Design of Personalized VR Short Video Content Distribution Algorithm based on Artificial Neural Network

Han Zhong[1], Donghyuk Choi[2], Wonho Choi[3], Yu Zheng[4]*
School of Culture and Media, Anyang Institute of Technology, Anyang, China[1]
Dept of Animation & Visual Effects, Dongseo University, Busan, South Korea[2]
Dept of Visual Contents, Graduate School, Dongseo University, Busan, South Korea[3, 4]
School of Media, Anyang Normal University, Anyang, China[4]

***Abstract*—In order to improve the video quality, reduce the number of video jams, and improve the video transcoding rate, a new personalized distribution algorithm of VR short video content based on artificial neural network is proposed. BP neural network is used to compress the original video and determine the execution mode and cache location of VR short video cache file; Transcoding cached high bit rate VR short video stream to generate low bit rate video stream to meet the needs of different network bandwidth; Build a video distribution model, and build a multicast distribution tree based on this model, that is, add some relay servers to minimize the video distribution cost; Finally, through the algorithm of minimizing the distribution cost of VR short video, the bandwidth loss and response delay are effectively reduced to achieve the goal of minimizing the distribution cost. The experimental results show that the video blocking times of this method are always less than six times, which effectively reduces the video blocking times. The PSNR value is high, the increase is large, up to 0.5, and the video transcoding rate is improved, up to 92%.**

Keywords—Artificial neural network; VR short video; BP neural network; video cache; cost optimization

I. INTRODUCTION

With the rapid development of network information technology, people rely more and more on the Internet and smart phones for entertainment. The emergence of various short video applications has met the needs of users for fragmented reading, social interaction and personalized expression, but problems have also arisen [1, 2]. At present, the short video industry is still in the rapid development stage. In view of the current situation of the industry environment, it is necessary to analyze the environment of the short video distribution platform to provide a reference solution to the problem. Traditional wireless network access and transmission modes are single, and usually rely on network infrastructure such as base stations and core networks to provide services. Once network congestion or force majeure occurs, users will not be able to obtain corresponding services [3, 4]. In this context, the growing user demand for content services and the limited network carrying capacity have become the main contradiction restricting the development of video. In order to meet the requirements of high quality and low latency for the development of short video, and reduce the network load at the same time, it is important to study the personalized distribution method of short video content.

II. LITERATURE REVIEW

Literature [5] proposed an online short video content distribution method based on federated learning. First, based on the federated learning method, the user group interest vector prediction algorithm is proposed by training the interest prediction model using the user group's local album data, and the user group interest vector representation is obtained; Then, with the user group's interest vector as the input, the corresponding short video content distribution strategy is designed in real time based on the combined confidence upper bound algorithm. The experimental results show that the short video content distribution strategy can effectively improve the accuracy of short video distribution, and help to improve the profits of video content providers. Literature [6] proposed a video distribution method based on QoE awareness. According to the characteristics of optical wireless fusion network structure, considering link state information and scalable video coding structure, a QoE evaluation model was established. Particle swarm optimization algorithm is used to select the optimal video transmission rate, and then analyze the node transmission capacity and node matching degree, so as to select the transmission path for the service and ensure the reliable transmission of the service. The experimental results show that this method can effectively improve the utilization of network resources and reduce network congestion while enhancing the quality of user experience. Literature [7] proposed a video distribution method based on random linear network coding. The source node uses RLNC to segment the video files to be distributed, and the encoded data pieces are distributed to mobile devices on the road. After receiving the data slice, each mobile device will re encode and distribute it to other devices. When the device receives a certain amount of linearly independent data, it will decode the data slice to obtain the original data. Three strategies are designed: mode switching, wireless access point selection, and active re coding to reduce the delay of video distribution. Experimental results show that this method has higher distribution efficiency.

From the above analysis, we can see that the existing methods improve the video distribution effect to a certain extent. In order to further improve the video quality, reduce the

*Corresponding Author.

number of video blocks, and improve the video transcoding rate, this paper proposes a personalized VR short video content distribution algorithm based on artificial neural network. The main research contents of this method are as follows:

1) Solve the problem of VR short video cache reuse by reasonably deploying and placing VR short video files.

2) The drift error adaptive fast video transcoder transcodes the buffered high bit rate VR short video stream to generate a low bit rate video stream to meet the needs of different network bandwidth.

3) Improve the video distribution model, generate a multicast distribution tree, and conduct personalized distribution of VR short video content to effectively reduce bandwidth loss and response delay, so as to minimize the distribution cost.

III. VR Short Video Preprocessing

A. Video Compression

In recent years, with the development of artificial neural network theory and technology, its application in video compression has gradually attracted people's attention. Compared with some traditional compression methods, the artificial neural network technology [8, 9] has good fault tolerance, self-organization and adaptability. Therefore, it is not necessary to use some predetermined data coding algorithm in the video compression process, but can independently complete the video compression completely according to the information characteristics of the video itself. BP neural network [10] is a typical and commonly used network structure in artificial neural network. This paper mainly introduces a video compression method based on BP neural network.

BP neural network refers to a multilayer feedforward network using BP algorithm. Fig. 1 is a simple feedforward network, which consists of an input layer, an output layer and a hidden layer (middle layer). Where, w_{zh} and w_{vh} are the weighted values between the input layer and the intermediate layer (hidden layer), hidden layer and output layer respectively. The neurons in the hidden layer and the output layer first sum the input signals of the previous layer by weighting, and then perform the pre specified transformation to generate the output signals. The output of the hidden layer neurons is also connected to the output layer neurons by weighting.

Let the input vector of the network be $Z=\{z_1,z_2,...,z_n\}$, where n represents the number of neurons in the input layer. The transformation function adopted by neurons is:

$$D(n)=1+\frac{1}{\exp(w_{zh}-w_{vh})} \tag{1}$$

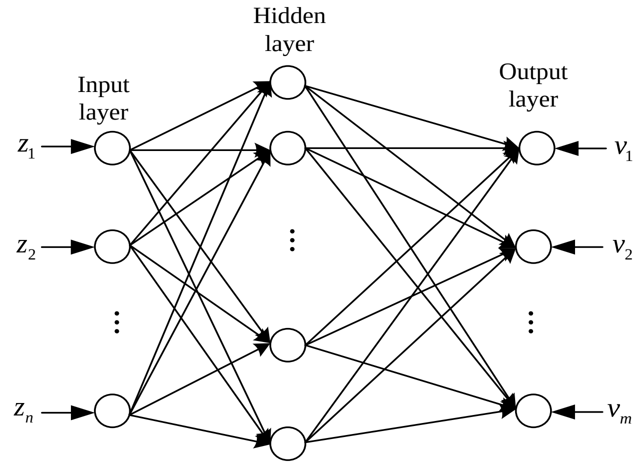

Fig. 1. Multilayer forward network.

The output vector corresponding to Z is $V=\{v_1,v_2,...,v_m\}$, where m represents the number of neurons in the output layer. If the expected output of the network is required to be $F=\{f_1,f_2,...,f_m\}$, the error function can be defined as:

$$E(n)=\sum_{i,j=1}^{m}\left(f_{ij}-v_{ij}\right)^2 \tag{2}$$

In the formula, f_{ij} represents the network training speed; v_{ij} represents the network convergence rate.

The BP algorithm uses the gradient descent method to adjust the network weight to reduce the above error function, namely:

$$\eta(n+1)=\eta(n)-\alpha\frac{E(n)}{\lambda(\eta)} \tag{3}$$

In the formula, $\eta(n)$ represents the approximation error of the neural network; $E(n)$ represents the difference function between the actual output and the expected output; $\lambda(\eta)$ represents generalization capability of neural network; α represents the weight adjustment rate, usually $0.01\le\alpha\le1$.

The general weight adjustment formula is:

$$\Delta\alpha=\sum_{i=1}^{m}E_{ij}\left|G_{ij}-H_v\right| \tag{4}$$

In the formula, G_{ij} represents the weight correction from the i-th node of a layer to the j-th node of the next layer; H_v represents the output of the node; E_{ij} represents the endpoint equivalent error of node i and node j, which is equivalent back transmitted from the error of the output layer:

$$\eta_v = \Delta\alpha\left(v_i - f_i\right) \tag{5}$$

In the formula, v_i represents that node i is located in the output layer; f_i represents that node i is in hidden layer.

For a VR short video with N frame, when BP neural network is used for compression, the number of neurons in the input and output layers of the network is required to be n^2, while the number of neurons in the hidden layer is much less than that in the input layer. When VR short video is transmitted in the network, the state of each neuron in the hidden layer, as some transformation result of the input information, must contain some information of the original VR short video. Therefore, when the output unit can reproduce the original VR short video, it can be considered that the output of hidden layer neurons is the compression result of the original input VR short video. Fig. 2 describes the VR short video compression process [11].

B. Video Cache

This section will focus on solving the problem of VR short video, and solve the problem of VR short video cache reuse by reasonably deploying and placing VR short video files. After the VR short video compression is completed in Section 2.1, determine the execution mode of the VR short video cache file and whether the cache location is appropriate [12].

Suppose user videos are stored in K clusters, $K = \left\{k_1^1, k_2^1, ..., k_L^1, ..., k_1^r, k_2^r, ..., k_L^r\right\}$, where k_l^i represents the VR short video file l stored in user cache k^i. Update VR short video files and store them in each blank cache. Among them, cluster member k_l^i only stores some files in the VR short video library. Due to storage restrictions, these files cannot contain all VR short video files. Therefore, it is expected to deploy more file contents in k_l^i and cache and reuse them.

For each cluster, selecting the video file to update is also an important issue. In each cluster, cluster file library L_p represents all video file records in the cluster, and $L_p = z_1 \cup z_2, ..., \cup z_l$ represents an independent cluster popularity distribution. On this basis, collect the VR short video file statistics cached by the whole cluster members in the cluster, and then update the missing videos. It is not difficult to find out which VR short video files are missing in the cluster according to the VR short video file statistics [13].

At the same time, considering the problem of video storage capacity, the storage formula of VR short video files is expressed as:

$$V_f = Z \subseteq C_m, m = 1, 2, ..., M \tag{6}$$

Among them, the maximum possibility of updating the number of VR short video files is as follows:

$$\mu_{\max} = M - C_m \tag{7}$$

Therefore, to update the VR short video file library L_p, assume that each file requested to be updated is equal to:

$$\phi_{L_p} = \left\{p \notin L_p, \mu_{\max} \le M\right\} \tag{8}$$

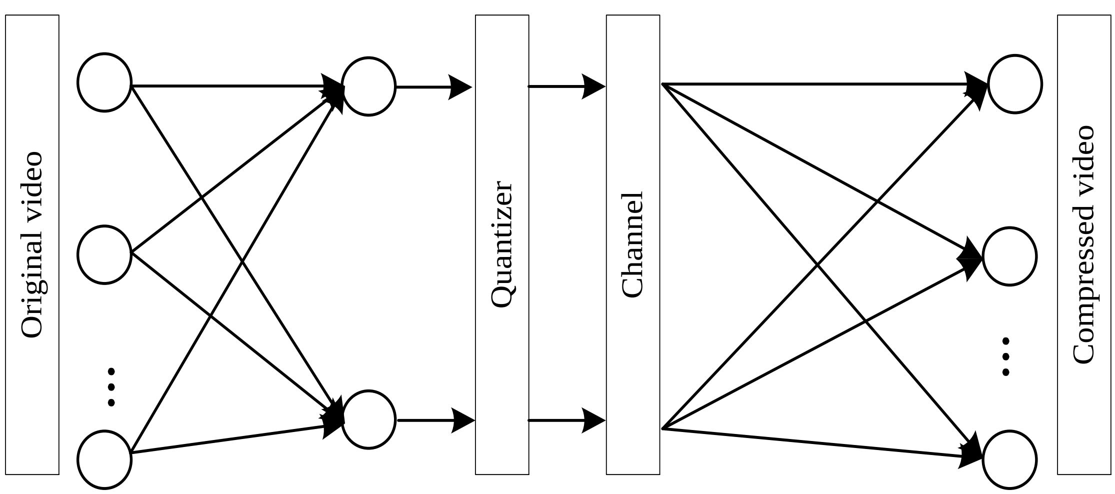

Fig. 2. VR short video compression process.

In the video cache, the most popular VR short video file ϕ_{L_p}, which is not in the original cluster, may be selected to offset the request. At the same time, a popular VR short video file is arranged and stored in the cluster head and the cluster center to facilitate user access and make the user's viewing experience better and smoother. And through constant iteration and change, users can obtain the latest VR short video information without requiring additional micro cache or auxiliary devices to simplify VR short video cache.

C. Video Transcoding

It is one of the most important applications of VR short video transcoding to generate low bit rate video streams to meet the needs of different network bandwidth [14]. Generally, the simplest and most direct way to realize VR short video transcoding is to use the so-called full decoding and full encoding scheme, that is, first decode the encoded video stream to generate a video sequence, and then re encode them to generate a new video stream. Because the recoding process requires a very complex motion estimation and mode selection process, this scheme is difficult to achieve real-time video transcoding, which greatly limits the practical application of this scheme.

To solve these problems, researchers have proposed various video transcoding structures, among which the most influential are the so-called open loop structure and closed loop structure. The open loop structure and the closed loop structure completely reuse the macroblock mode and motion vector information in the original bitstream, so there is no need for high complexity motion estimation and mode selection. Among them, the open-loop structure can directly transcode in the compressed domain (DCT domain) [15]. The transcoding speed is very fast, but due to the existence of drift error, the transcoded video frames are seriously distorted. The closed-loop structure introduces a closed loop in the encoding part to eliminate drift error, which basically eliminates the impact of drift error on the video quality after transcoding. However, the introduction of a closed loop also increases the complexity of the transcoder, making the speed of the closed-loop structure lower than that of the open loop structure. To solve these problems, this paper designs a fast video transcoder with drift error adaptive.

The closed loop for eliminating drift error in the closed-loop video transcoder can be divided into two parts, in which the part before the error buffer input completes the accumulation of drift error; the part after the error buffer output completes the drift error compensation. Suppose Y is the DCT coefficient obtained by decoding in the original code stream, and W is the DCT coefficient obtained by error compensation operation. The new quantized DCT coefficient T obtained by quantization operation is obtained as follows:

$$T = \frac{\left(Y+W\right)^2}{Q^2} \tag{9}$$

In the formula, Q^2 represents quantization operation. If $0 \le \left|Y+W\right| < Q^2$, then the non-zero error W does not affect the value of T. Therefore, if a threshold θ is given in advance, the error compensation operation can be determined according to the size of error W and threshold θ. Based on this idea, this paper proposes the working principle of the drift error adaptive fast video transcoder as shown in Fig. 3 [16]. For VR short video stream, there are three different types of coded video frames (A frame, B frame and C frame). The transcoder of this structure operates as follows:

1) The current transcoding video is A frame: no error compensation operation is required, but error accumulation operation is required.

2) The current transcoding video is a B frame: error compensation and error accumulation operations are required, but error compensation for B frame is performed in blocks.

3) The current transcoding video is C frame: in order to speed up the transcoder, C frame requires neither error compensation nor error accumulation.

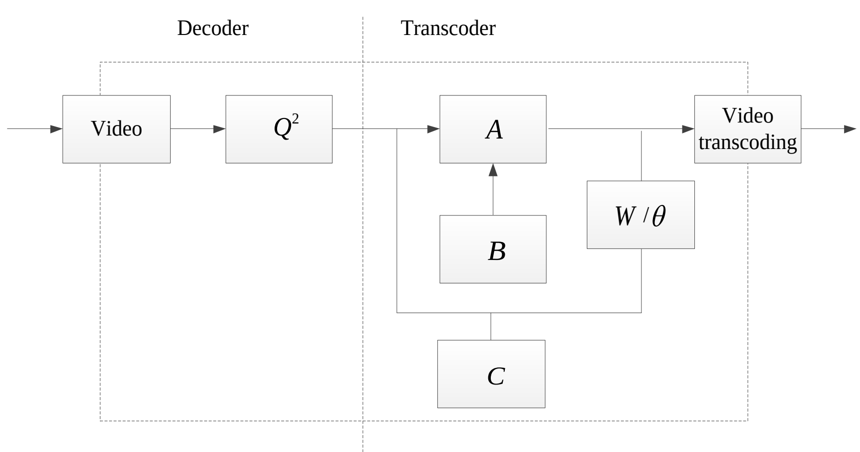

Fig. 3. Working principle diagram of video transcoder.

IV. VR Short Video Distribution Algorithm Considering Cost Optimization

The development of video applications has put forward higher requirements for video distribution in terms of real-time, mobility and interactivity. How to efficiently distribute video, alleviate the contradiction between bandwidth pressure and response delay, and reduce the cost of video distribution has become a difficult and hot spot in video distribution application research. Based on the above video compression, video caching and video transcoding results, this section proposes a cost optimized VR short video distribution method. This method can effectively reduce the bandwidth loss and response delay, thus minimizing the allocation cost. In the process of video distribution, a multicast distribution tree is built to realize the communication and video distribution between the source server and the proxy server.

A. Video Distribution Model

In the VR short video distribution framework, it mainly includes source server, proxy server and relay server. This article assumes that the source server stores all video streaming media, and each proxy server manages at least one area. Users can request and download videos from proxies that are closer to them. In order to meet the needs of users, a VR short video distribution method combining active distribution and passive distribution is adopted. The VR short video distribution model is shown in Fig. 4.

In Fig. 4: 0 represents the source server, 1 represents the relay server, 2 represents the request server, and 3 and 4 represent the proxy server. The directed bold line segment in the figure represents the VR short video distribution process; the directed virtual line segment represents the process that the proxy server sends a request to the source server when there is no video requested by the user; the black directed solid line represents the distribution link.

As shown in Fig. 4, when a user sends a video streaming media request to the local proxy server, the source server quantifies the video based on the user's interest in the video. H_ε represents the maximum number of requests for video streaming media in the streaming media distribution system, H_ω represents the number of requests for video streaming media from the proxy server, and the source server quantifies the user's interest in video streaming media in the proxy server through the H_ε / H_ω ratio.

In the active video distribution strategy, the source server distributes the video streaming media through the multicast distribution tree, and caches it to some proxy servers according to the quantitative proportion. The user directly obtains the video content from the proxy server. The passive video distribution strategy aims at sending video access requests to the proxy server by users, but when they fail to hit, the proxy server sends video requests to the source server. According to the received video request, the source server distributes the video to the request proxy server through the multicast distribution tree, and then forwards it to the requesting user. In the process of multicast distribution, some relay proxy servers can be added to minimize the cost of video distribution. VR short video distribution can effectively reduce user access delay and bandwidth loss.

B. Multicast Distribution Tree

With the rapid increase of types and quantities of VR short videos, users have put forward higher requirements for network bandwidth and access delay. In the process of VR short video distribution, this paper comprehensively considers the impact of personalized demand, link bandwidth and delay on the video distribution cost. In the VR short video distribution model, the source server distributes the video to the proxy server by building a multicast distribution tree. During the construction of multicast distribution tree, based on the 4VR short video distribution model shown in Fig. 4, some relay servers can be added to minimize the video distribution cost. The multicast distribution tree model is shown in Fig. 5, where node 0 represents the source server, node 1, 3, 6, 7, 8 represents the proxy server, and node 2, 4, 5 represents the relay proxy server. The directed line segment between two points represents the communication link between servers. The number on the directed line segment represents the link cost when bandwidth and delay are considered comprehensively. In this paper, we want to find an optimized video distribution path from the source server to the proxy servers to minimize the cost of video distribution.

C. VR Short Video Distribution Cost Minimization Algorithm

In the process of VR short video distribution, the problem of cost minimization is proposed for satisfying users' best experience of video requests and minimizing distribution costs. The performance of the video distribution network discussed in this paper does not calculate the delay and bandwidth cost between the proxy server and the end user. Since users request video from proxy servers in their regions, access time and bandwidth costs will not be affected by video copy allocation.

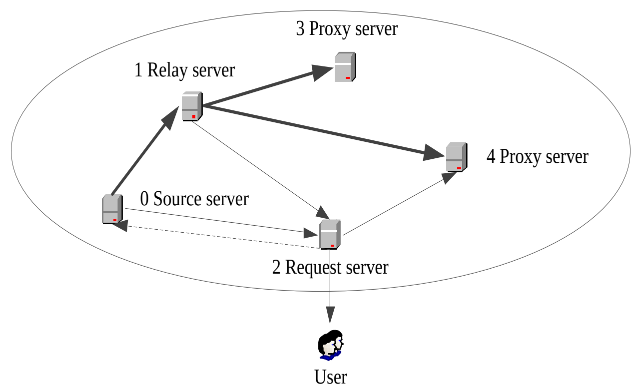

Fig. 4. VR short video distribution model.

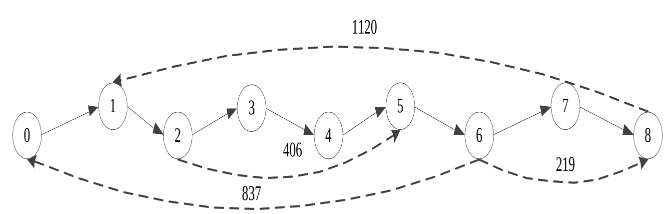

Fig. 5. Schematic diagram of multicast distribution tree model.

The video in the source server is divided into β different video streaming objects. Each video streaming media object is represented by the symbol U, so the video streaming media object set is $U=\{u_1,u_2,...,u_\beta\}$. The storage limit of each proxy server is I_a, and ϑ_a represents the number of video streaming media objects requested by proxy server a. During the active distribution of VR short videos, δ_a represents the degree of interest of users in proxy server a in video streaming media objects, and κ_a represents whether proxy server a caches video streaming media objects:

$$\kappa_a=\begin{cases}1, & a\in U\\ 0, & a\notin U\end{cases} \tag{10}$$

In the formula, 1 shows that the video object is actively distributed to the proxy server; 0 that the video object is passively distributed to the proxy server.

The cost function of VR short video distribution delay is as follows:

$$C_a=\sum_{a=1}^{n}\sum_{i=1}^{m}k^i\left(1-\kappa_a\right)-d_t \tag{11}$$

In the formula, d_t represents the unit retrieval delay of the proxy server for video streaming media objects.

The bandwidth cost function of VR short video distribution is as follows:

$$C_a^2=\sum_{a=1}^{n}\sum_{i=1}^{m}\left(\zeta_{ai}+\chi_{aj}\right)^2 \tag{12}$$

In the formula, ζ_{ai} represents the size of the video streaming media object; χ_{aj} represents the unit bandwidth of the video streaming media object requested by the proxy server.

When bandwidth and delay are considered at the same time, coefficients φ_1 and φ_2 are used to balance the delay and bandwidth costs to meet the needs of multiple applications and systems. Therefore, the cost minimization function of VR short video distribution is as follows:

$$C_{\min}^{'}=\varphi_1C_a+\varphi_2C_a^2 \tag{13}$$

The constraints of formula (13) are as follows:

$$\sum_{a,i=1}^{n}\kappa_ak^i\le\tau_i \tag{14}$$

$$\chi_{aj}\le\chi_{\max} \tag{15}$$

$$0.1\le\varphi_1<1;0.1\le\varphi_2<1 \tag{16}$$

$$\varphi_1+\varphi_2\ge1 \tag{17}$$

Formula (14) is the storage capacity limit of the proxy server. Formula (15) indicates that the link bandwidth between the proxy server and the video streaming media object cannot exceed the maximum bandwidth between two points. Formulas (16) and (17) are the constraints of balance coefficients φ_1 and φ_2.

This paper summarizes the problem of minimizing the distribution cost of VR short video, that is, giving the size of different video streaming media objects and the number of video streams requested, whether there is an allocation strategy to minimize the total cost in formula (13), and whether the video capacity stored in the proxy server is full of formula (14), the size of the link bandwidth meets formula (15), and the relationship between the balance coefficients meets the restrictions of formulas (16) and (17).

V. Experimental verification

In order to verify the application effect of personalized VR short video content distribution algorithm based on artificial neural network, experimental analysis is carried out. The methods in literature [5] and literature [6] are used as comparison methods and compared with the proposed methods.

A. *Experimental Samples and Indicators*

In the simulation experiment, the experimental hardware platform is Pentium IV 2.4GHz CPU with 8GB of memory. The experimental data comes from the COIN (Comprehensive Interactive Video Analysis), an open-source tutorial video dataset, which contains 11827 tutorial videos involving 180 tasks in 12 fields of daily life. Video test sequences are selected from this dataset, all of which are standard test sequences. When analyzing the performance of simulation results, this paper uses Matlab software to convert the experimental data comparison into specific images, which provides strong evidence for the conclusions of this paper.

When comparing the proposed method with traditional methods, the following three performance indicators are mainly considered:

1) Stuck times: the number of times a video stream is stuck during transmission and playback.

2) PSNR: A function of the square difference of pixels between the original video file and the received video file, which is a standard objective indicator to measure the video picture quality.

3) Transcoding rate: it can directly reflect the complexity of video transcoding and indirectly reflect the rate of video distribution.

B. *Analysis of Experimental Results*

1) Stuck times: The comparison results of the stuck times of the methods in literature [5], literature [6] and the proposed method in the video distribution process are shown in Fig. 6.

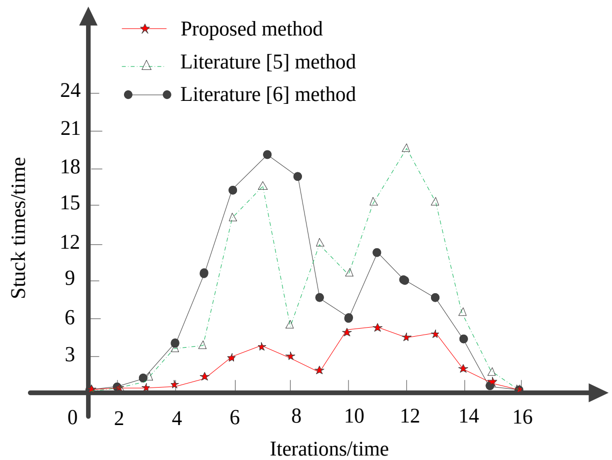

Fig. 6. Comparison results of stuck times.

2) PSNR: The PSNR comparison results of literature [5] method, literature [6] method and the proposed method are shown in Fig. 7.

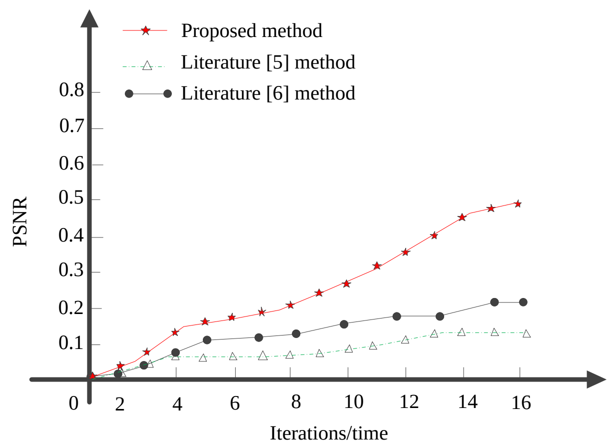

Fig. 7. Comparison results of PSNR.

3) Transcoding rate: Fig. 8 shows the video transcoding rate comparison results of three different methods.

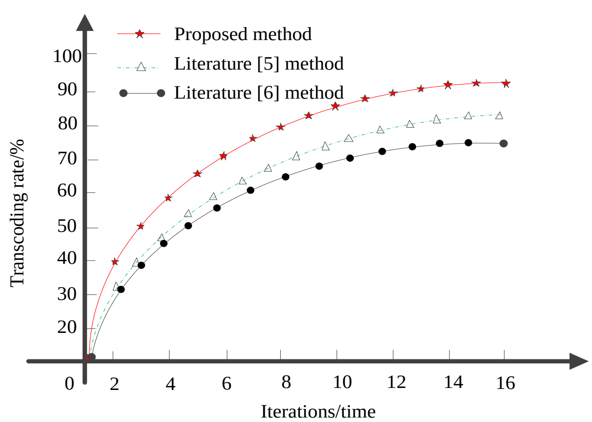

Fig. 8. Transcoding rate comparison results.

VI. Discussion

Through the above experiments, conclusions can be drawn as follows:

1) As shown in Fig. 6, the number of video clicks of the proposed method is always less than six times, while the maximum number of clicks of the method in literature [5] reaches 20 times, and the maximum number of clicks of the method in literature [6] reaches 19 times. Compared with traditional methods, the proposed method can provide users with a higher quality user experience during video distribution, that is, improve the video quality level of users when the video is played smoothly. This is because the design method can select the most popular VR short video files not in the original cluster in the video cache to offset the request. At the same time, the copies of popular VR short video files are stored in the cluster header to facilitate user access, thus making the user's viewing experience smoother.

2) It can be seen from Fig. 7 that although the PSNR values of the three methods have an overall improvement trend, the variation range of the methods in literature [5] and literature [6] is small. The PSNR value of this method is high, with a large increase, up to 0.5, indicating that the average video quality level obtained by users is high, indicating that the video distribution quality is high. The reason for this result is that the design method uses BP neural network to compress the original video and determine the implementation mode and cache location of VR short video cache files; Convert and cache high bit rate VR short video streams to generate low bit rate video streams, which not only meets the needs of different network bandwidth, but also improves the PSNR value.

3) It can be seen from Fig. 8 that the video transcoding speed of this method is significantly higher than that of [5] and [6], and the maximum can reach 92%. There are two main reasons for this result: on the one hand, the new coding coefficients and drift errors in the conversion encoder structure are calculated mainly through look-up tables rather than complex multiplication and division operations, which greatly reduces the overhead of the converter and improves the speed of the converter; On the other hand, the improvement of the speed of fast transcoding structure also comes from the selective error compensation operation of video frames. If the error compensation operation is performed, the table lookup structure cannot be used to calculate new coding coefficients and drift errors. Instead, the table lookup structure can be used to perform corresponding operations. It can be seen that the adaptive drift error control method greatly reduces the complexity of video frame transcoding, and can also improve the speed of video transcoding.

VII. Conclusion

In order to meet the requirements of high quality and low delay in short video development and reduce network load, this paper proposes a personalized distribution algorithm of VR short video content based on artificial neural network. Firstly, the original video is compressed by BP neural network, and the implementation method and cache location of VR short video cache file are determined; Secondly, the cached high bit rate VR short video streams are converted to generate low bit rate video streams to meet the needs of different network bandwidth; Third, establish a video distribution model, and build a multicast distribution tree based on this model, that is, add some relay servers; Finally, the minimum allocation cost of VR short video is realized through the algorithm of minimizing the allocation cost of VR short video. The experimental results show that the video blocking times of this method are always less than six times, which effectively reduces the video blocking times. The PSNR value is high, the increase is large, up to 0.5, and the video transcoding rate is improved, up to 92%. The application effect of this method is fully verified, which shows that it has certain application value. However, in the process of experimental analysis, the design method has not been compared with other more methods, and there are still some deficiencies. In the next research process, supplementary research will be carried out to further improve the application effect of this method.

References

[1] Cicco L D, Mascolo S, Palmisano V. QoE-driven Resource Allocation for Massive Video Distribution. Ad Hoc Networks, 2019, 89(6):170-176.

[2] Liu W, Zhang T. Joint incentive mechanism based on video content cooperative delivery in mobile edge computing. Application Research of Computers, 2021, 38(9):2803-2810.

[3] Li Y Y, Zhou J H. Integrated ICN Slice and Video Caching as a Transmission Service. Telecommunication Engineering, 2022, 62(4):510-516.

[4] Zhang S H, Luo Y W, Zhang M, Sun Q F, Yang X L. Research progress of mobile video prefetching for mobile content distribution networks. Telecommunications Science, 2021, 37(1):8-21.

[5] Dong W T, Li Z, Chen X. Online short video content distribution strategy based on federated learning. Journal of Computer Applications, 2021, 41(06):1551-1556.

[6] Wang R Y, Yang Y, Wu D P. QoE-aware video distribution mechanism in FiWi. Journal on Communications, 2018, 39(01):1-13.

[7] Chen X, Wang L, Sun Z W. Research on Internet of Vehicles D2D Video Distribution Technology Based on Network Coding. Computer Engineering, 2019, 45(02):58-63.

[8] Pereira S, Abreu E, Iakunin M, Cavaco A, Salgado R, Canhoto P. Method for solar resource assessment using numerical weather prediction and artificial neural network models based on typical meteorological data: Application to the south of Portugal. Solar Energy, 2022, 236(4):225-238.

[9] Altintas E, Tolon M, Karabuga A, Utlu Z. Application of artificial neural network model for forecast energy efficiency of the cryogenic liquefaction system in the meaning of sustainability. International Journal of Global Warming, 2021, 24(3/4):420-444.

[10] Chen W, Li X, Chen X, Xiong Y. Research on influence mechanism of running clothing fatigue based on BP neural network. Journal of Intelligent and Fuzzy Systems, 2020, 40(8):1-11.

[11] Cao Q, Zhou C. Adaptive Compression Method of Embedded Online Video Inter Frame Signal. Computer Simulation, 2021, 38(3):87-91.

[12] Kafiloglu S S, Gur G, Alagoz F. Cooperative Caching and Video Characteristics in D2D Edge Networks. IEEE Communications Letters, 2020, 24(11):2647-2651.

[13] Wu D, Liu Q, Wang H, Yang Q, Wang R. Cache Less for More: Exploiting Cooperative Video Caching and Delivery in D2D Communications. IEEE Transactions on Multimedia, 2019, 21(7):1788-1798.

[14] Yoon J, Banerjee S. Hardware-assisted, Low-cost Video Transcoding Solution in Wireless Networks. IEEE Transactions on Mobile Computing, 2020, 19(3):581-597.

[15] Ahn W, Nam S, Son M, Lee H, Choi S. End-to-end double JPEG detection with a 3D convolutional network in the DCT domain. Electronics Letters, 2020, 56(2):82-85.

[16] Lv Y, Zhu B, Wu D P. Video multipath transmission mechanism with load balancing in the FiWi network. Journal of Xidian University, 2021, 48(3):63-70.