model with the acronym L-ASGCN model.

4 L-ASGCN model performance testing and analysis

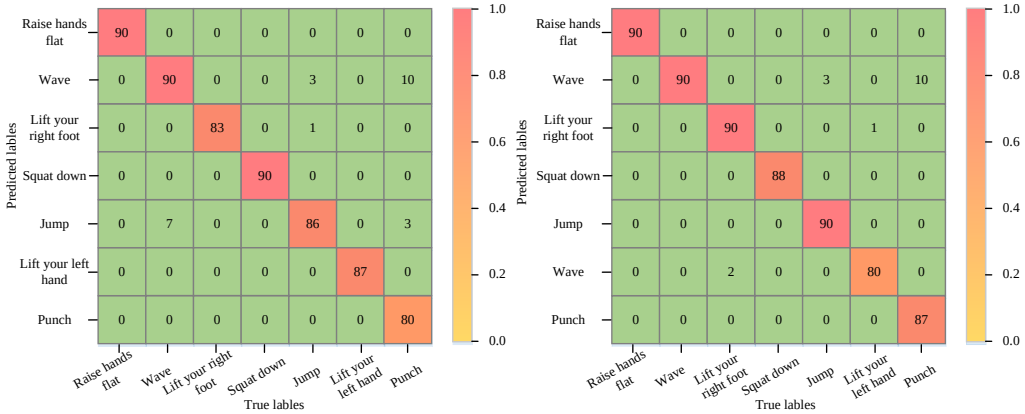
ASGCN, STGCN, and GCN are used as control models for controlled experiments in the study in order to verify the complexity of the intelligent HCI model that is suggested. The equipment required for the experiment is a computer with Intel Xeon w9-3495X CPU, 16GB of running memory, and RTX 2080 Ti graphics card. The experiments mainly used NTU RGB-D dataset, Interaction Action RGB-D dataset and Kinetics dataset. The Kinetics dataset comprises a diverse range of Internet videos, encompassing a multitude of daily activity scenarios. Its diversity and scale render it the benchmark dataset for behavior recognition algorithms. The Interaction Action RGB-D dataset is designed to record the interaction behavior between two individuals, providing detailed multimodal data on human actions and interaction scenarios. The NTU RGB-D dataset represents a comprehensive behavior recognition dataset that encompasses a diverse array of human activities, thereby providing a wealth of human behavior recognition scenarios for deep learning models. The selected dataset facilitates the deep feature learning of the model, particularly in the context of bone and joint data, thereby enhancing the accuracy of human pose estimation. In terms of evaluation indicators, accuracy, loss rate, recall rate, and F1 score are key measures of model performance. The accuracy of a model reflects its ability to predict correctly. The loss represents the accumulation of prediction errors during the training process. The recall measures the model's ability to recognize positive instances. The F1 score is the harmonic average of accuracy and recall, which can be used to evaluate the overall performance of the model.



(a) Comparison chart of loss rates for different models (b) Comparison chart of accuracy of different models

Figure 7: Comparison of loss and accuracy of different models

Accuracy and loss rate are important indicators of model performance, the study used NTU RGB-D dataset to train the STGCN model and L-ASGCN model for 30 min respectively, and then used the Interaction Action RGB-D dataset as the input to conduct the comparison experiment of accuracy and loss rate. Figure 7(a) shows the evolution of the loss rates for the STGCN and L-ASGCN models. Based on the figure's trend of curves, it is evident that both models' loss rates drop as the iterations increases. However, the loss rate curve of the STGCN model shows obvious oscillations before convergence, while the L-ASGCN model has no obvious oscillations before convergence. In addition, the loss rate of the STGCN model after convergence is about 0.91%, while the loss of the L-ASGCN model after convergence is only 0.56%, so the proposed model has some advantages in loss rate. Figure 7(b) represents the comparison of the accuracy rates of the STGCN model and the L-ASGCN model. The results presented in Figure 7(b) suggest that, initially, the L-ASGCN model's accuracy is not as high as the STGCN models. However, after a few iterations of the model, the accuracy of the suggested model grows significantly. The AverA of the proposed model, when the two models converge, is 95.39%, which is 1.41% greater than the AverA of the STGCN model. Additionally, the suggested model's accuracy smoothness is superior to that of the control model.



(a) STGCN model confusion matrix (b) L-ASGCN model confusion matrix

Figure 8 Comparison diagram of confusion matrices for different models

The confusion matrix (ConM) of the STGCN and L-ASGCN models are compared in Figure 8, with Figure 8(a) showing the ConM the STGCN model produced using the Interaction Action RGB-D dataset. The ConM produced by the suggested model using this dataset is shown in Figure 8(b). The average score obtained by the proposed model is about 87.86, and the average score of the STGCN model is 86.57. By contrasting the aforementioned findings, it is evident that the proposed model of the study has some development because its AR effect on the same dataset is superior to that of the control model.

Table 1: Comparison of output times of different models on the Kinetics dataset for each module

Model name	Feature extraction time (s)	Characterized recombination time (s)	Action recognition time (s)	Total Time (s)
L-ASGCN	3.1	0.7	0.3	4.1
ASGCN	3.5	1.1	0.4	5.0
STGCN	3.7	1.1	0.6	5.4
GCN	4.2	1.7	0.8	6.7

Table 1 presents a comparison of the output time and total elapsed time for each module on the Kinetics dataset. It can be observed that the total elapsed time is lowest for the L-ASGCN model, followed by the ASGCN model, the STGCN model, and the GCN model. With regard to the time required for the output of each module, the feature recombination time consumption of the ASGCN model and the STGCN model is identical. This is due to the fact that the ASGCN model is not optimized for feature recombination during the process of improvement. Consequently, the feature recombination module of the ASGCN model and the feature recombination module of the STGCN model are both subject to the same time constraints. The research-proposed model performs best across all modules and in terms of the overall output elapsed time, according to the experimental data, demonstrating its superior computational and feature-processing capabilities.

Table 2: Comparison of actual scene recognition performance between L-ASGCN model and ASGCN model

Behavior	Accuracy (%)		Recognition time (s)	
	L-ASGCN	ASGCN	L-ASGCN	ASGCN
Raise hands flat	94.5	90.2	1.13	1.51
Lift left hand	95.9	87.3	1.21	1.39
Lift right hand	94.7	86.9	1.26.	1.40
Cross hands	91.2	91.1	0.91	1.11
Lift left foot	92.3	88.6	1.22	1.29
Lift right foot	92.1	88.4	1.22	1.31
Squat down	90.1	90.8	1.08	1.33
Punching	89.6	88.1	1.39	1.45
Jump	94.9	90.2	1.21	1.30
Wave	87.2	89.9	1.29	1.26

To test the effectiveness of the model's application in real life, the study randomly selects 10 volunteers for testing, each action is done 30 times during the test, and the behavioral actions of the volunteers are inputted into the experimental model in video form. The L-ASGCN model has the highest recognition rate for the hand-raising action, and a lower AR rate for the hand-waving, but the overall recognition rates of the model are all around 90%. Additionally, a comparison of the L-ASGCN and ASGCN models' findings reveals that the former has better accuracy and requires less time to recognize an action, which supports the suggested model's AR efficiency.

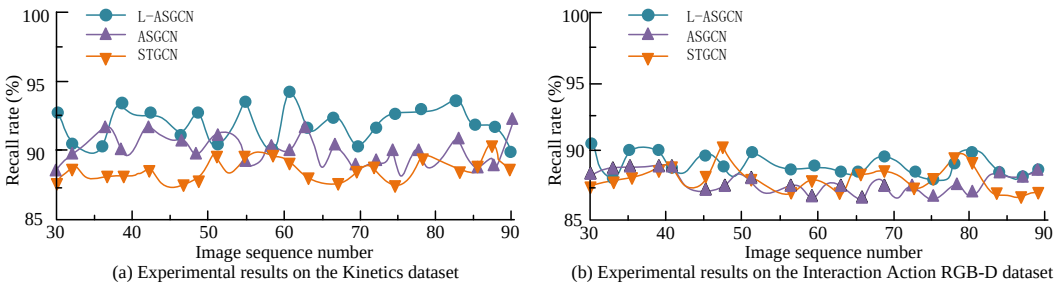


Figure 9: Comparison of the change in recall of each model on different datasets

Figure 9 represents the recall comparison of L-ASGCN model, ASGCN model, and STGCN model on Kinetics dataset and Interaction Action RGB-D dataset. The experimental outcomes of the three models on the Kinetics dataset are shown in Figure 9(a). The average recall of L-ASGCN model, ASGCN model and STGCN model are 93.06%, 91.71% and 87.86% respectively. Figure 9(b) represents the trend of the recall of each model on the Interaction Action RGB-D dataset. Based on the results in Figure 9(b), the average recalls of L-ASGCN model, ASGCN model and STGCN model are calculated as 89.46%, 87.92% and 86.11%, respectively.

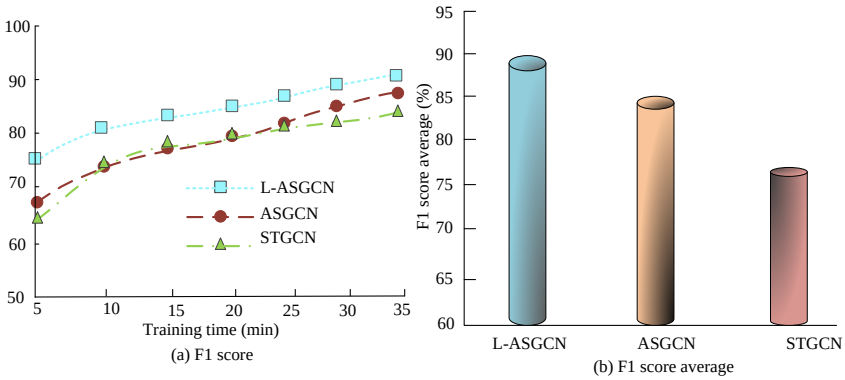


Figure 10: Schematic of F1 scores vs. average F1 scores for each model

Figure 10 shows a comparison of the F1 scores and average F1 scores of three models, L-ASGCN, ASGCN, and STGCN, on the Interaction Action RGB-D dataset. Figure 10(a) illustrates the relationship between the F1 scores of the three models and training time. The F1 scores of each model increase as the training time increases. Among the experimental models, the F1 score of the L-ASGCN model is the highest. Figure 10(b) shows the average F1 scores of each model after repeating the experiment three times. The L-ASGCN model achieved an average F1 score of 89.79%, followed by the ASGCN model with 85.97%, and the STGCN model with 78.66%.

5 Discussion

In the field of human behavior recognition, GCN has emerged as an effective data analysis tool. In order to achieve more precise and efficient action recognition capabilities and to promote the intelligent development of HCI technology, an L-ASGCN model was studied and constructed. The loss rate of the L-ASGCN model after convergence was 0.56%, with an overall recognition rate of approximately 90%, an average recall rate of 93.06%, and an average F1 score of 89.79%. The model demonstrated significant advantages over STGCN and ASGCN in multiple performance indicators. The primary rationale for this outcome was that L-ASGCN streamlines the convolutional kernel of the model through the utilization of displacement operations, thereby reducing the overall computational complexity and enhancing operational efficiency. In comparison to the studies conducted by Ahmad et al. [21] and Tong et al. [22], the enhanced performance of the L-ASGCN algorithm in processing human motion data represented a significant advancement in the application of GCN in the field of motion recognition. In particular, with regard to recall and model efficiency, L-ASGCN offered a more refined feature extraction and recombination mechanism than the dynamic virtual network embedding algorithms explored by Zhang et al. [23]. This was because it not only processes features but also suppresses performance loss caused by excessive computation, thereby optimizing model performance. Nevertheless, the L-ASGCN algorithm employed in this study is not without its limitations. Chief among these was the fact that the model is unable to capture global physical dependencies between joints, and that the motion capture is based on fixed skeletons. Future work should aim to enhance the generalization ability of models for different types of actions and to optimize real-time action recognition technology. In conclusion, the L-ASGCN model has considerable potential for application in human behavior recognition tasks, and can be applied in the fields of medical rehabilitation, intelligent monitoring, and interactive media. Furthermore, L-ASGCN has established a foundation for more efficient real-time action recognition in various dynamic environments in the future.

6 Conclusion

In the contemporary era where intelligent development has become mainstream, HCI is an important part of the development of intelligent terminal devices. An excellent HCI model can facilitate the intelligent terminal's understanding of human commands so as to better serve humans. In view of this, the study adopts the ASGCN algorithm and the LSTM-based ED module for fusion, so as to construct an intelligent AR model. The outcomes indicated that the L-ASGCN model achieved a loss of only 0.56% after convergence on the test dataset, with an

AverA of 95.39%. Additionally, the study tested the recall of the proposed model, and the results showed that the average recall of the L-ASGCN model was 93.06%, which is 1.35% higher than the ASGCN model and 5.20% higher than the STGCN model. Regarding the test experiments on F1 scores, the L-ASGCN model achieved an average F1 score of 89.79%, while the ASGCN model and the STGCN model achieved average F1 scores of 85.97% and 78.66%, respectively. These results indicated that the model proposed in the study had a higher overall performance. Meanwhile, the study also tested the proposed model for AR in real-life scenarios. The outcomes revealed that the proposed model has an overall recognition rate of around 90% for routine actions, indicating its feasibility and advancement. At the same time, there are some shortcomings in this study, such as the proposed model only captures the local physical dependence between joints and motion capture based on a fixed skeleton, so the model needs to be further optimized to address the shortcomings.

Reference

[1] Stephan Diederich, Alfred Benedikt Brendel, Stefan Morana, and Lutz Kolbe. On the design of and interaction with conversational agents: An organizing and assessing review of human-computer interaction research. Journal of the Association for Information Systems, 23(1): 96-138, 2022. https://doi.org/10.17705/1jais.00724

[2] Barbara Rita Barricelli， Daniela Fogli. Digital twins in human-computer interaction: A systematic review. International Journal of Human–Computer Interaction, 40(2): 79-97, 2024. https://doi.org/10.1080/10447318.2022.2118189

[3] Li Xiaofei, Jiang Miao, Du Yiming, Ding Xin, Xiao Chao, Wang Yanyan, Yang Yanyu, Zhuo Yizhi, Zheng Kang, Liu Xianglan, Chen Lin, Gong Yi, Tian Xingyou, Zhang Xian. Self-healing liquid metal hydrogel for human–computer interaction and infrared camouflage. Materials Horizons, 10(8): 2945-2957, 2023. https://doi.org/10.1039/d3mh00341h

[4] Rajdeep Ghosh, Souvik Phadikar, Nabamita Deb, Nidul Sinha, Pranesh Das, Ebrahim Ghaderpour. Automatic eyeblink and muscular artifact detection and removal from EEG signals using k-nearest neighbor classifier and long short-term memory networks. IEEE Sensors Journal, 23(5): 5422-5436, 2023. https://doi.org/10.1109/JSEN.2023.3237383

[5] Anitha Rani Inturi, V. M. Manikandan, Vignesh Garrapally. A novel vision-based fall detection scheme using keypoints of human skeleton with long short-term memory network. Arabian Journal for Science and Engineering, 48(2): 1143-1155, 2023. https://doi.org/10.1007/s13369-022-06684-x

[6] M. Kalpana Chowdary, Tu N. Nguyen & D. Jude Hemanth. Deep learning-based facial emotion recognition for human–computer interaction applications. Neural Computing and Applications, 35(32): 23311-23328, 2023. https://doi.org/10.1007/s00521-021-06012-8

[7] Liu, Hai, Liu, Tingting, Zhang, Zhaoli, Sangaiah, Arun Kumar, Yang, Bing, Li, Youfu. Arhpe: Asymmetric relation-aware representation learning for head pose estimation in industrial human–computer interaction. IEEE Transactions on Industrial Informatics, 18(10): 7107-7117, 2022. https://doi.org/10.1109/TII.2022.3143605

[8] Zhang, Hao, Zhang, Dongzhi, Wang, Zihu, Xi, Guangshuai, Mao, Ruiyuan, Ma, Yanhua, Wang, Dongyue, Tang, Mingcong, Xu, Zhenyuan, Luan, Huixin. Ultrastretchable, self-healing conductive hydrogel-based triboelectric nanogenerators for human–computer interaction. ACS Applied Materials and Interfaces, 15(4): 5128-5138, 2023. https://doi.org/10.1021/acsami.2c17904

[9] Zhang, Ronghui, Jiang, Chunxiao, Wu, Sheng, Zhou, Quan, Jing, Xiaojun, Mu, Junsheng. Wi-Fi sensing for joint gesture recognition and human identification from few samples in human-computer interaction. IEEE Journal on Selected Areas in Communications, 40(7): 2193-2205, 2022. https://doi.org/10.1109/JSAC.2022.3155526

[10] Alaa Bessadok, Mohamed Ali Mahjoub, Islem Rekik. Graph neural networks in network neuroscience. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(5): 5833-5848, 2022. https://doi.org/10.48550/arXiv.2106.03535

[11] Lingfei Wu, Yu Chen, Kai Shen, Xiaojie Guo, Hanning Gao, Shucheng Li, Jian Pei, Bo Long. Graph neural networks for natural language processing: A survey. Foundations and Trends® in Machine Learning, 2023, 16(2): 119-328. https://doi.org/10.48550/arXiv.2106.06090

[12] Qian Wang, Youfa Liu. Energy Levels Based Graph Neural Networks for Heterophily. Journal of Physics Conference Series. 1948(1): 012042. https://doi.org/10.1088/1742-6596/1948/1/012042

[13] Qian Wang, Youfa Liu. Energy Levels Based Graph Neural Networks for Heterophily. Journal of Physics Conference Series. 1948(1): 012042. https://doi.org/10.1088/1742-6596/1948/1/012042

[14] Kevin Kiningham, Philip Levis, Christopher Ré. GRIP: A graph neural network accelerator architecture. IEEE Transactions on Computers, 72(4): 914-925, 2022. https://doi.org/10.1109/TC.2022.3197083

[15] Adem Aylin, akt Erman, Dadeviren Metin. Selection of suitable distance education platforms based on human–computer interaction criteria under fuzzy environment. Neural Computing and Applications, 34(10): 7919-7931, 2022. https://doi.org/10.1007/s00521-022-06935-w

[16] Milani Alireza Sadeghi, Cecil-Xavier Aaron, Gupta

Avinash, Cecil J, Kennison Shelia. A systematic review of human–computer interaction (HCI) research in medical and other engineering fields. International Journal of Human–Computer Interaction, 40(3): 515-536, 2024. https://doi.org/10.1080/10447318.2022.2116530

[17] Xie Yaochen, Xu Zhao, Zhang Jingtun, Wang Zhengyang, Ji Shuiwang. Self-supervised learning of graph neural networks: A unified review. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(2): 2412-2429, 2022. https://doi.org/10.48550/arXiv.2102.10757

[18] He Jinbao, Yang Jie. Network security situational level prediction based on a double-feedback Elman model. Informatica, 46(1): 87-93, 2022. https://doi.org/10.31449/inf.v46i1.3775

[19] Utkin Lev V, Zhuk Kirill D. Improvement of the deep forest classifier by a set of neural networks. Informatica, 44(1):1-13, 2020. https://doi.org/10.31449/inf.v44i1.2740

[20] Yuan Hao, Yu Haiyang, Gui Shurui, Ji Shuiwang. Explainability in graph neural networks: A taxonomic survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(5): 5782-5799, 2022. https://doi.org/10.1109/TPAMI.2022.3204236

[21] Ahmad Tasweer, Jin Lianwen, Zhang Xin, Lai Songxuan, Tang Guozhi, Lin Luojun. Graph convolutional neural network for human action recognition: A comprehensive survey. IEEE Transactions on Artificial Intelligence, (2): 128-145, 2021. https://doi.org/10.1109/TAI.2021.3076974

[22] Tong Houjie, Qiu Robert C, Zhang Dongxia, Yang Haosen, Ding Qi, Shi Xin. Detection and classification of transmission line transient faults based on graph convolutional neural network. CSEE Journal of Power and Energy Systems, 7(3): 456-471, 2021. https://doi.org/10.17775/CSEEJPES.2020.04970

[23] Zhang Peiying, Wang Chao, Kumar Neeraj, Zhang Weishan, Liu Lei. Dynamic virtual network embedding algorithm based on graph convolution neural network and reinforcement learning. IEEE Internet of Things Journal, 9(12): 9389-9398, 2021. https://doi.org/10.48550/arXiv.2202.02140

Appendix

Table A summarizes the content of the above research.

Table A: Summary of related work

Research contents	Researchers	Key findings	Potential Shortcomings
Human-computer interaction	Chowdary et al. [6]	Using deep learning techniques to recognize emotions and promote intelligent human-computer interaction	Not taking into account the differences in emotional expression across different cultural backgrounds
	Liu et al. [7]	Solving the problem of adjacent pose information processing and mislabeling gap in head pose estimation	Robustness in ever-changing environments may need to be improved
	Zhang et al. [8]	Constructing a glove based human-machine interaction system using a frictional electric nanogenerator. Extracting and analyzing multidimensional signal features to achieve gesture visualization and robotic arm control	More data support may be required for the recognition of complex gestures
	Zhang et al. [9]	Dynamic mode of gesture recognition using WiFi and Radar sensing technology	System complexity and power consumption may be obstacles in practical applications
GCN research contents	Bessadok et al. [10]	Medical image recognition based on learning deep graph neural networks; Combining DNN and GNN for identifying human brain neuronal activity	Unknown generalization ability for large-scale image datasets
	Wu et al. [11]	A natural language processing model based on GCN; Introduced ED technology to achieve global encoding of input data	Adjustments may be needed to adapt to different natural language processing tasks

	Zhu et al. [12]	Unsupervised image analysis based on GCN and DNN; Improving GCN pooling method through DNN to achieve cluster structure recovery	Further validation is needed to evaluate the effectiveness of unsupervised methods on diverse datasets
	Zhu et al. [13]	Develop a new graph convolutional framework to address the homogeneity assumption problem of GCN; Introducing interpretable compatibility matrices to model heterogeneity in graphs	Insufficient scalability of the framework
	Kiningham et al. [14]	GNN gas pedal architecture with low latency inference design; Combining vertex and edge operations, introducing a high-performance matrix multiplication engine	Further consideration is needed for real-time performance and computational resource consumption