

## RESEARCH ARTICLE

# An Intelligent Retrieval Method for Audio and Video Content: Deep Learning Technology Based on Artificial Intelligence

MAOJIN SUN 

CEICloud Data Storage Technology (Beijing) Company Ltd., Beijing 101111, China

e-mail: samuelconz1@mail.com


**ABSTRACT** To address the challenges of efficient intelligent retrieval and cross-modal analysis brought by the surge in audio-video data, this study proposes an intelligent retrieval method for audio-video content based on deep learning techniques, aimed at improving retrieval efficiency and accuracy. This method extracts audio features using the Visual Geometry Group Network (VGG) and employs an adaptive clustering keyframe extraction algorithm (SKM) to extract video features. By integrating cross-learning within an embedding network, it enhances retrieval efficiency and accuracy. The test results on the CMU-MOSEI dataset demonstrate that our method outperforms traditional models such as Principal Component Analysis (PCA), Canonical Correlation Analysis (CCA), and state-of-the-art deep learning models like Deep Canonical Correlation Analysis (DCCA) and Domain-Adversarial Neural Network (DANN) in multimodal data processing and real-world retrieval tasks. In video processing, the average fidelity is 0.693, and the average compression ratio is 0.936, representing improvements of 30.75% and 7.09%, respectively, compared to traditional methods. Through the application of deep learning technology, this study not only optimizes the processing of single modalities but also enhances the handling of cross-modal data through a cross-learning framework.

**INDEX TERMS** Audio-video content retrieval, deep learning, feature extraction, cross-modal retrieval, intelligent retrieval.

## I. INTRODUCTION

As the core of the next generation of mobile communications, 5G technology has demonstrated immense potential and influence worldwide since its commercialization in 2019. The ultra-high speed, ultra-low latency, and massive connectivity capabilities of 5G communication networks provide robust technical support for emerging applications such as the Internet of Things (IoT), smart cities, and autonomous driving. The academic community has conducted extensive research on 5G applications. Tyokighir et al. provide a detailed analysis of the latest advancements in 5G mobile communication technology and systematically introduce the application prospects of technologies

such as millimeter-wave (mmWave), massive multiple-input multiple-output (MIMO), small cells, and mobile edge computing (MEC) [1]. In the field of 5G mobile communications, Xia et al. (2024) proposed a novel mmWave tilted beam phased array antenna to address the shielding issue of mmWave radiation by the metal frame of smartphones. This study resolves the impedance mismatch and radiation distortion issues in smartphones, significantly enhancing the performance of 5G smartphones [2]. In terms of 5G network management and optimization, Yeh et al. (2024) explored the application of deep learning techniques in 5G open RAN, proposing an intelligent network application (xApp) for network slicing, achieving automated and intelligent deployment while maintaining service level agreements (SLA) [3]. Regarding 5G communication network positioning technology, Zhou et al. proposed an indoor positioning

The associate editor coordinating the review of this manuscript and approving it for publication was Renato Ferrero .

method utilizing the multi-beam characteristics of a single 5G base station. The study shows that this method significantly improves positioning accuracy in various indoor scenarios, with an average absolute error of 1.55 meters, a 54.7% improvement over traditional single-beam methods. This research provides new insights for the application of 5G technology in indoor positioning, demonstrating the potential of 5G in enhancing positioning accuracy and reliability [4].

In the development of 5G technology applications, the internet and mobile internet have notably benefited, with significantly increased network speeds for smartphones and continuous advancements in emerging internet media technologies [5]. The number of self-media platforms has increased, making the self-media market more active [6]. These factors collectively reduce the difficulty of audio-video content creation and enhance convenience, greatly stimulating the enthusiasm of content creators [7]. Consequently, the volume of audio-video content data we face has also increased, and the diversification of audio-video content imposes new requirements on regulation [8]. Audio-video content retrieval, as a key aspect of regulation, is not only a necessary infrastructure for multiple audio-video portal websites but also a core business component for relevant government departments [9]. Nonetheless, many audio-video content retrieval tasks still rely on manually established retrieval dictionaries [10]. Especially when searching for specific individuals, specific scenes in a large number of videos, or specific sentences in extensive voice content, there is a lack of mature solutions. The application of artificial intelligence analysis shows potential to address these issues [11]. However, limited by technical capabilities and development costs, relevant entities have yet to form a mature solution based on artificial intelligence analysis. Therefore, this study proposes to introduce deep learning technology into the retrieval of audio-video content to address the challenges of rapid and accurate retrieval, developing an intelligent retrieval method for both audio and video content, thereby improving the efficiency of audio-video content retrieval.

Research on audio and video content retrieval has been explored by computer scientists both domestically and internationally since the early stages. Initial retrieval methods were based on visual features within multimedia files, such as color, lines, and human posture, to extract characteristics like color distribution, proportion, texture, shape, and angles [12], [13]. A landmark achievement in the field of audio and video content retrieval is IBM's (International Business Machines Corporation) QBiC (Query By Image Content) system, introduced in the 1990s. This system focused on the retrieval of graphic and video content [14]. QBiC not only included image and video search capabilities but also pioneered the commercialization of content-based image retrieval technology. The system first analyzed the input image or video frames, extracting features such as outlines, object textures, colors, structures, and shading. It then selected the most appropriate query method based on user

preferences for highly matched feature processing. In addition to QBiC, other systems such as Excalibur's Retrieval Wave [15], Virage's Virage [16], and Columbia University's VisualSEEK [17] adopted advanced retrieval concepts. These systems transformed images into features based on human vision, constructing highly correlated indices to accomplish retrieval tasks. Tsinghua University's TV-FI system [18] and Microsoft Research Asia's iFind system [19] demonstrated robust comprehensive performance and advanced technology. These systems utilized advanced technologies such as MIRC (Medical Imaging Resource Center) and MIREX (Maritime Information Retrieval and Exchange System), achieving remarkable results in text, image, and video retrieval domains.

In practical applications, various technical approaches have emerged in the field of audio and video content retrieval. On the theoretical research front, scholars have conducted in-depth discussions on audio and video content retrieval, although existing studies are often relatively independent. In audio content retrieval, research mainly focuses on two primary directions: content-based retrieval and template-based retrieval. Content-based retrieval methods emphasize classification and recognition using high-level information from audio. This includes keyword retrieval and audio indexing research, aiming to achieve accurate classification and retrieval through analyzing audio content characteristics [20]. Specific applications in this area include the new deep video action clustering network proposed by Peng et al. and the Group Discovery Machine (GDM) filter-based feature selection algorithm [21]. Xu et al. discussed the applicability of collaborative representation learning methods in their study, providing new insights into cross-learning framework research for audio feature extraction [22]. Vujošević and Dukanović, contributed to audio classification and annotation, classifying audio data into pure speech, non-pure speech, audio, and environmental sounds, and establishing a set of audio classification annotation standards [23]. On the other hand, template-based retrieval methods, or fixed audio retrieval, involve providing a predefined audio template and searching for the most similar audio segments in the audio library, returning these as retrieval results to the user. This method is particularly effective in practical applications, especially in scenarios with clear search targets [24]. In specific research and system implementations, Hao et al. (2021) used Long Short-Term Memory (LSTM) networks and attention mechanisms to construct a Chinese isolated word recognition system based on Triphones. This study enhanced the model's accuracy in recognizing audio slices, achieving more effective speech recognition and information retrieval [25]. Furthermore, Xie et al. (2023) evaluated the relevance of text-based audio retrieval through crowdsourced evaluations. The study results indicated that using binary relevance generated from audio descriptions in contrastive learning is sufficient for effective audio retrieval without the need for crowdsourced evaluations [26]. These studies not

only advance audio retrieval technology but also provide more possibilities for audio data management, classification, and utilization, while reducing misrecognition rates and improving the accuracy of audio classification.

Compared to research on audio content retrieval, video content retrieval presents more significant challenges and has undergone various stages of development with technological advancements. Early methods for video content retrieval typically relied on low-level visual features, such as color and texture features [27]. As research progressed, retrieval techniques gradually shifted from static low-level visual features to high-level motion features [28]. Liu et al. (2021) proposed a perceptual quality model based on reference reduction and applied it to point cloud-based video compression rate control. This model enhanced the perceptual quality of video point cloud compression by reducing reference information, effectively controlling the compression rate and improving the efficiency and quality of video transmission [29]. Zhong and Chang used optical flow methods to simulate motion vectors for each pixel and clustered pixel trajectories to locate regions of relevant motion in video content [30]. Sikha & Soman (2021) explored the use of saliency maps extracted using a dynamic mode decomposition framework. By highlighting attention-grabbing parts of images and combining them with a salient edge detection model, this method accurately identified image texture and color features, thereby constructing high-dimensional feature vectors for image retrieval [28]. Hu & Li (2023) proposed a robust visual SLAM (Simultaneous Localization and Mapping) system. This system employed ORB-SLAM2 (Oriented FAST and Rotated BRIEF Simultaneous Localization and Mapping 2) for a two-stage coarse-to-fine tracking process to improve system localization accuracy in dynamic environments [31]. Zheng et al. (2024) conducted an empirical study on the correlation between the fairness of deep neural networks and neuron coverage criteria. Their findings indicated a significant correlation between neuron coverage criteria and the fairness of deep neural networks, providing new perspectives and methods for evaluating the fairness of deep learning models, which is crucial for improving the accuracy of video retrieval [32].

Additionally, methods that combine static visual features with high-level motion features have been developed to enhance the accuracy of video content retrieval. Lu et al. (2021) proposed a distance-based video anomaly detection method using Locality-Sensitive Hashing (LSH) to map similar samples into the same bucket. This method integrates optimized hash functions and contrastive learning strategies, allowing semantically similar samples to be closer, effectively achieving video anomaly detection in dynamic environments and with imbalanced data [33]. Wu et al. (2023) proposed a real-time stereo matching method based on spatial attention-guided upsampling. By introducing a spatial attention mechanism, this method improved the accuracy of stereo matching and enabled real-time processing [34].

Jin et al. introduced an unsupervised discrete hashing (UDH) method, which optimizes binary constraints in an unsupervised framework using graph-based semantic loss and orthogonal consistency loss, effectively mitigating the impact of quantization errors [35]. The advantage of this unsupervised hashing algorithm lies in its ability to train the hash mapping function without requiring labeled data for supervised learning.

With advancements in deep learning technology, the field of video content retrieval has undergone significant innovations, particularly in key frame extraction and feature matching. Deep learning has made it more efficient and accurate to automatically extract key frames from videos and analyze their image features [36]. Naik & Soni (2021) developed a 3D convolutional neural network for video classification that simultaneously learns the spatial and temporal features of video frames, offering particular advantages for video content classification. This method demonstrated outstanding performance and efficiency when handling complex video datasets [37]. Furthermore, Kızıltepe et al. (2021) proposed a model combining convolutional neural networks (CNN) and recurrent neural networks (RNN). This model optimizes video classification performance by identifying informative regions in each frame and selecting key frames based on the similarity of these regions, significantly improving video classification accuracy and proving the method's effectiveness in practical applications [38]. Chen et al. (2023) introduced a traffic prediction method based on visual quantization features, achieving accurate traffic flow predictions by extracting visual quantization features. This method holds high reference value for applications of visual feature extraction technology [39]. Additionally, the disparity multi-scale fusion network detection method proposed by Chen et al. is valuable for research on multimodal data processing and the extraction of audio and video features in a shared feature space [40]. Pan et al. (2023) studied explainable multimodal neural networks, achieving accurate estimations of gamer engagement by integrating multimodal information such as video, audio, and text. This estimation method provides an effective reference for implementing multimodal retrieval tasks [41]. These studies demonstrate that the application of deep learning, particularly convolutional neural networks, in video key frame extraction and content retrieval can significantly enhance efficiency and accuracy.

Although substantial research has been conducted on audio-visual content retrieval, there remains a significant research gap in handling cross-modal data. Current methods predominantly rely on single-modal feature extraction, which fails to fully exploit the multimodal characteristics of audio-visual data, thereby limiting retrieval efficiency and accuracy. Additionally, existing deep learning models face challenges in computational resources and time costs when processing large-scale data, lacking effective real-time retrieval solutions. Furthermore, many current approaches overly depend on metadata, such as keywords, titles, or descriptions, which

prove impractical in large-scale datasets [42]. Moreover, less popular multimedia data often lack corresponding meta-data. Commonly used mapping functions in metadata-based retrieval methods are typically rigid and fixed, employing hard-coded mechanisms, which constrain their broader applicability [43].

Addressing this issue, this paper proposes a processing framework based on AI-driven error correction and collaborative video super-resolution to tackle the problem of enhancing degraded video quality. Innovatively, this research combines the VGG network-based audio feature extraction method with the adaptive clustering keyframe extraction algorithm (SKM) for video feature extraction, addressing the limitations of single-modal feature extraction. Additionally, an embedding network is designed to map audio and video features into a shared feature space, enabling comparative analytical learning across modalities and significantly improving the handling of cross-modal data. The design of this embedding network is innovative in its effective integration of multimodal data, particularly by introducing cross-modal contrastive learning, which further enhances the model's retrieval performance. Experimental data demonstrate that this approach effectively improves the efficiency and quality of audio-visual content retrieval, offering a new perspective for audio-visual content retrieval.

The structure of this paper is arranged as follows: Firstly, the research background, objectives, and the main challenges currently faced in the related field are introduced, along with the proposed research methods. Subsequently, a review of the cutting-edge research in this domain is conducted. Following this, the paper provides a detailed description of the key components, such as the network structure of the AI-driven error correction and collaborative video super-resolution framework. In the results analysis section, the effectiveness and usability of the proposed method are validated through comparisons with various approaches under different dataset conditions. The discussion section compares and contrasts the proposed method with existing methods in the field, highlighting its differences and advantages, and outlining its research contributions. Finally, the conclusion section provides a comprehensive summary of the research findings and innovative approaches to video quality enhancement, identifies the limitations of the study, and suggests key directions for future improvement and research.

## II. DESIGN OF AUDIO-VISUAL CONTENT RETRIEVAL MODEL

Building on existing research, this chapter will provide a detailed introduction to our proposed intelligent retrieval model for audio-visual content. Utilizing deep learning techniques, we aim to achieve more efficient and accurate retrieval of audio-visual content through the extraction and integration of audio and video features. The following sections will systematically elaborate on various aspects of the model design, including the audio feature extraction algorithm, the video feature extraction algorithm, the design

of the audio-visual embedding network, and the optimization methods for the loss function.

### A. AUDIO FEATURE EXTRACTION ALGORITHM

After inputting the audio signal, a frame is extracted per second, and the VGG network is used to analyze and extract features from each frame of the audio signal. Studies have shown that consecutive audio segments often exhibit similar characteristics, such as emotional attributes. Next, the extracted audio feature sequence is evenly divided into  $t$  data blocks, and the emotional information of each data block is assessed. Finally, the  $s$  data blocks with the most prominent emotional attributes are selected to represent the features of the entire audio for subsequent training. The detailed process is shown in Figure 1.

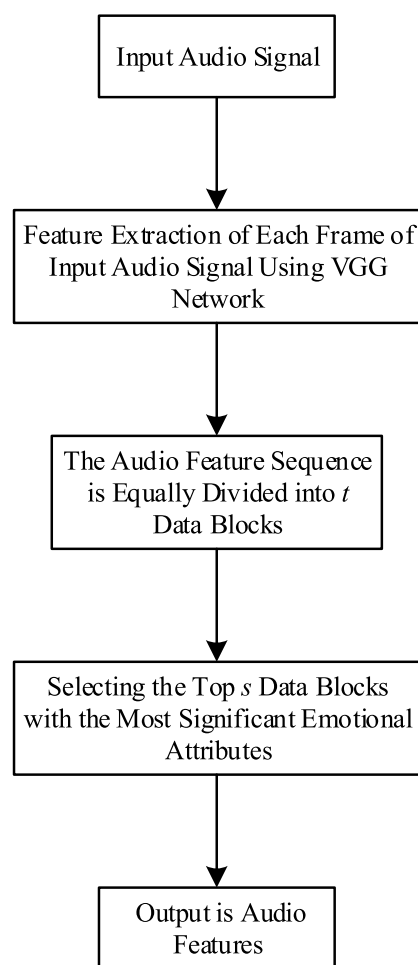


FIGURE 1. Basic process of the audio feature extraction algorithm.

Additionally, the model consists of two main components: one part is the Long Short-Term Memory (LSTM) network with a bidirectional extension mechanism, whose specific computational model is described below; the other part is the attention mechanism-based computational layer. The LSTM network maintains long-term gradient flow through an autoregressive mechanism, with its internal weights being

updated in real-time based on contextual information. This dynamic adjustment according to the input sequence is achieved by the following nodes within the LSTM structure.

(1) The update of the input unit state depends on the current input vector  $x_t$  and the previous hidden state  $h_{t-1}$ . The specific formula is as follows:

$$s_t = \sigma(b_i + W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1}) \quad (1)$$

In the equation,  $s_t$  represents the cell state at the current time step;  $\sigma$  denotes the sigmoid activation function, which controls the flow of information;  $b_i$  is the bias vector for the input unit;  $W_{xi}$  is the weight matrix of the input unit for the current input  $x_t$ ;  $x_t$  is the input vector at the current time step, representing the input data;  $W_{hi}$  is the weight matrix of the input unit for the hidden state  $h_{t-1}$  from the previous time step;  $h_{t-1}$  is the hidden state vector from the previous time step, containing information passed from the previous time step;  $W_{ci}$  is the weight matrix of the input unit for the cell state  $c_{t-1}$  from the previous time step;  $c_{t-1}$  is the cell state vector from the previous time step, containing information passed from the previous time step.

(2) The function of the forget gate is to control the information passed from the previous cell state  $o_t$ , determining the amount of information to retain or discard. The calculation formula is as follows:

$$f_t = \sigma(b_f + W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1}) \quad (2)$$

In the equation,  $f_t$  represents the information retention degree of the forget gate;  $\sigma$  denotes the sigmoid activation function, which controls the flow of information;  $b_f$  is the bias vector for the forget gate;  $W_{xf}$  is the weight matrix of the forget gate for the current input  $x_t$ ;  $x_t$  is the input vector at the current time step;  $W_{hf}$  is the weight matrix of the forget gate for the hidden state  $h_{t-1}$  from the previous time step;  $h_{t-1}$  is the hidden state vector from the previous time step;  $W_{cf}$  is the weight matrix of the forget gate for the cell state  $c_{t-1}$  from the previous time step;  $c_{t-1}$  is the cell state vector from the previous time step. This process ensures that the network can forget unnecessary accumulated information, thereby optimizing the memory process.

(3) The update formulas for the output unit  $o_t$  and the hidden state  $h_t$  are as follows:

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (3)$$

In the equations,  $o_t$  represents the output vector;  $\sigma$  is the activation function;  $W_{xo}$  is the weight matrix for the output unit;  $x_t$  is the input vector at the current time step;  $W_{ho}$  is the weight matrix for the hidden state of the output unit;  $h_{t-1}$  is the hidden state vector from the previous time step;  $W_{co}$  represents the current state of the output unit;  $c_t$  is the cell state vector at the current time step;  $b_o$  is the bias vector for the output unit.

$$h_t = o_t \tanh(c_t) \quad (4)$$

In the equation,  $h_t$  represents the hidden state vector at the current time step;  $o_t$  denotes the output vector;  $\tanh$  is the

activation function;  $c_t$  is the cell state vector at the current time step.

In the first layer of this model, the activation function acts as a feature "scoring" mechanism, providing a quantitative assessment of the concentration of features in the input data. The specific calculation formula can be expressed as follows:

$$u_t = W^T \tanh(W_f h_{ff} + W_b h_{tb} + \beta) \quad (5)$$

In the formula,  $h_{ff}$  and  $h_{tb}$  represent the forward and backward output vectors of the LSTM network, respectively, while  $W^T$ ,  $W_f$ ,  $W_b$ , and  $\beta$  are the weights and bias parameters of the scoring function. This scoring mechanism effectively identifies key information in the data by evaluating the concentration of features.

The second layer of the network uses the softmax function to process the output scores from the first layer, converting these scores into values between 0 and 1, which reflect the concentration of features in each data block. The formula is as follows:

$$p_t = \frac{e^{u_t}}{\sum_k e^{u_k}} \quad (6)$$

In the formula,  $p_t$  represents the probability distribution after being processed by the softmax function, used to measure the feature concentration of each data block relative to others;  $u_t$  is the quantitative assessment result of the feature concentration of the input data;  $e^{u_t}$  is the exponent of the quantitative assessment result of the feature concentration of the input data;  $\sum e^{u_k}$  is the sum of the exponents of the feature scores of all data blocks, serving as the normalization factor. This approach allows the model to identify and select data blocks with the highest feature concentration, thereby improving the overall accuracy and efficiency of the experiment.

## B. VIDEO FEATURE EXTRACTION ALGORITHM

This study proposes an improved k-medoids clustering algorithm named SKM(SOFM-k-medoids). The algorithm features a preprocessing step that automatically calculates the number of clusters and the initial cluster centers, which are directly applied in the k-medoids clustering process. Initially, the algorithm performs feature analysis on image frames using a developed image saliency region deep feature extraction algorithm. This step allows for the automatic determination of the number of clusters based on changes in inter-frame similarity, addressing the issue of inaccurate keyframe counts that can arise from a fixed number of clusters in traditional methods. Subsequently, the SOFM algorithm is used to pre-cluster the initial data, yielding more precise cluster centers. This approach mitigates the problem of slow convergence and excessive iterations caused by improper initial cluster center selection. After the preprocessing stage, the determined number of clusters and cluster centers serve as input parameters for the k-medoids algorithm, which then performs the final clustering operation. By optimizing parameter settings through preprocessing, this method effectively

avoids performance fluctuations caused by manually setting the number of clusters and cluster centers, thereby significantly enhancing the efficiency and stability of the clustering algorithm.

This study utilizes deep features of salient regions in images to measure inter-frame similarity. The specific implementation steps are as follows:

(1) For the input of  $n$  frames, the image saliency region feature extraction algorithm is applied to each frame individually to extract the deep features of the salient regions  $X_i$  of each frame, representing the features of each frame.

(2) Cosine distance is used to calculate the distance  $D = \{d_1, d_2, \dots, d_{n-1}\}$  between each frame and its adjacent frame. The specific distance calculation formula is as follows:

$$d_i = \text{distance}(X_i, X_{i+1}) \quad (7)$$

In the formula,  $d_i$  represents the cosine distance between the  $i$ th frame and its adjacent frame;  $\text{distance}$  is the function used to calculate the cosine distance between two frames.

A statistical analysis is conducted on the distance values in the set  $D$  to compute the mean  $\mu$  and standard deviation  $\sigma$  of the inter-frame similarity, using the following formulas:

$$\begin{cases} \mu = \frac{1}{n-1} \sum_{i=1}^{n-1} d_i \\ \sigma^2 = \frac{1}{n-1} \sum_{i=1}^{n-1} (\mu - d_i)^2 \end{cases} \quad (8)$$

In the formulas,  $\mu$  represents the mean of inter-frame similarity;  $\sigma$  is the standard deviation of inter-frame similarity;  $n$  is the total number of frames.

Based on the comparison between the inter-frame similarity value  $d_i$  and the threshold  $\mu + \delta\sigma$ , the moments of content change are determined.  $\delta$  is a preset parameter used to adjust sensitivity. Each time  $d_i$  exceeds the set threshold, it indicates a significant change in frame content. Finally, the number of detected content changes is used as the number of clusters  $K$ . This method effectively adapts to the changes in video content to determine the number of clusters, improving the accuracy and efficiency of keyframe extraction.

When using SOFM for clustering, the algorithm process mainly includes the following steps:

(1) Adaptively determine the number of neurons in the output layer, which is the number of clusters  $K$ . The topological structure of the output layer is designed as  $K \times 1$ , and the weight parameters of each neuron are initialized to be consistent with the input layer dimensions.

(2) Process the deep feature vectors  $X_i$  of each image region. By comparing the similarity between the weight vectors of the neurons and the input data, the neuron with the highest similarity is identified as the winning neuron.

(3) After determining the winning neuron, calculate the weight update values for its neighboring neurons. The update value is based on the distance between two neurons and is inversely proportional to the distance to the winning neuron.

The calculation expression is as follows:

$$T_{j,N} = \exp(-S_{ij}^2/2\sigma^2) \quad (9)$$

In the formula,  $T_{j,N}$  represents the weight vector of the neurons surrounding the winner;  $s_{ij}$  is the distance between neuron  $i$  and neuron  $j$ .

(4) Based on the above calculations, first update the weights of the winning neuron, then gradually update the weights of its neighboring neurons as follows:

$$\Delta w_{ji} = \eta(t)T_{j,N}(t)(x_i - w_{ji}) \quad (10)$$

In the formula,  $\Delta w_{ji}$  represents the weight update amount for the winning neuron  $j$  with respect to the input vector  $i$ ;  $\eta(t)$  is the learning rate;  $T_{j,N}(t)$  is the neighborhood function of neuron  $j$  at time  $t$ ;  $x_i$  is the input vector  $i$ ;  $w_{ji}$  is the weight vector of neuron  $j$  at the input vector  $i$ .

(5) If the iteration limit is reached or the neural network stabilizes, proceed to the next step; otherwise, return to step (2) and continue.

(6) Finally, according to the clustering results  $C = \{C_1, C_2, \dots, C_K\}$  of SOFM, obtain the center  $u_i$  of each cluster.

Through the above steps, SOFM can adaptively compute the cluster centers based on the input data without presetting the cluster centers.

The goal of the k-medoids clustering algorithm is to divide a set of image feature sets into a predetermined number of  $k$  clusters, maximizing the similarity within the same cluster and minimizing the similarity between clusters. The algorithm achieves this by minimizing the total squared error function (SSE), which represents the sum of the distances between the image feature vectors and their cluster centers across all clusters. The specific expression is as follows:

$$SSE = \sum_{i=1}^k \sum_{x \in C_i} \| \text{distance}(x, u_i) \|^2 \quad (11)$$

In the formula,  $x$  represents the image feature vector;  $C_i$  corresponds to each cluster;  $u_i$  is the cluster center;  $\text{distance}(x, u_i)$  represents the cosine distance between  $x$  and  $u_i$ .

The algorithm flow is illustrated in Figure 2, and the specific content is described as follows:

(1) Obtain the initial cluster centers from the SOFM clustering algorithm results  $\{u_1, u_2, \dots, u_k\}$ , use these centers as the initial cluster centers for the k-medoids algorithm, and adaptively determine the number of clusters  $K$ ;

(2) Initialize each cluster  $C$  to  $C_i = \phi, i = 1, 2, \dots, K$ ;

(3) Traverse each image feature vector  $X_i$ , calculate its cosine distance to each cluster center, and assign it to the cluster with the nearest center  $X_i$ ;

(4) For each cluster, compute the pairwise distances of all points within the cluster and select the image feature vector that minimizes the total distance to the cluster center as the new cluster center;

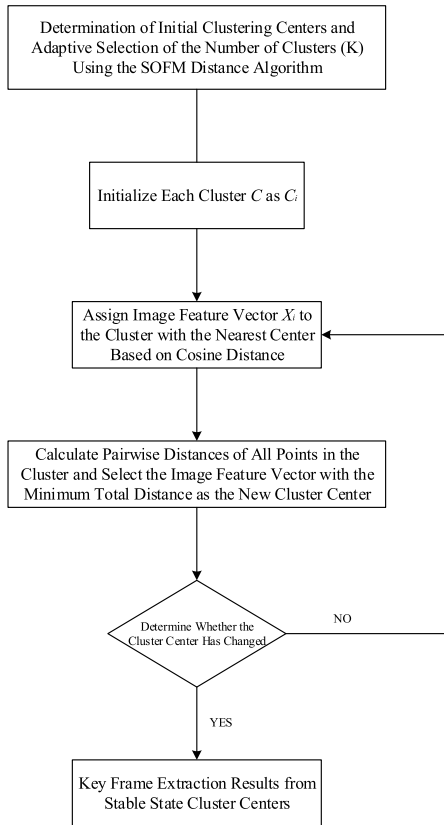


FIGURE 2. Flowchart of video feature extraction.

(5) If all cluster centers remain unchanged, the clustering has reached a stable state; proceed to the next step, otherwise, return to step (3);

(6) After the iteration is complete, use the cluster centers of the K clusters as the result for keyframe extraction.

### C. AUDIO AND VIDEO EMBEDDED NETWORK DESIGN

After performing feature selection on audio and video data, an embedding network was designed to enable comparative analytical learning of these two different types of data. This network aims to map audio and video feature vectors into a shared feature space using neural network techniques, hence referred to as the embedding space. The network primarily comprises three components: a label prediction classifier, a sample mining network, and a feature mapping network. The inclusion of the label prediction classifier aims to maximize the proximity of similar data and the separation of different data in the space when both modalities are mapped into the common subspace, achieving the principle of “similarity closeness and difference separation.” Through this design, each modality retains its semantic information in the common space, maintaining semantic distinction. This ensures that cross-modal data of the same type can match each other, while different types can highlight their differences, thereby ensuring the accuracy of matching while increasing its flexibility and diversity.

In the subspace network embedding, a feedforward network with a softmax activation function is introduced as a classifier. The core function of this structure is to process the input audio and video features. Based on the input training data, this classifier outputs the probability distribution of the semantic categories corresponding to each data item. The expression for the semantic discrimination loss function within the classifier modality is as follows:

$$L_{lp} = -\frac{1}{n} \sum_{i=1}^n (y_i (\log \hat{p}_i(v_i) + \log \hat{p}_i(m_i))) \quad (12)$$

In the formula,  $L_{lp}$  represents the multi-class cross-entropy of the semantic classification model;  $n$  denotes the number of samples in each training batch;  $y_i$  represents the label information of the sample;  $\hat{p}_i(v_i)$  and  $\hat{p}_i(m_i)$  are the probabilities predicted by the model for the semantic categories  $v_i$  and  $m_i$  of the  $i$ -th sample, respectively.

Before discussing sample mining, it is essential to understand the concept of metric learning. Metric learning aims to optimize algorithms that rely on nearest-neighbor strategies by using an appropriate distance function. Deep metric learning, a specific form of this approach, is closely related to embedding network learning. In this method, entities are mapped into an embedding space, and the distance function is trained on all samples within the subspace to ensure similar entities cluster together while different entities are spread apart. The core of deep metric learning lies in the proper selection of a loss function, which often involves integrating specific sample information. However, constructing a large number of sample pairs during training not only increases the data processing burden but also includes many low-information, ineffective samples that contribute minimally to gradient updates. This necessitates an optimized processing strategy to avoid slow training speeds or getting stuck in local optima.

To improve training efficiency and model accuracy, this study introduces a hard sample mining technique before performing the mapping computation. This technique effectively accelerates algorithm convergence and enhances learning efficiency. Initially, through feature selection and classifier classification processing, the video features for each video-audio pair are selected as the anchor, the same class audio features are chosen as positives, and different class features are defined as negatives. In the constructed triplets, based on the distance relationship between the negative examples and the anchor, they can be classified into easy, semi-hard, and hard triplets. Given that easy triplets contain low information and are easy to identify, this study focuses on training with hard and semi-hard triplets. By optimizing the loss function, the distribution of samples in the encoding space is adjusted so that the distances between same-class samples are minimized, and distances between different-class samples are maximized. The sample mining operation involves inputting all training data into the neural network, obtaining the encoding for each sample, calculating

the distances between anchors and positives/negatives, and determining the difficulty category of the triplets based on these distances (see Figure 3). The calculation formula is as follows:

$$\cos(\mathbf{v}, \mathbf{m}) = \frac{\sum_{k=1}^N \mathbf{v}_k \mathbf{m}_k}{\sqrt{\sum_{k=1}^N \mathbf{v}_k^2} \sqrt{\sum_{i=1}^N \mathbf{m}_i^2}} \quad (13)$$

$$L = \max\{0, m + \cos(\psi_v, \psi_{m+}) - \cos(\psi_v, \psi_{m-})\} \quad (14)$$

In the formula,  $\mathbf{v}$  and  $\mathbf{m}$  represent the sample vector and the reference vector, respectively;  $\mathbf{v}_k$  and  $\mathbf{m}_k$  denote the  $k$ -th components of vectors  $\mathbf{v}$  and  $\mathbf{m}$ , respectively;  $N$  represents the dimension of the vector;  $L$  represents the loss value;  $m$  represents the set margin value;  $\cos(\psi_v, \psi_{m+})$  and  $\cos(\psi_v, \psi_{m-})$  represent the cosine similarities between the positive/negative samples and the reference sample, respectively.

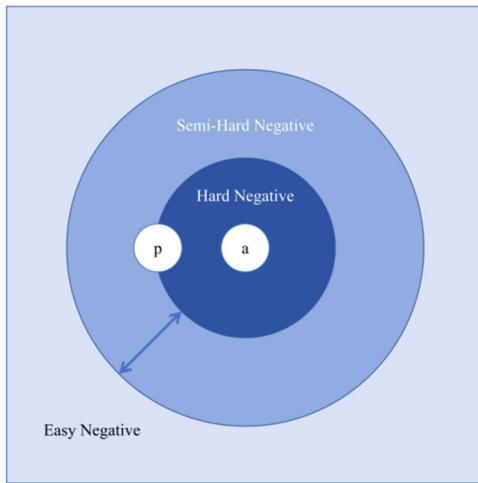


FIGURE 3. Positional relationships among triplets.

#### D. LOSS FUNCTION

In designing the loss function, two core elements were primarily considered: inter-modal similarity and intra-modal consistency. Inter-modal similarity provides theoretical support for cross-modal matching, while intra-modal consistency ensures that the features within the subspace maintain structural stability in the original feature space. Focusing solely on inter-modal relationships may result in the feature data losing the characteristics of their respective modalities during training. Therefore, this study elaborates on the construction method of the loss function from both inter-modal and intra-modal dimensions.

In this paper, considering the inherent defects of the tri-state loss function,  $\mathbf{X} = \{(x_i, y_i)\}_{i=1}^N$  is introduced to represent the training sample set, where  $(x_i, y_i)$  is the samples and the labeling information corresponding to them. All

samples consist of class  $c$ , expressed as  $y_i \in [i, 2, \dots, c]$ , and  $\{x_i^c\}_{i=1}^N$  denotes all samples. In order to realize the effective distinction between positive and negative samples, the distance between the baseline query and the negative examples is greater than the threshold value  $\alpha$ , and the distance between the baseline query and the positive examples does not exceed  $\alpha - m$  i.e., the spacing between the positive and negative samples is at least  $m$ . Therefore, it is necessary to design the corresponding function  $f$ , and through training to make the distribution of positive and negative samples in the subspace to meet the above distance relationship. Therefore the maximum loss function is defined as follows:

$$L_m = (1 - y_{ij})[\alpha - d_{ij}] + y_{ij}[\alpha - m] \quad (15)$$

The above equation, when  $y_i = y_j$ , can be obtained from  $y_{ij} = 1$ , otherwise  $y_{ij} = 0$ ;  $d_{ij}$  is the Euclidean distance between  $f(x_i)$ .

The loss function makes use of such a sampling strategy that positive samples are clustered within a sphere of radius size  $\alpha - m$ , and negative samples are spaced  $m$  apart from the positive samples, as shown in Figure 3 below. Given a benchmark vector  $x_i^c$ , and the other samples are ranked according to the similarity. In this sorting result, there are  $N_{c-1}$  positive samples and the number of negative samples is  $\sum_{k \neq c} N_k$ . Each benchmark is as close as possible to the set of positive samples and there is an interval of  $m$  between it and the set of negative samples, and it is also desirable that the distance between the ANCHOR and the negative samples is greater than the boundary  $\alpha$ , so for the positive samples the loss function is.

$$L_P(x_i^c; f) = \frac{1}{|P_{c,i}^*|} \sum_{x_j^c \in P_{c,j}^*} L_m \quad (16)$$

In the formula,  $P_{c,i}^*$  represents the positive sample set;  $|P_{c,i}^*|$  denotes the size of the positive sample set;  $L_m$  represents the loss value.

For the negative sample  $L_N(x_i^c; f)$ ;

$$L_N(x_i^c; f) = \sum_{x_j^k \in N_{c,i}^*} \frac{w_{ij}}{\sum_{x_j^k \in P_{c,j}^*} w_{ij}} L_m \quad (17)$$

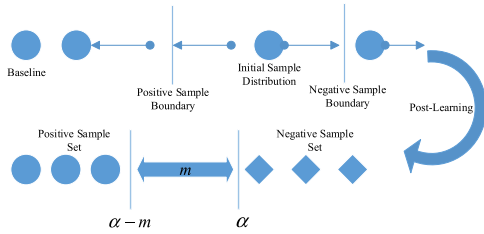
In the formula,  $N_{c,i}^*$  represents the negative sample set.

The overall loss function  $L_{\text{inter}}(x_i^c; f)$  is defined as:

$$L_{\text{inter}}(x_i^c; f) = L_P(x_i^c; f) + L_N(x_i^c; f) \quad (18)$$

To avoid losing these key characteristics during training and thereby affecting the accuracy of the experiments, this study introduces an intra-modal structure-preserving loss function. This loss function aims to maintain the structural integrity of the data within each modality during model training, ensuring that the distance between similar features remains small while the distance between different features is relatively large.





**FIGURE 4. Structural changes in the feature space after loss function training.**

In the audio modality, to preserve its intra-modal structure, the feature data should satisfy the following relationship:

$$d(\mathbf{m}_i, \mathbf{m}_j) < d(\mathbf{m}_i, \mathbf{m}_k) \quad (19)$$

$$\text{if } d(\tilde{\mathbf{m}}_i, \tilde{\mathbf{m}}_j) < d(\tilde{\mathbf{m}}_i, \tilde{\mathbf{m}}_k)$$

where  $\mathbf{m}_i, \mathbf{m}_j$  and  $\mathbf{m}_k$  represent the audio feature data in the subspace, respectively.  $\tilde{\mathbf{m}}_i, \tilde{\mathbf{m}}_j$  and  $\tilde{\mathbf{m}}_k$  represent the unmapped audio features. Considering that the matching relationship between audio and video needs to be more flexible, the maximum threshold distance function is not directly chosen to be applied in this study, but redefined on the basis of this, and its expression is:

$$L_{\text{intra}} = \lambda_1 \sum_{i \neq k \neq j} c_{ijk}(v) (v_i^T v_j - v_i^T v_k) + \lambda_2 \sum_{i \neq k \neq j} c_{ijk}(m) (m_i^T m_j - m_i^T m_k) \quad (20)$$

$$c_{ijk}(x) = \text{sign}(x_i^T x_j - x_i^T x_k) - \text{sign}(\tilde{x}_i^T \tilde{x}_j - \tilde{x}_i^T \tilde{x}_k) \quad (21)$$

To enhance the flexibility of the constraints, a sign function was introduced in the study. Here,  $x_i, x_j$  and  $x_k$  represent the feature data in the common subspace;  $\tilde{x}_i, \tilde{x}_j$  and  $\tilde{x}_k$  denote the feature data before mapping;  $\lambda_1$  and  $\lambda_2$  are the weight parameters in the loss function;  $c_{ijk}(v)$  and  $c_{ijk}(m)$  are the coefficients between the triplets  $i, j,$  and  $k$  based on  $v$  and  $m$ , respectively.

The application of the sign function is particularly crucial in multimodal learning, as it introduces nonlinearity to avoid the rigidity and singular matching results that may arise from loss functions dependent on Euclidean distance. By combining inter-modal and intra-modal loss functions, this study forms a comprehensive loss function framework for training sample data within the subspace. In this framework, feature data are first mapped to the target subspace through an embedding network, and then the sample data are integrated into triplet form for training. This method effectively integrates the characteristics of different modalities and optimizes interactions between data, thereby enhancing the overall performance and matching accuracy of the model. In this process, four forms of triplets,  $(v_i, m_i, m_j), (m_i, v_i, v_j), (v_i, v_j, v_k),$  and  $(m_i, m_j, m_k),$  are constructed. The overall multimodal training loss function is defined as:

$$L_{\text{multi-modal}} = \lambda_1 L_{\text{inter}} + \lambda_2 L_{\text{intra}} \quad (22)$$

### III. EXPERIMENTAL RESULTS

After designing and implementing the intelligent audio-video content retrieval model, we conducted extensive experiments to verify its effectiveness. This chapter provides a detailed introduction to the experimental setup, the selection and processing methods of the dataset, and the analysis and discussion of the experimental results. Through these experiments, we evaluated the model's performance under different conditions and compared its advantages and disadvantages with existing methods.

#### A. DATA SET COLLECTION AND PROCESSING

In this study, the CMU-MOSEI dataset was selected for cross-modal analysis and comparison of the original video and audio data. The dataset's video and audio resources include both Chinese and English parts, with English resources primarily sourced from YouTube and Chinese resources from platforms such as Youku, Bilibili, and Ximalaya. The video content encompasses various types, including interviews, speeches, daily conversations, and film clips. Video data is stored in MP4 format, maintaining consistent resolution and frame rate to ensure data uniformity and processability. Audio data is stored in WAV format with a sampling rate of 16kHz to ensure high-quality audio for analysis. A total of 6,859 paired audio-video files were obtained based on retrieval requirements.

To establish the pairing relationship between video and audio, we referred to the FLICKR30K image-text retrieval database model, where each video segment corresponds to five audio clips. Each audio clip is assigned a weight  $p$  based on its matching degree with the video  $p(0.45, 0.25, 0.15, 0.1, 0.05)$ . The annotation work was carried out by students and faculty with relevant professional knowledge, ensuring consistency and accuracy in the annotations.

Regarding data cleaning and preprocessing, both video and audio data underwent rigorous cleaning and preprocessing steps, which included removing noise and irrelevant data, and unifying the formats and resolutions of the video and audio files to ensure standardization in the processing. In terms of leveraging multimodal characteristics, the study utilized the multimodal features of video, audio, and text (through speech transcription) to enhance the model's performance in cross-modal retrieval tasks. The dataset contains sufficiently diverse scenarios and content, improving the model's generalization ability.

Due to the limited research on cross-modal retrieval and the lack of a unified evaluation method, this study employed evaluation metrics including Recall@K and Mean Average Precision (MAP). Recall@K refers to the proportion of test data where at least one standard video pair is included in the top K results for each video query. Different K values were used to comprehensively evaluate the model's performance. Mean Average Precision (MAP) measures the average performance of all test query results, ensuring comprehensive and accurate evaluation. During MAP calculation, all possible

matching results were considered to avoid data bias. The formula for calculating MAP is as follows:

$$\text{Map} = \frac{1}{N} \sum_{j=1}^k p_j^* \text{rel}(j) \tag{23}$$

Among them,  $p_j$  is the weight based on audio quality,  $N$  is the number of audio files similar to the query video, and  $\text{rel}(j)$  is a binary function indicating whether the audio is within the annotation range.

**B. DATASET TRAINING**

To evaluate the model’s performance, training and testing were conducted on the CMU-MOSEI dataset. The dataset was divided into 90% for training and 10% for testing. For cross-validation, a 5-fold cross-validation method was employed to assess the model’s performance. The dataset was split into five equal parts, with different parts used as the validation set in each experiment, while the remaining parts were used for training. This ensured that each part of the data was used for validation once, providing a robust assessment of the model’s performance. To prevent overfitting, an early stopping strategy was adopted. The current highest validation accuracy was recorded, and if there was no improvement in validation accuracy after an additional 10 epochs of training, the training was stopped. Based on relevant literature in the field [44], [45], [46] and the specific data requirements of this study, the optimal hyperparameters were determined as follows: a learning rate of 0.001, a batch size of 128, 40 iterations, 2 LSTM layers, 1024 neurons per layer, and a dropout rate of 0.2. During hyperparameter tuning, multiple experiments with different combinations of hyperparameters were conducted to determine the best parameter settings. Table 1

**TABLE 1. Hyperparameter tuning results.**

Hyperparameter	Test Value	Optimal Value
Learning Rate	[0.0001, 0.001, 0.01]	0.001
Batch Size	[32, 64, 128, 256]	128
Number of Iterations	[20, 40, 60]	40
Number of LSTM Layers	[1, 2, 3]	2
Number of Neurons per Layer	[512, 1024, 2048]	1024
Dropout Rate	[0.2, 0.4, 0.6]	0.2

presents the specific hyperparameter tuning results, where the optimal hyperparameter settings effectively enhanced the model’s performance.

In the ablation study, system components were removed to evaluate their contributions to model performance, as shown in Table 2. Removing the LSTM network resulted in an 8% decrease in recall and a 7% decrease in average precision. Eliminating the attention mechanism led to a 5% decrease in recall and a 6% decrease in average precision. Removing the SKM algorithm caused a 10% drop in fidelity and an 8% reduction in compression ratio. These results indicate that each component plays a crucial role in enhancing model performance.

**TABLE 2. Ablation study results.**

Component Removed	Recall Decrease Ratio	Average Precision Decrease Ratio	Fidelity Decrease Ratio	Compression Ratio Decrease Ratio
LSTM Network	8%	7%	/	/
Attention Mechanism	5%	6%	/	/
SKM Algorithm	/	/	10%	8%

The experiments were conducted on a system equipped with an NVIDIA GTX 2070 GPU, using the average results from five runs as the evaluation standard for model performance. During the data preprocessing stage, audio and video data were divided into equal-length segments. Input features were selected and then entered into a common subspace through the combined efforts of a classifier, sample mining, and a mapping network. Further training and adjustments were carried out using a loss function. Finally, the model’s performance was evaluated using the test set. The system generated an audio list based on the input video as retrieval results, calculating the average recall and average precision accordingly.

In the experimental setup, for processing video data, the network adopted a structure with two fully connected layers, with the number of nodes set to 2048 and 512, respectively. For audio data processing, a fully connected network was

constructed with nodes set to 2048, 1024, and 512 sequentially. The activation functions chosen were sigmoid and tanh, aiming to enhance the model's nonlinear processing capabilities through different functional characteristics. To address the imbalance in the magnitude of audio and video data features and prevent model overfitting, L2 normalization was introduced in the experiment. Regarding the tuning of the training process, the optimal batch size was determined to be 550, with a total training period of 40 epochs, achieving the best balance between training efficiency and result accuracy. Additionally, the model used the ADAM optimizer for parameter optimization, with the learning rate set to 0.001 and a dropout probability of 0.2 to further enhance the model's generalization ability.

### C. DATASET TRAINING

#### 1) GENERATION OF ADVERSARIAL EXAMPLES

To validate the robustness of the model under adversarial examples, additional adversarial experiments were designed and conducted. This study utilized FGSM (Fast Gradient Sign Method) and PGD (Projected Gradient Descent) adversarial attack methods to generate adversarial samples. These methods generate adversarial examples by adding subtle perturbations to the input data to test the model's robustness. Multiple sets of adversarial samples were generated for different types of input data to ensure the comprehensiveness and representativeness of the experiments.

#### 2) EXPERIMENTAL SETUP AND METHODS

To comprehensively evaluate the model's robustness under different attack intensities, three levels of perturbation strength were set: 0.01, 0.1, and 0.3, representing mild, moderate, and severe adversarial attacks, respectively. The same dataset used for model training was selected for testing to ensure consistency in the generation and testing environment of the adversarial samples. The dataset was preprocessed to ensure uniform format and features for each sample.

The specific experimental steps were as follows:

- (1) For each perturbation strength, adversarial samples were generated using FGSM and PGD methods.
- (2) The generated adversarial samples were input into the model, and the model's output and accuracy on these samples were recorded.
- (3) The performance difference of the model on normal samples and adversarial samples was compared to analyze the model's robustness under different perturbation strengths.
- (4) Accuracy was used as the evaluation metric to quantify the model's robustness. The accuracy changes of the model on normal samples and adversarial samples were compared to assess the model's performance under different attack intensities.

### D. RESULTS ANALYSIS

#### 1) COMPARATIVE ANALYSIS OF KEY FRAME EXTRACTION ALGORITHMS

To validate the effectiveness of the proposed SKM-based key frame extraction algorithm, this study compares it with

three traditional key frame extraction methods: the shot boundary-based extraction method (Method 1), the inter-frame difference-based extraction method (Method 2), and a clustering-based algorithm (Method 3). The advantages and disadvantages of these four methods in practical applications are summarized in Table 3.

The results obtained from the comparative analysis are shown in Table 4.

From the data analysis in Table 4, Method 1 demonstrates the most outstanding performance in terms of compression ratio, achieving an average compression ratio of 96.6%, the highest among all methods. However, its fidelity is the lowest, with an average of only 53%. In stark contrast, Method 2 excels in fidelity, reaching 70.9%, the highest of all methods, but its compression ratio is relatively low at 87.4%. Method 3 shows a balanced performance in both aspects, with an average fidelity of 64.5% and an average compression ratio of 93.1%. Meanwhile, the algorithm proposed in this paper also exhibits good balance, achieving a fidelity of 69.3% and an average compression ratio of 93.6%.

The comparison curves of key frame extraction results are shown in Figures 5 and 6.

According to Figure 5, it can be observed that the curve for Method 2 is at the highest position, indicating its optimal fidelity. Following in order are the proposed algorithm, Method 3, and Method 1, with Method 1 having the lowest fidelity. The specific data show that the average fidelity

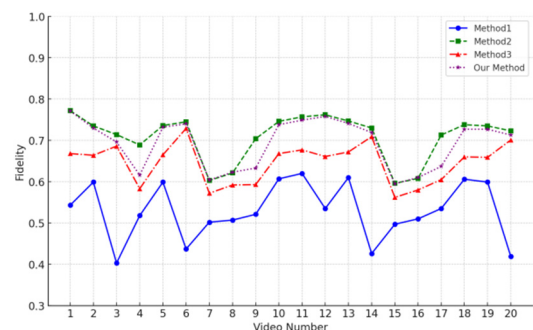


FIGURE 5. Comparison of video fidelity among different types of algorithms.

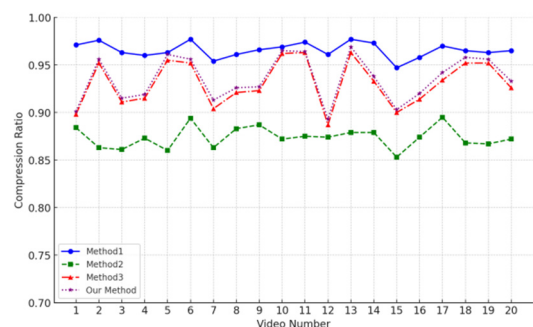


FIGURE 6. Comparison of video compression.

**TABLE 3. Comparison of advantages and disadvantages of different key frame extraction methods.**

Method	Advantages	Disadvantages
SKM-based Extraction Method	High accuracy in key frame extraction; Efficient in processing dynamic content videos; Capable of handling various types of videos effectively	High computational complexity; Relatively complex parameter tuning
Shot Boundary-based Extraction Method (Method 1)	Simple implementation; Effective for videos with distinct shot boundaries	May miss important frames within shots; Ineffective for videos without clear shot boundaries
Inter-frame Difference-based Extraction Method (Method 2)	Good at capturing significant changes between frames; Useful for videos with gradual transitions	Sensitive to noise and minor changes; May lead to redundant frames if changes are frequent
Clustering Algorithm-based Method (Method 3)	Effectively groups similar frames; Reduces redundancy by selecting representative frames from clusters	High computational cost; Requires predefined number of clusters

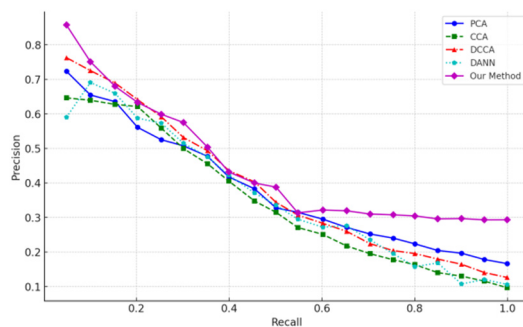
of the algorithm proposed in this study is 0.693, which is 2.26% lower than that of Method 2, 7.45% higher than that of Method 3, and 30.75% higher than that of the shot boundary-based Method 1.

From Figure 6, it can be seen that Method 1 ranks highest in terms of compression ratio, followed closely by the proposed algorithm, then Method 3, and Method 2 with the lowest compression ratio. Detailed data analysis reveals that the average compression ratio of the algorithm in this study is 0.936, which is 3.11% lower than Method 1, 0.54% higher than Method 3, and 7.09% higher than Method 2. These results confirm that the SKM-based key frame extraction algorithm has a significant advantage in overall performance.

### 2) COMPARATIVE ANALYSIS OF MULTIPLE MODELS

This study compares the proposed model with existing audio-visual cross-modal retrieval models and conducts experiments on the CMU-MOSEI dataset. These models can be categorized into linear and nonlinear types based on their processing mechanisms. Linear models, such as PCA and CCA, mainly explore linear correlations between modalities. Nonlinear models, such as DCCA and DANN, are suitable for different application environments. Although DCCA is typically used for image-text retrieval, it has been applied to audio-visual cross-modal retrieval in this study, demonstrating outstanding performance.

Figure 7 shows the experimental results, indicating that traditional linear models perform poorly in cross-modal retrieval, whereas nonlinear models using deep neural networks exhibit higher performance. Our proposed model utilizes a similarity loss function, which not only enhances effectiveness but also facilitates convergence during training. The PR curves of the five models on the dataset, displayed in Figure 7, clearly reveal the performance differences among the models. Notably, the new model demonstrates exceptional performance on the CMU-MOSEI dataset.



**FIGURE 7. PR Curves of different models on the CMU-MOSEI dataset.**

To further investigate the computational efficiency of the models, this study compares the runtime and resource utilization of different models. The runtime and resource

**TABLE 4.** Comparison of different types of key frame extraction algorithms.

	Method1		Method2		Method3		Our Method	
	Fidelit y	Compressi on Ratio	Fidelit y	Compressi on Ratio	Fidelit y	Compressi on Ratio	Fidelit y	Compressi on Ratio
1	0.543	0.971	0.772	0.884	0.668	0.898	0.771	0.901
2	0.599	0.976	0.735	0.863	0.664	0.952	0.730	0.956
3	0.403	0.963	0.714	0.861	0.686	0.911	0.696	0.915
4	0.518	0.960	0.689	0.873	0.583	0.915	0.616	0.919
5	0.599	0.963	0.736	0.860	0.665	0.955	0.732	0.961
6	0.437	0.977	0.745	0.894	0.729	0.952	0.740	0.956
7	0.502	0.954	0.603	0.863	0.572	0.904	0.604	0.913
8	0.507	0.961	0.621	0.883	0.592	0.921	0.623	0.926
9	0.521	0.966	0.704	0.887	0.593	0.923	0.633	0.927
10	0.607	0.969	0.746	0.872	0.668	0.962	0.738	0.965
11	0.620	0.974	0.757	0.875	0.677	0.963	0.749	0.964
12	0.535	0.961	0.762	0.874	0.661	0.887	0.758	0.893
13	0.610	0.977	0.747	0.879	0.672	0.963	0.741	0.969
14	0.426	0.973	0.730	0.879	0.709	0.933	0.719	0.938
15	0.497	0.947	0.596	0.853	0.562	0.900	0.595	0.903
16	0.510	0.958	0.608	0.874	0.580	0.914	0.610	0.920
17	0.535	0.970	0.713	0.895	0.605	0.934	0.637	0.942
18	0.606	0.965	0.738	0.868	0.660	0.952	0.727	0.958
19	0.599	0.963	0.735	0.867	0.659	0.952	0.727	0.956
20	0.419	0.965	0.723	0.872	0.701	0.926	0.713	0.933
Average	0.530	0.966	0.709	0.874	0.645	0.931	0.693	0.936

consumption of various models on the CMU-MOSEI dataset are shown in Table 5.

From the results in Table 5 and Figure 8, it can be observed that both linear models PCA and CCA, as well as nonlinear models DCCA and DANN, have runtimes exceeding 100 seconds on the dataset. This is primarily due to the complex computational processes these methods rely on when handling multimodal data. In contrast, the proposed method in this study has a runtime of only 90 seconds, significantly reducing the processing time.

As shown in Figure 9, compared to the CPU and GPU utilization rates of both linear and nonlinear models, the proposed model demonstrates overall lower utilization levels. Specifically, the CPU utilization is 70% and the GPU utilization is 50%, indicating more optimized resource usage.

The results in Figure 10 show that in terms of resource consumption, the proposed research model consumes only 7GB of memory. This indicates that while maintaining efficient retrieval, the proposed method significantly reduces resource consumption. This improvement not only enhances the real-time performance and accuracy of retrieval but also provides a more efficient solution for large-scale data processing.

#### E. ROBUSTNESS TEST RESULTS ANALYSIS

The robustness test results of the model are shown in Figures 11 and 12.

From the results in Figure 11, it can be seen that under FGSM attack, the model demonstrates good robustness with

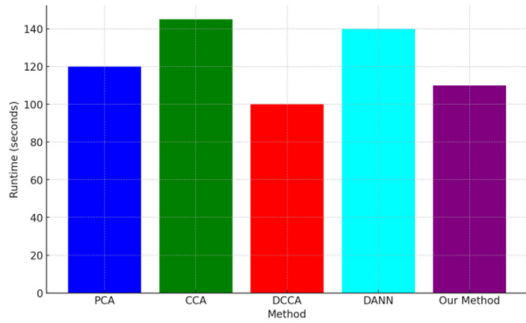


FIGURE 8. Comparison of runtime among different models.

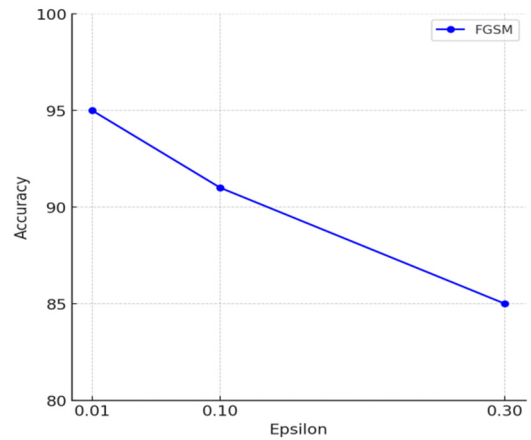


FIGURE 11. Robustness test results of the model under FGSM attack.

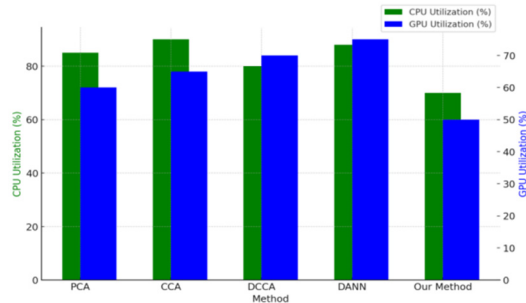


FIGURE 9. CPU and GPU utilization of different models.

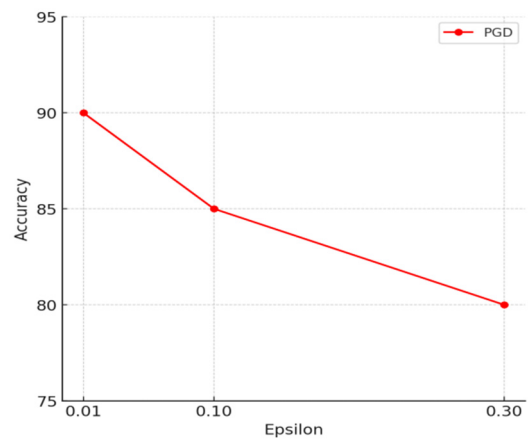


FIGURE 12. Robustness test results of the model under PGD attack.

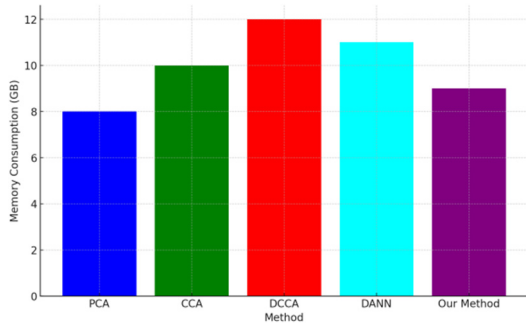


FIGURE 10. Comparison of resource consumption among different models.

TABLE 5. Comparison of runtime and resource consumption of different models on the CMU-MOSEI dataset.

Meth od	Runti me (s)	CPU Utilizat ion (%)	GPU Utilizat ion (%)	Memory consumpt ion (GB)
PCA	120	85	60	8
CCA	150	90	65	10
DCC A	100	80	70	12
DAN N	130	88	75	11
Our Meth od	90	70	50	7

an accuracy of 95% when the perturbation intensity is 0.01. As the perturbation intensity increases to 0.1, the model’s accuracy drops to 91%. With a further increase in perturbation intensity to 0.3, the model’s accuracy decreases to 85%. These results indicate that while the model maintains high accuracy under low-intensity attacks, its robustness gradually weakens with increasing perturbation intensity.

For PGD attack (Figure 12), the model’s accuracy is 90% at a perturbation intensity of 0.01. When the perturbation intensity increases to 0.1, the accuracy drops to 85%, and at an intensity of 0.3, the accuracy falls to 80%. Compared to FGSM attacks, the model performs worse under PGD attacks, particularly at higher perturbation intensities, where the accuracy significantly decreases. This indicates that PGD

attacks have a more pronounced impact on the model, making it more vulnerable to such attacks.

Overall, the model can maintain high accuracy under low-intensity perturbations, and although its performance declines under high-intensity perturbations, the accuracy remains above 80%, demonstrating good robustness. Specifically, under FGSM attacks, the model shows high accuracy across different perturbation intensities, indicating strong resistance to this type of attack. Under PGD attacks, despite the more significant drop in accuracy, the model still maintains a high level of accuracy at 80%, showing that the model retains a certain degree of robustness against more complex attacks.

#### IV. DISCUSSION

In the context of the massive growth of multimedia data, intelligent retrieval technology for audio and video content has gained increasing attention. This study proposes an intelligent retrieval method for audio and video content based on deep learning. By comprehensively applying audio and video feature extraction techniques, the method utilizes the VGG network to extract audio features and employs the self-adaptive clustering key frame extraction algorithm (SKM) to extract video features. Combined with a cross-learning framework of embedding networks, this approach significantly enhances retrieval efficiency and accuracy. The test results on the CMU-MOSEI dataset indicate that this method outperforms traditional PCA and CCA models as well as the latest DCCA and DANN deep learning models in multimodal data processing and actual retrieval tasks. The proposed method achieved an average fidelity of 0.693 and an average compression ratio of 0.936, which represent improvements of 30.75% and 7.09% over traditional methods, respectively.

Traditional retrieval techniques often rely on metadata and simple visual or audio features, such as basic attributes of color, texture, or sound. Existing studies, including the method combining deep triplet neural networks and clustered canonical correlation analysis (CCA) proposed by Zeng et al. [47] and the multi-resolution audio-video feature fusion (MRAV-FF) method by Fish et al. [48], have made progress in specific functionalities. However, their performance in practical applications, especially in cross-modal audio-video data processing, has been suboptimal. This study leverages long short-term memory (LSTM) networks and attention mechanisms to process audio data, and employs adaptive clustering algorithms and deep feature extraction techniques for video data. This comprehensive approach not only optimizes the processing of individual modalities but also enhances cross-modal data processing capabilities through a cross-learning framework. Experimental results on the CMU-MOSEI dataset indicate that, compared to traditional methods like PCA, CCA, and deep models such as DCCA and DANN, the proposed model demonstrates superior performance in multimodal data processing and retrieval tasks, with improved recall and mean average precision (MAP).

In current similar research, the Transformer-based feature fusion network for cross-modal retrieval proposed by

Zhang & Cao (2023) holds certain value. However, its extraction and processing of audio features are relatively simplistic, leading to deficiencies in audio-video matching accuracy [49]. In contrast, the method proposed in this paper achieves greater accuracy in matching by performing fine-grained analysis and processing of audio and video features. On another front, the large-scale video retrieval method based on convolutional neural networks (CNN) proposed by Zhang et al., which implements keyframe extraction and feature aggregation strategies to achieve low storage costs and high search efficiency, shows practical value but faces computational efficiency challenges when handling large-scale data [50]. Addressing such issues, this study combines adaptive clustering algorithms with deep feature extraction techniques to significantly improve the efficiency of large-scale data processing while also enhancing the accuracy of feature extraction.

In terms of video feature extraction, the proposed adaptive clustering keyframe extraction algorithm (SKM) addresses the issue of fixed cluster numbers potentially leading to inaccurate keyframe quantities in traditional methods by automatically calculating the number of clusters and initial cluster centers. Comparative analysis shows that the proposed method outperforms several traditional methods in both fidelity and compression ratio, demonstrating higher overall performance. Compared to the dynamic mode decomposition-based feature extraction method proposed by Sikha and Soman, the extraction method in this study exhibits greater robustness and efficiency in handling complex video data [28].

This study theoretically introduces new approaches for intelligent retrieval of audio-visual content, achieving efficient multimodal data processing through a cross-learning framework. This method not only optimizes single-modality processing but also significantly enhances retrieval performance by enabling comparative analysis of cross-modal data through a feature embedding network. The study improves the efficiency and accuracy of audio-visual content retrieval and provides new theoretical and technical references for the development of future cross-modal retrieval technologies. By applying deep learning techniques, this study demonstrates strong potential and broad applicability in handling large-scale multimedia data, offering valuable insights and experiences for researchers and developers in related fields. Experimental results on the CMU-MOSEI dataset confirm the feasibility and effectiveness of the proposed method in practical applications, offering a new perspective for intelligent multimodal data retrieval.

Despite the significant progress made, there are several limitations to this study. Firstly, the current method incurs high computational resources and time costs when processing large-scale and diverse datasets, which may significantly impact its application in resource-constrained real-world scenarios. Future research should focus on optimizing algorithms to reduce computational complexity, exploring parallel computing and distributed processing techniques, and

utilizing hardware acceleration technologies such as GPUs and FPGAs to enhance computational efficiency. Secondly, there is a need to further improve real-time retrieval capabilities. Future studies should develop more efficient real-time processing algorithms, explore more effective data structures such as inverted indexes or hash tables to accelerate the retrieval process, and design synchronous processing frameworks for multimodal data to reduce latency. Lastly, the model's preprocessing requirements for audio-visual data are relatively high, which may increase the complexity and preparation time of practical applications. Although the model performs well in experiments, its robustness and generalizability need further validation and improvement when handling more diverse and noisy data. Future research could enhance the model's robustness by incorporating more advanced preprocessing techniques, explore generalizability across more varied datasets, and develop more efficient real-time and parallel processing algorithms to improve computational efficiency and processing speed, thereby better meeting practical application needs.

## V. CONCLUSION

This paper presents an intelligent retrieval method for audio-visual content based on deep learning, aiming to enhance the efficiency and accuracy of such content retrieval. The method employs long short-term memory (LSTM) networks and attention mechanisms for fine-grained feature extraction of audio data. Compared to traditional methods, the combination of LSTM and attention mechanisms significantly improves the precision of audio feature extraction. Additionally, an adaptive clustering keyframe extraction algorithm (SKM) is proposed, which addresses the issue of inaccurate keyframe numbers in traditional methods by automatically calculating the number of clusters and initial cluster centers. In experiments, this method demonstrates superior performance in terms of fidelity and compression ratio, with average fidelity and compression ratio values of 0.693 and 0.936, respectively, representing improvements of 30.75% and 7.09% over traditional methods. In multimodal data processing, audio and video features are mapped to a shared feature space through an embedding network, effectively enhancing cross-modal data processing capabilities. Experimental results on the CMU-MOSEI dataset indicate that, compared to traditional PCA and CCA models as well as recent DCCA and DANN models, the proposed method achieves significant improvements in recall and mean average precision (MAP).

This method holds substantial practical significance for organizations requiring efficient management and retrieval of large volumes of audio-visual data, such as multimedia content providers, video surveillance systems, educational and training platforms, research institutions, and enterprises. It not only significantly reduces the search time for audio-visual content and enhances user experience but also enables more comprehensive retrieval and analysis by handling cross-modal audio and video data, making it

applicable in various real-world scenarios. Moreover, the method demonstrates robustness and efficiency in processing complex video data, indicating strong potential for practical applications.

## DECLARATIONS

### DATA AVAILABILITY

All data analyzed in this study are available on request from the author.

## DECLARATIONS

Ethical approval not required.

## COMPETING INTERESTS

The author declare no competing interests.

## REFERENCES

- [1] S. S. Tyokighir, J. Mom, K. E. Ukhurebor, and G. Igwe, "New developments and trends in 5G technologies: Applications and concepts," *Bull. Electr. Eng. Informat.*, vol. 13, no. 1, pp. 254–263, Feb. 2024, doi: [10.11591/eei.v13i1.6032](https://doi.org/10.11591/eei.v13i1.6032).
- [2] X. Xia, C. Yu, F. Wu, Z. Jiang, S. Zheng, S.-Y. Tang, Y. Yao, and W. Hong, "Millimeter-wave beam-tilted phased array antenna for 5G-enabled IoT devices," *IEEE Internet Things J.*, vol. 11, no. 1, pp. 1496–1508, Jan. 2024, doi: [10.1109/JIOT.2023.3290563](https://doi.org/10.1109/JIOT.2023.3290563).
- [3] S.-P. Yeh, S. Bhattacharya, R. Sharma, and H. Moustafa, "Deep learning for intelligent and automated network slicing in 5G open RAN (ORAN) deployment," *IEEE Open J. Commun. Soc.*, vol. 5, pp. 64–70, 2024, doi: [10.1109/OJCOMS.2023.3337854](https://doi.org/10.1109/OJCOMS.2023.3337854).
- [4] X. Zhou, L. Chen, and Y. Ruan, "Indoor positioning with multibeam CSI from a single 5G base station," *IEEE Sensors Lett.*, vol. 8, no. 1, pp. 1–4, Jan. 2024, doi: [10.1109/lse.2023.3338252](https://doi.org/10.1109/lse.2023.3338252).
- [5] M. P. Nayak, S. R. N. Thirtha, and A. S. Vaishnavi, "Review paper on 5G network," *Int. J. Adv. Res. Sci., Commun. Technol.*, vol. 2, no. 1, pp. 592–597, Aug. 2022.
- [6] S. Sharma, "Social media—An opportunity for organisations," *Int. J. Sci., Eng. Manage.*, vol. 9, no. 10, pp. 96–101, Oct. 2022.
- [7] Z. Lu and I. Nam, "Research on the influence of new media technology on internet short video content production under artificial intelligence background," *Complexity*, vol. 2021, no. 1, Jan. 2021, Art. no. 8875700.
- [8] J. Farchy, G. Bideau, and S. Tallec, "Content quotas and prominence on VOD services: New challenges for European audiovisual regulators," *Int. J. Cultural Policy*, vol. 28, no. 4, pp. 419–430, Jun. 2022.
- [9] E. M. Saoudi and S. Jai-Andaloussi, "A distributed content-based video retrieval system for large datasets," *J. Big Data*, vol. 8, no. 1, pp. 1–26, Jun. 2021.
- [10] A. Nagrani, P. H. Seo, B. Seybold, A. Hauth, S. Manen, C. Sun, and C. Schmid, "Learning audio-video modalities from image captions," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 407–426.
- [11] J. L. Alcazar, F. Caba, L. Mai, F. Perazzi, J. Y. Lee, P. Arbelaez, and B. Ghanem, "APES: Audiovisual person search in untrimmed video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, 2021, pp. 1720–1729.
- [12] W. Zhou, H. Li, and Q. Tian, "Recent advance in content-based image retrieval: A literature survey," 2017, *arXiv:1706.06064*.
- [13] H. Qazanfari, M. M. AlyanNezhadi, and Z. N. Khoshdaregi, "Advancements in content-based image retrieval: A comprehensive survey of relevance feedback techniques," 2023, *arXiv:2312.10089*.
- [14] D. Petkovic, W. Niblack, M. Flickner, D. Steele, D. Lee, J. Yin, J. Hafner, F. Tung, H. Treat, R. Dow, and M. Gee, "Recent applications of IBM's query by image content (QBIC)," in *Proc. ACM Symp. Appl. Comput.*, Feb. 1996, pp. 2–6.
- [15] H. M. Haav and T. L. Lubi, "A survey of concept-based information retrieval tools on the web," in *Proc. 5th East-Eur. Conf. ADBIS*, vol. 2, Sep. 2001, Sep. 2001, pp. 29–41.
- [16] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. C. Jain, and C. F. Shu, "Virage image search engine: An open framework for image management," *Proc. SPIE*, vol. 2670, pp. 76–87, Mar. 1996.



- [17] J. R. Smith and S. F. Chang, "Querying by color regions using the VisualSEEK content-based visual query system," *Intell. Multimedia Inf. Retr.*, vol. 7, no. 3, pp. 23–41, 1997.
- [18] Z. W. Tu, K. M. Wang, Y. Dong, and Y. Zhao, "Research and implementation of the retrieval system based on texture and shape," *Appl. Mech. Mater.*, vol. 419, pp. 581–586, Oct. 2013.
- [19] Z. Chen, L. Wenyin, C. Hu, M. Li, and H. J. Zhang, "iFind: A web image search engine," in *Proc. 24th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Sep. 2001, p. 450.
- [20] I. Manco, E. Benetos, E. Quinton, and G. Fazekas, "MusCaps: Generating captions for music audio," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8, doi: [10.1109/IJCNN52387.2021.9533461](https://doi.org/10.1109/IJCNN52387.2021.9533461).
- [21] B. Peng, J. Lei, H. Fu, Y. Jia, Z. Zhang, and Y. Li, "Deep video action clustering via spatio-temporal feature learning," *Neurocomputing*, vol. 456, pp. 519–527, Oct. 2021, doi: [10.1016/j.neucom.2020.05.123](https://doi.org/10.1016/j.neucom.2020.05.123).
- [22] Y. Xu, E. Wang, Y. Yang, and Y. Chang, "A unified collaborative representation learning for neural-network based recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 11, pp. 5126–5139, Nov. 2022, doi: [10.1109/TKDE.2021.3054782](https://doi.org/10.1109/TKDE.2021.3054782).
- [23] L. Vujošević and S. Dukanović, "Deep learning-based classification of environmental sounds," in *Proc. 25th Int. Conf. Inf. Technol. (IT)*, Feb. 2021, pp. 1–4, doi: [10.1109/IT51528.2021.9390124](https://doi.org/10.1109/IT51528.2021.9390124).
- [24] L. Carvalho and G. Widmer, "Passage summarization with recurrent models for audio-sheet music retrieval," 2023, *arXiv:2309.12111*.
- [25] Q. Hao, F. Wang, X. Ma, and P. Zhang, "A speech recognition algorithm of speaker-independent Chinese isolated words based on RNN-LSTM and attention mechanism," in *Proc. 14th Int. Congr. Image Signal Process., Biomed. Eng. Informat. (CISP-BMEI)*, Oct. 2021, pp. 1–4, doi: [10.1109/CISP-BMEI53629.2021.9624368](https://doi.org/10.1109/CISP-BMEI53629.2021.9624368).
- [26] H. Xie, K. Khorrani, O. Räsänen, and T. Virtanen, "Crowdsourcing and evaluating text-based audio retrieval relevances," 2023, *arXiv:2306.09820*.
- [27] Y. Deng and B. S. Manjunath, "Content-based search of video using color, texture, and motion," in *Proc. Int. Conf. Image Process.*, vol. 2, 1997, pp. 534–537, doi: [10.1109/icip.1997.638826](https://doi.org/10.1109/icip.1997.638826).
- [28] O. K. Sikha and K. P. Soman, "Dynamic mode decomposition based salient edge/region features for content based image retrieval," *Multimedia Tools Appl.*, vol. 80, no. 10, pp. 15937–15958, Apr. 2021, doi: [10.1007/s11042-020-10315-8](https://doi.org/10.1007/s11042-020-10315-8).
- [29] Q. Liu, H. Yuan, R. Hamzaoui, H. Su, J. Hou, and H. Yang, "Reduced reference perceptual quality model with application to rate control for video-based point cloud compression," *IEEE Trans. Image Process.*, vol. 30, pp. 6623–6636, 2021, doi: [10.1109/TIP.2021.3096060](https://doi.org/10.1109/TIP.2021.3096060).
- [30] D. Zhong and S.-F. Chang, "Spatio-temporal video search using the object based video representation," in *Proc. Int. Conf. Image Process.*, vol. 1, 1997, pp. 21–24, doi: [10.1109/icip.1997.647374](https://doi.org/10.1109/icip.1997.647374).
- [31] X. Hu and Q. Li, "Dynamic visual SLAM system based on weighted static features," *Proc. SPIE*, vol. 12754, pp. 36–44, Aug. 2023, doi: [10.1117/12.2684168](https://doi.org/10.1117/12.2684168).
- [32] W. Zheng, L. Lin, X. Wu, and X. Chen, "An empirical study on correlations between deep neural network fairness and neuron coverage criteria," *IEEE Trans. Softw. Eng.*, vol. 50, no. 3, pp. 391–412, Mar. 2024, doi: [10.1109/tse.2023.3349001](https://doi.org/10.1109/tse.2023.3349001).
- [33] Y. Lu, C. Cao, Y. Zhang, and Y. Zhang, "Learnable locality-sensitive hashing for video anomaly detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 2, pp. 963–976, Feb. 2023, doi: [10.1109/TCSVT.2022.3205348](https://doi.org/10.1109/TCSVT.2022.3205348).
- [34] Z. Wu, H. Zhu, L. He, Q. Zhao, J. Shi, and W. Wu, "Real-time stereo matching with high accuracy via spatial attention-guided upsampling," *Int. J. Speech Technol.*, vol. 53, no. 20, pp. 24253–24274, Oct. 2023, doi: [10.1007/s10489-023-04646-w](https://doi.org/10.1007/s10489-023-04646-w).
- [35] S. Jin, H. Yao, Q. Zhou, Y. Liu, J. Huang, and X. Hua, "Unsupervised discrete hashing with affinity similarity," *IEEE Trans. Image Process.*, vol. 30, pp. 6130–6141, 2021, doi: [10.1109/TIP.2021.3091895](https://doi.org/10.1109/TIP.2021.3091895).
- [36] M. Awan and J. Shin, "Semantic video segmentation with dynamic keyframe selection and distortion-aware feature rectification," *Image Vis. Comput.*, vol. 110, Jun. 2021, Art. no. 104184, doi: [10.1016/j.imavis.2021.104184](https://doi.org/10.1016/j.imavis.2021.104184).
- [37] K. J. Naik and A. Soni, "Video classification using 3D convolutional neural network," in *Advancements in Security and Privacy Initiatives for Multimedia Images*. Hershey, PA, USA: IGI Global, 2021, pp. 1–18.
- [38] R. S. Kiziltepe, J. Q. Gan, and J. J. Escobar, "A novel keyframe extraction method for video classification using deep neural networks," *Neural Comput. Appl.*, vol. 35, no. 34, pp. 24513–24524, Dec. 2023, doi: [10.1007/s00521-021-06322-x](https://doi.org/10.1007/s00521-021-06322-x).
- [39] J. Chen, M. Xu, W. Xu, D. Li, W. Peng, and H. Xu, "A flow feedback prediction based on visual quantified features," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 10067–10075, Sep. 2023, doi: [10.1109/TITS.2023.3269794](https://doi.org/10.1109/TITS.2023.3269794).
- [40] J. Chen, Q. Wang, W. Peng, H. Xu, X. Li, and W. Xu, "Disparity-based multiscale fusion network for transportation detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 18855–18863, Oct. 2022, doi: [10.1109/TITS.2022.3161977](https://doi.org/10.1109/TITS.2022.3161977).
- [41] S. Pan, G. J. W. Xu, K. Guo, S. H. Park, and H. Ding, "Video-based engagement estimation of game streamers: An interpretable multimodal neural network approach," *IEEE Trans. Games*, early access, Dec. 29, 2023, doi: [10.1109/tg.2023.3348230](https://doi.org/10.1109/tg.2023.3348230).
- [42] N. Giatrakos, E. Kougioumtzi, A. Kontaxakis, A. Deligiannakis, and Y. Kotidis, "EasyFlinkCEP: Big event data analytics for everyone," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 474–481, doi: [10.1145/3459637.3482094](https://doi.org/10.1145/3459637.3482094).
- [43] H. Lu, M. Zhang, X. Xu, Y. Li, and H. T. Shen, "Deep fuzzy hashing network for efficient image retrieval," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 1, pp. 166–176, Jan. 2021.
- [44] Y. Huang, C.-G. Gu, and H.-J. Yang, "Junk-neuron-deletion strategy for hyperparameter optimization of neural networks," *Acta Phys. Sinica*, vol. 71, no. 16, 2022, Art. no. 160501, doi: [10.7498/aps.71.20220436](https://doi.org/10.7498/aps.71.20220436).
- [45] Y. Mao, A. Pranolo, A. P. Wibawa, A. B. Putra Utama, F. A. Dwiyanto, and S. Saifullah, "Selection of precise long short term memory (LSTM) hyperparameters based on particle swarm optimization," in *Proc. Int. Conf. Appl. Artif. Intell. Comput. (ICAAIC)*, May 2022, pp. 1114–1121, doi: [10.1109/ICAAIC53929.2022.9792708](https://doi.org/10.1109/ICAAIC53929.2022.9792708).
- [46] I. S. Kervanci and F. Akay, "LSTM hyperparameters optimization with hparam parameters for Bitcoin price prediction," *Sakarya Univ. J. Comput. Inf. Sci.*, vol. 6, no. 1, pp. 1–9, Apr. 2023, doi: [10.35377/saucis...1172027](https://doi.org/10.35377/saucis...1172027).
- [47] D. Zeng, Y. Yu, and K. Oyama, "Deep triplet neural networks with cluster-CCA for audio-visual cross-modal retrieval," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 16, no. 3, pp. 1–23, Aug. 2020, doi: [10.1145/3387164](https://doi.org/10.1145/3387164).
- [48] E. Fish, J. Weinbren, and A. Gilbert, "Multi-resolution audio-visual feature fusion for temporal action localization," 2023, *arXiv:2310.03456*.
- [49] G. Zhang and J. Cao, "Feature fusion based on transformer for cross-modal retrieval," *J. Phys., Conf.*, vol. 2558, no. 1, Aug. 2023, Art. no. 012012, doi: [10.1088/1742-6596/2558/1/012012](https://doi.org/10.1088/1742-6596/2558/1/012012).
- [50] C. Zhang, B. Hu, Y. Suo, Z. Zou, and Y. Ji, "Large-scale video retrieval via deep local convolutional features," *Adv. Multimedia*, vol. 2020, pp. 1–8, Jun. 2020, doi: [10.1155/2020/7862894](https://doi.org/10.1155/2020/7862894).



**MAOJIN SUN** was born in Shandong, China, in 1981. He received the B.S. degree in computer science and technology, the M.S. degree in computer science, and the Ph.D. degree in logistics engineering and management from Dalian Maritime University, Liaoning, China, in 2004, 2010, and 2019, respectively.

From 2004 to 2007, he was the Project Manager of Panasonic Software Development Center Dalian Company Ltd. Since 2013, he has been the Chairman of CEICloud Data Storage Technology (Beijing) Company Ltd., Beijing, China. His research interests include big data processing, artificial intelligence, video technological development, and edge computing.

• • •