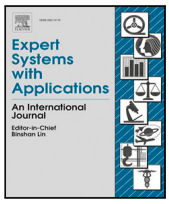
Expert Systems With Applications 276 (2025) 127128

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



Multi-Temporal Granularity Concept Induction for semantically driven video summarization

Junren Huang, Yu Xin *, Jiangbo Qian, Yihong Dong

Ningbo University College of Information Science and Engineering, Ningbo, 315211, ZheJiang, China

ARTICLE INFO

Keywords:
Video summarization
Multi-temporal granularity
Concept induction
General semantic

ABSTRACT

The existing video summarization methods mainly focus on extracting and analyzing visual features, but often overlook the higher-level semantic connections between video frames. This approach, while addressing surface-level visual elements, fails to fully understand the complex scenes, characters, and events of videos and their temporal associations, resulting in summaries that lack representativeness. To address this issue, this paper proposes a video summarization method using Multi-Temporal Granularity Concept Induction, MTGC-VS. This method first models the temporal connections and semantic information of the input video through a concept encoder at multiple granularities, identifying the most representative semantic prototypes in the video as video concepts. These concepts are then integrated to form the general semantics of the video. Based on this, a semantic enhancer strengthens the relevance of each frame to general semantics, identifying the key content that aligns best with the general meaning, enhancing the semantic consistency between the summary and the original video, and making the summary more semantically representative. The performance of the proposed method is validated through extensive experiments conducted on the TVSum and SumMe datasets and compared with that of the current state-of-the-art methods. The experimental results demonstrate the effectiveness of the MTGC-VS method.

1. Introduction

Early video summarization techniques primarily relied on simple frame sampling or rule-based methods. While these methods are straightforward, they often fail to capture the semantic information inherent in video content, leading to summarizations that lack coherence and representativeness. However, with advancements in deep learning technologies, significant progress has been made in the field of video summarization (Money & Agius, 2008). Deep learning approaches, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have proven effective in understanding video content and generating high-quality summarizations. More recently, the Transformer model has also been applied to video processing tasks. The key innovation of the Transformer model lies in its self-attention mechanism, which is crucial for understanding long-term dependencies in video content (Vaswani et al., 2017). By learning the features of video data, these methods are able to better capture both the temporal and semantic characteristics of videos.

Despite these advancements, video summarization continues to face significant challenges. Accurately identifying and extracting the most important and representative content from videos remains a complex task. Videos not only contain rich visual information, but also encompass temporal aspects that are crucial for understanding their narrative structures. The significance of an event depends not only on its visual representation but also on its position and duration in the overall narrative, which complicates the identification of key information (Aner & Kender, 2002). Furthermore, with ongoing technological progress, the integration of multimodal data (e.g., video and text) to generate more comprehensive video summarizations has emerged as a prominent research area. Multimodal video summarization combines visual content with information from other modalities, such as text, thereby providing richer context and a deeper understanding of the video content. In this context, we propose the MTGC-VS method based on the Transformer model. This method models multi-granularity temporal relationships within videos using a concept encoder, extracting the most representative concepts in video semantics and integrating multiple concepts to capture the general semantics of the video. Additionally, by leveraging textual information from video descriptions as pseudo-labels, the accuracy of the general semantics is enhanced. Building upon this foundation, our semantic enhancer assesses the relevance of each frame to the general semantics, ensuring the accurate capture of each significance within the video, thereby producing a more representative summary. The primary contributions of this work are as follows:

* Corresponding author.
E-mail addresses: 2211100256@nbu.edu.cn (J. Huang), xinyu@nbu.edu.cn (Y. Xin).

https://doi.org/10.1016/j.eswa.2025.127128
Received 13 January 2025; Received in revised form 6 February 2025; Accepted 1 March 2025
Available online 10 March 2025

(1) To extract the general semantics of a video, we propose a concept encoder. This model identifies the conceptual prototypes within the input video, treating these concepts as high-level semantics, thereby achieving conceptual decoupling and enhancing the interpretability of the video semantics.

(2) We adopt multi-granularity grouping to extract both local and global concepts. This strategy employs a recursive induction method to extract global concepts, ensuring the consistency of these global concepts and the semantic dependency between local and global concepts.

(3) We utilize text features extracted from video captions as labels for the text modality and establish an auxiliary supervision mechanism between the text and video modalities. This ensures semantic consistency across multimodal conditions and enhances the representativeness of the video summary.

2. Related works

Video summarization methods are generally divided into unsupervised and supervised approaches, each of which has contributed significantly to the field. However, despite the progress, there remain key limitations that the proposed method aims to address.

In unsupervised video summarization, techniques such as clustering have been widely used (Basavarajaiah & Sharma, 2021; De Avila et al., 2011; Wu et al., 2017), but these methods struggle with adapting to changes in shot complexity, leading to summaries that may fail to capture the essential temporal dynamics. To overcome this, some researchers have turned to adversarial learning and reinforcement learning. For example, Mahasseni et al. (2017) introduced a generative adversarial framework that combines an LSTM-based summarizer and a discriminator, achieving more informative summaries. Yuan et al. (2019) incorporated a cycle consistency loss to reduce information loss, and Zhou et al. (2018) proposed the Deep Summarization Network (DSN), which assigns selection probabilities to frames to determine their inclusion in the summary. Other studies such as Apostolidis et al. (2020) and Lei et al. (2018) utilized reinforcement learning, specifically actor-critic models, to navigate the labor-intensive and ambiguous labeling process, while Lan and Ye (2021) combined VAE-LSTM with a Pointer Network and de-redundancy mechanism to generate compact summaries. However, despite these advances, unsupervised methods still face challenges in fully capturing the temporal evolution and semantic richness of videos, which often results in summaries that do not reflect the true content in the intended way. The proposed method seeks to address these gaps by incorporating video captions alongside visual features, effectively enriching the semantic representation and enhancing the temporal coherence of the generated summaries.

In supervised video summarization, methods rely on human-annotated data, which allows the summaries to align more closely with human perspectives. Gong et al. (2014) introduced the SeqDPP method, which maintains temporal dependencies in video content. Building on this, RNN-based methods have shown considerable promise due to their ability to model sequential data. Zhao et al. (2017) employed bidirectional LSTMs to capture both forward and backward temporal relationships, and Ji et al. (2020) enhanced the performance by replacing the mean squared error loss with Huber loss to mitigate the impact of anomalous video content. Despite their success, RNN-based models are prone to short-term memory issues, which hinder their ability to capture long-range dependencies. To address this, more recent approaches have adopted Transformer-based models and attention mechanisms. Fajtl et al. (2019) utilized global self-attention for key frame detection, while Zhu et al. (2022) proposed a multi-scale hierarchical attention method to learn both local and global features. Apostolidis et al. (2021) and Li et al. (2022) further refined these models by combining global and local attention or considering cross-video semantic dependencies. While Transformer-based methods have improved the temporal modeling of videos, they still largely rely on visual features extracted by traditional CNNs, which may lack the nuanced semantic information necessary for generating high-quality summaries. To address this limitation, the proposed method introduces a multimodal framework that aligns visual features with text features derived from video captions, enabling the integration of both temporal and semantic information in a more cohesive and contextually rich manner.

Despite the success of both unsupervised and supervised approaches, there are still significant challenges, particularly regarding the integration of temporal dynamics and semantic content. Clustering-based methods and even more advanced adversarial and reinforcement learning models can fail to adequately represent the temporal evolution of video content, while RNN and Transformer-based supervised methods can struggle with semantic richness, often relying on conventional CNNs for visual feature extraction. Our proposed method overcomes these limitations by incorporating video captions into the learning process, enhancing the semantic representation of the video while preserving important temporal information. By multimodally aligning visual and textual features, the proposed method offers a more comprehensive and effective solution to video summarization.

3. The proposed method

The challenge of video summarization lies in understanding the semantic information embedded in a video through its spatiotemporal relationships, ensuring that the generated summary remains semantically consistent with the original video. Video semantics encompass comprehensive representations of various entities, relationships, and scenes, such as characters, objects, actions, and settings. By abstracting and decoupling the semantic features of a video, we can identify a set of the most representative and interpretable prototypes, referred to as video concepts. While video concepts are semantically independent, they are interrelated through the temporal structure of the video, with each concept playing a role in the overall video semantics and exhibiting temporal dependencies with one another.

For a video V consisting of N frames, denoted as $V = \{f_i | 1 \le i \le N\}$, a set of video concepts $\boldsymbol{C} = \{\boldsymbol{c}_i | 1 \le i \le m\}$ can be induced based on the spatial features of each frame f_i and the temporal features between each frame and its adjacent frames. These video concepts represent the features of the main entities. By integrating the video concepts $\boldsymbol{C}$, the general semantics $\boldsymbol{A}$ can be obtained and subsequently used to score the importance of the frame sequence, thereby selecting a video summarization from V.

In response, this paper presents the MTGC-VS method. The overall process is illustrated in Fig. 1.

3.1. Concept encoder

3.1.1. Extraction of global concepts

Global concepts capture the general semantics of videos. Since they remain unique and are unaffected by time granularity, the global concepts extracted at different time scales are consistent. In this approach, we group video frames at multiple time granularities to generate global sequences with varying temporal resolutions. By leveraging the conceptual consistency within each group of global sequences, we then extract global concepts using the concept encoder.

In the concept encoder, we use a set of learnable concept embeddings $\boldsymbol{C}^{(G)}$ to represent multiple global concepts. As shown in the upper half of Fig. 2, for the input video $V = \{f_i | 1 \le i \le N\}$, each frame f_i is processed through a pretrained image encoder to obtain its spatial features $\boldsymbol{x}_i$, resulting in a feature sequence representation for V denoted as $\boldsymbol{X} = \{\boldsymbol{x}_i | 1 \le i \le N\} \in \mathbb{R}^{N \times d}$, where d represents the feature dimensionality. By grouping $\boldsymbol{X}$ with time granularities α, β, and γ, three types of global sequences $\boldsymbol{X}^{(G)}$ can be obtained at different time granularities, as shown in Eq. (1).

$$\boldsymbol{X}_i^{(G_\alpha)} = \{\boldsymbol{x}_i, \ldots, \boldsymbol{x}_{(\lceil N/\alpha \rceil - 1)\alpha + i} | 1 \le i \le \alpha\}, \tag{1a}$$

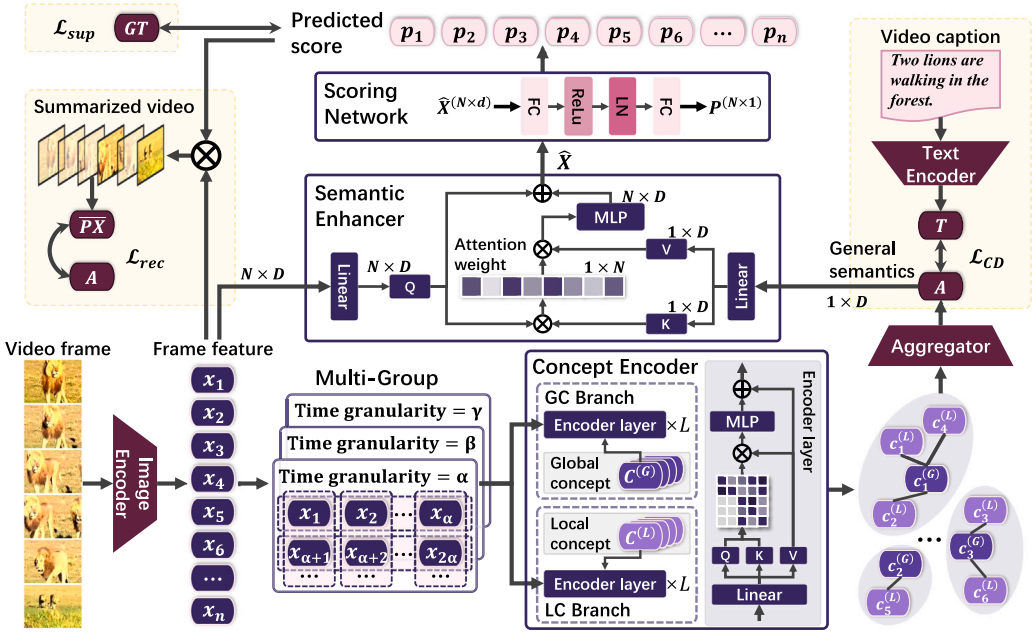


Fig. 1. Overview of the proposed architecture. For each input video, features of every video frame are extracted through a pretrained image encoder and mapped to a frame embedding. The embedding sequence is recombined through multi-group modules and then inputted into the concept encoder, which aims to induce video concepts that run through the frames, analyzing features related to the overall semantic relevance of the video. The general semantics obtained through concept integration is supervised by video captions. The frame embedding sequence and general semantics are taken as inputs to enhance the key semantic features within each frame. A scoring network evaluates the sequence for frame importance scoring.

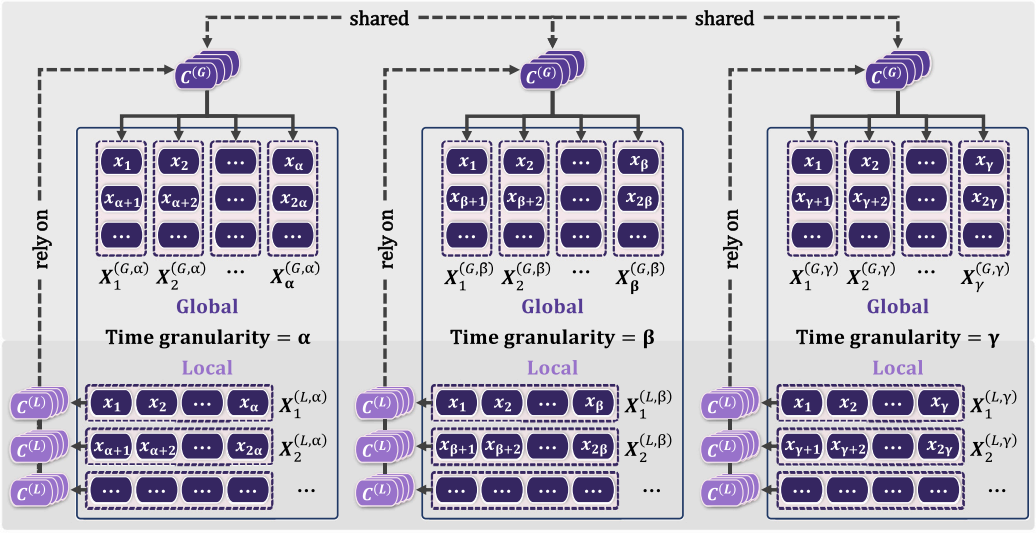


Fig. 2. The concept encoder consists of two branches: global and local. Each column corresponds to a different temporal granularity (α, β, and γ). The upper row represents the global branch of shared global concepts $\boldsymbol{C}^{(G)}$. The bottom row represents the local branch that independently extracts local concepts $\boldsymbol{C}^{(L)}$.

$$\boldsymbol{X}_i^{(G_\beta)} = \left\{\boldsymbol{x}_i, \ldots, \boldsymbol{x}_{([N/\beta]-1)\beta+i} | 1 \le i \le \beta\right\}, \tag{1b}$$

$$\boldsymbol{X}_i^{(G_\gamma)} = \left\{\boldsymbol{x}_i, \ldots, \boldsymbol{x}_{([N/\gamma]-1)\gamma+i} | 1 \le i \le \gamma\right\}. \tag{1c}$$

Here $\boldsymbol{X}_i^{(G)}$ represents the ith global sequence $\boldsymbol{X}^{(G)}$. By concatenating $\boldsymbol{C}^{(G)} = \left\{\boldsymbol{c}_i^{(G)} | 1 \le i \le m_G\right\}$ and $\boldsymbol{X}_i^{(G)}$, we obtain the sequence shown in Eq. (2), which serves as the input to the global concept branch in the concept encoder.

$$\boldsymbol{g}_i^0 = \left[\left\{\boldsymbol{c}_i^{(G)} | 1 \le i \le m_G\right\}; \boldsymbol{X}_i^{(G)}\right]. \tag{2}$$

The global concept branch consists of L encoder layers. In each layer, the input sequence is linearly mapped to obtain the query (Q), key (K), and value (V) needed to compute the temporal dependency. The attention weights are then calculated as shown in Eq. (3). To enhance global concept extraction and reduce the influence of initialization noise on the concept features, we apply a Mask$(\cdot)$ operation to mask the weights between $\boldsymbol{C}^{(G)}$ and $\boldsymbol{X}_i^{(G)}$. The result is then passed through a multi-layer perceptron (MLP) to obtain the encoded result of that layer. The process for the ℓth layer is shown in Eq. (4).

$$\text{Attention} = \text{softmax}\left(\text{Mask}\left(\frac{Q \cdot K^T}{\sqrt{d}}\right)\right)V, \tag{3}$$

$$\boldsymbol{g}_i^\ell = \text{Encoder Layer}^{(G)}\left(\boldsymbol{g}_i^{\ell-1}\right). \tag{4}$$

To maintain consistency among video concepts across multiple global sequences, the set $\left\{\boldsymbol{c}_i^{(G)} | 1 \le i \le m_G\right\}$ extracted from $\boldsymbol{X}_i^{(G)}$ is concatenated with the next global sequence $\boldsymbol{X}_{i+1}^{(G)}$ to form $\boldsymbol{g}_{i+1}^0$, as shown

in Eq. (5).

$$\boldsymbol{g}_{i+1}^{0} = \left[\left\{\boldsymbol{c}_i^{(G)} | 1 \le i \le m_G\right\}; \boldsymbol{X}_{i+1}^{(G)}\right]. \tag{5}$$

All global sequences sequentially undergo the process, which enables the induction of a consistent set of m_G global concepts $\left\{\boldsymbol{c}_i^{(G)} \mid 1 \le i \le m_G\right\}$ across multiple temporal granularities. This cyclical process is illustrated in Eq. (6).

$$\boldsymbol{g}_{i+1}^{\ell} = \text{Encoder Layer}^{(G)}\left(\boldsymbol{g}_{i+1}^{\ell-1}\right). \tag{6}$$

When selecting multiple time granularities, to maintain the diversity of global concepts and avoid excessive dependence on the local semantics, we adopt a coprime processing approach. The rule is as follows: taking α as the base, β is set to $2\alpha - k_1$, and γ is set to $2\beta - k_2$, where k_1 and k_2 control the coprime relationship. This method generates global sequences $\boldsymbol{X}_i^{(G_\alpha)}$, $\boldsymbol{X}_i^{(G_\beta)}$, and $\boldsymbol{X}_i^{(G_\gamma)}$, minimizing temporal receptive field overlaps between sequences, thus obtaining more diverse global temporal relationships. By modeling these sequences through the concept encoder, consistent global concept features can be effectively induced.

3.1.2. Extraction of local concepts

Compared to global concepts, local concepts exhibit greater diversity and contain more detailed semantics. Therefore, when extracting local concepts, it is crucial to preserve their semantic dependency on global concepts. Additionally, since videos are a form of streaming data, local concepts are temporally sensitive and can drift over time. Thus, it is essential to account for the impact of this drift on global concepts when extracting local ones.

In response, we use local sequences with different temporal receptive fields for concept extraction, performing local concept extraction separately. This method ensures the diversity and temporal sensitivity of local concepts while maintaining their dependency on global concepts across different time granularities.

As shown in the lower half of Fig. 2, the features $\boldsymbol{X} = \{\boldsymbol{x}_i | 1 \le i \le N\}$ of length N are divided into groups using three time granularities: α, β, and γ. $\boldsymbol{X}$ is partitioned into non-overlapping local sequences $\boldsymbol{X}^{(L)}$ under these different time granularities, as shown in Eq. (7).

$$\boldsymbol{X}_i^{(L_\alpha)} = \left\{\boldsymbol{x}_{(i-1)\alpha+1}, \ldots, \boldsymbol{x}_{i\alpha} | 1 \le i \le \lceil \frac{N}{\alpha} \rceil\right\}, \tag{7a}$$

$$\boldsymbol{X}_i^{(L_\beta)} = \left\{\boldsymbol{x}_{(i-1)\beta+1}, \ldots, \boldsymbol{x}_{i\beta} | 1 \le i \le \lceil \frac{N}{\beta} \rceil\right\}, \tag{7b}$$

$$\boldsymbol{X}_i^{(L_\gamma)} = \left\{\boldsymbol{x}_{(i-1)\gamma+1}, \ldots, \boldsymbol{x}_{i\gamma} | 1 \le i \le \lceil \frac{N}{\gamma} \rceil\right\}. \tag{7c}$$

Here, $\boldsymbol{X}_i^{(L)}$ denotes the ith local sequence of $\boldsymbol{X}^{(L)}$. For each local sequence $\boldsymbol{X}_i^{(L)}$, we propose using a set of learnable concept embeddings $\boldsymbol{C}^{(L,j)}$ to represent the local concepts within it, as shown in Eq. (8). Each $\boldsymbol{X}_i^{(L)}$ is concatenated with $\boldsymbol{C}^{(L,j)} = \left\{\boldsymbol{c}_i^{(L,j)} | 1 \le i \le m_L\right\}$ to form $\boldsymbol{l}_i^0$. Through the concept encoder, $\boldsymbol{C}^{(L,j)}$ induces the local concepts in the sequence, where m_L denotes the number of $\boldsymbol{c}_i^{(L,j)}$ and j represents the index within the sequence set.

$$\boldsymbol{l}_i^0 = \left[\left\{\boldsymbol{c}_i^{(L,j)} | 1 \le i \le m_L\right\}; \boldsymbol{X}_i^{(L)}\right]. \tag{8}$$

In the concept encoder, the local concept branch follows a structure similar to the global branch, consisting of multiple encoding layers. $\boldsymbol{l}_i^0$ serves as the input to this branch. The self-attention mechanism models the temporal relationships of elements in $\boldsymbol{X}_i^{(L)}$, enabling $\boldsymbol{C}^{(L,j)}$ to extract conceptual features within the local sequence. Similar to the global concept induction process, to mitigate the impact of noise during feature initialization, a masking operation, as shown in Eq. (3) is applied to the attention weights in the self-attention process of the local concept branch. The process for the lth layer is shown in Eq. (9).

$$\boldsymbol{l}_i^{\ell} = \text{Encoder Layer}^{(L)}\left(\boldsymbol{l}_i^{\ell-1}\right). \tag{9}$$

After obtaining the output $\boldsymbol{l}_i^L$, the $\boldsymbol{C}^{(L,j)}$ component is separated from the sequence to form the local concept corresponding to the local sequence $\boldsymbol{X}_i^{(L)}$. This local concept extraction process is applied sequentially to all local sequences, similar to $\boldsymbol{X}_i^{(L)}$. For $\lceil \frac{N}{\alpha} \rceil + \lceil \frac{N}{\beta} \rceil + \lceil \frac{N}{\gamma} \rceil$ local sequences, m local concept features $\left\{\boldsymbol{c}_i^{(L)} | 1 \le i \le m\right\}$ are obtained, where m is defined in Eq. (10).

$$m = m_L \times \left(\lceil \frac{N}{\alpha} \rceil + \lceil \frac{N}{\beta} \rceil + \lceil \frac{N}{\gamma} \rceil\right). \tag{10}$$

These local concepts semantically belong to the global video concepts and capture finer-grained conceptual features from different local segments, ensuring the diversity of local concepts. Additionally, we assess the similarity between the local and global concepts by calculating the Euclidean distance between each feature in $\left\{\boldsymbol{c}_i^{(L)} | 1 \le i \le m\right\}$ and $\left\{\boldsymbol{c}_i^{(G)} | 1 \le i \le m_G\right\}$. This calculation establishes a subordinate relationship between each $\boldsymbol{c}_i^{(L)}$ and $\boldsymbol{c}_i^{(G)}$, as shown in Eq. (11).

$$D_{ij} = \sqrt{\sum_{k=1}^{d}\left(\boldsymbol{c}_{i,k}^{(L)} - \boldsymbol{c}_{j,k}^{(G)}\right)^2}. \tag{11}$$

For each $\boldsymbol{c}_i^{(L)}$, a lower value of D_{ij} indicates that $\boldsymbol{c}_i^{(L)}$ is closer to $\boldsymbol{c}_j^{(G)}$. In this paper, the negative Euclidean distance between each local concept feature and all global concept features is classified and converted into one-hot encoding using the argmax method, identifying the corresponding global concept for each local concept. Since the argmax technique does not support gradient backpropagation, we introduce random noise sampled from the Gumbel$(0, 1)$ distribution. The Gumbel-argmax method, as shown in Eq. (12), is employed to obtain the one-hot results, enabling gradient backpropagation.

$$S_{ij} = \frac{\exp\left(-D_{i,j} + \mu_{i,j}\right)}{\sum_k \left(-D_{i,k} + \mu_{i,k}\right)}, \tag{12a}$$

$$\hat{\boldsymbol{S}}_i = \text{one-hot}\left(\boldsymbol{S}_i\right) - \text{sg}\left(\boldsymbol{S}_i\right) + \boldsymbol{S}_i. \tag{12b}$$

In this case, μ_i is sampled from the Gumbel$(0, 1)$ distribution, and sg$(\cdot)$ denotes the stop-gradient operation. As a result, all local concept features are assigned to the global concept features that are most similar to them.

3.2. Frame selection

3.2.1. Video semantics

The general semantics $\boldsymbol{A}$ of video V comprehensively represent the video concepts; thus, $\boldsymbol{A}$ can be characterized by the global concepts $\boldsymbol{C}^{(G)}$ and local concepts $\boldsymbol{C}^{(L)}$. In constructing the video semantics $\boldsymbol{A}$ from the global concepts $\boldsymbol{C}^{(G)}$ and local concepts $\boldsymbol{C}^{(L)}$, it is essential to account for the varying importance of the video concepts in the overall semantic expression. To achieve this, we first perform feature fusion by incorporating the average values of all local concepts $\boldsymbol{c}_j^{(L)}$ dependent on each global concept $\boldsymbol{c}_i^{(G)}$, yielding refined video concepts $\left\{\boldsymbol{c}_i | 1 \le i \le m_G\right\}$. Then, the dependency relationships among these $\left\{\boldsymbol{c}_i | 1 \le i \le m_G\right\}$ are modeled. Concepts with stronger dependency relations are assigned greater weights in the general semantics of the video. Based on the dependency relationships, we calculate the relevance weight of each concept to the general semantics $\boldsymbol{A}$ of the video, integrating the weighted concepts to obtain the final general semantics $\boldsymbol{A}$.

To calculate the weights between concepts, we employ an aggregator to model the dependency relationships among $\left\{\boldsymbol{c}_i | 1 \le i \le m_G\right\}$ and utilize an attention matrix to calculate the importance W_i of each concept $\boldsymbol{c}_i$. The calculation process is presented in Eq. (13).

$$W_i = \text{softmax}\left(\sum_{j=1}^{m_G} \text{Mask}\left(\frac{\boldsymbol{c}_i \cdot \boldsymbol{c}_i^T}{\sqrt{d}}\right)_{ij}\right), \tag{13}$$

where Mask$(\cdot)$ denotes the process of masking the diagonal elements of the attention matrix. A higher value of W_i indicates that $\boldsymbol{c}_i$ is deemed

more important in the general semantics. The weights $\{W_i|1 \le i \le m_G\}$ are used to weight the concepts $\{c_i|1 \le i \le m_G\}$, and the weighted video concept features are subsequently integrated to obtain the general semantic features $\boldsymbol{A}$ of the video, as shown in Eq. (14).

$$\boldsymbol{A} = \sum_{i=1}^{m_G} W_i \boldsymbol{c}_i. \tag{14}$$

3.2.2. Key frame selection

To ensure that the general semantics of the summarized video align with those of the original video, it is essential to select frames from the original video that exhibit high semantic similarity to its general semantics to serve as key frames in the summary. To achieve this, we use the semantic features $\boldsymbol{A}$ as the foundation for key frame selection. By modeling the dependency relationships between frame features and general semantics, we compute importance scores for the frames, thereby completing the key frame selection process for the summary.

The semantic enhancer and scoring network designed in this paper, as illustrated in Fig. 1, take the feature sequence $\{\boldsymbol{x}_i|1 \le i \le N\}$ and the semantic features $\boldsymbol{A}$ as inputs to the encoder, performing linear mapping on both. To enable each $\boldsymbol{x}_i$ to focus on its relevance to $\boldsymbol{A}$, the mapped result of $\boldsymbol{X}$ serves as Q, while the mapped results of $\boldsymbol{A}$ are used as K and V. Through the cross-attention layer within the encoder, relevance weights between the two are obtained. These weights are then used to encode $\boldsymbol{X}$, which is subsequently processed by MLP, yielding the encoded features for each frame, denoted as $\hat{\boldsymbol{X}} = \{\hat{\boldsymbol{x}}_i|1 \le i \le N\}$. The calculation formula for $\hat{\boldsymbol{X}}$ is shown in Eq. (15).

$$\hat{\boldsymbol{X}} = \mathrm{MLP}\left(\mathrm{softmax}\left(\frac{W_q\boldsymbol{X} \cdot W_k\boldsymbol{A}^{\mathrm{T}}}{\sqrt{d}}\right) \cdot W_v\boldsymbol{A}\right), \tag{15}$$

where $W_{q,k,v}$ represents the weights used for linear mapping and d denotes the dimensionality of the features. $\hat{\boldsymbol{X}}$ reflects the correlation between the original frame features $\boldsymbol{X}$ and the general semantics. By inputting $\hat{\boldsymbol{X}}$ into the scoring network, as shown in Fig. 1, importance scores $P = \{p_i|1 \le i \le N\}$ are obtained, where p_i indicates the importance of each frame f_i in the video. Consequently, the frames with the highest importance scores p_i can be selected as the key frames.

3.3. Training process

The training process of the proposed method is divided into two main stages. The first stage focuses on supervising the training process involving the general semantic features $\boldsymbol{A}$ to improve its accuracy in capturing the general semantics of the input video. The second stage involves supervised training to generate importance scores for the video frames. In this stage, alongside the mean squared error (MSE) loss $\mathcal{L}_{sup}$ for supervising the importance scores, a reconstruction loss $\mathcal{L}_{rec}$ is introduced to measure the semantic similarity between the summarized and the original video.

3.3.1. Stage one

Videos often contain significant visual redundancy in expressing general semantics. In contrast, a text description can convey the general semantics of a video more concisely and accurately using natural language. Building on this characteristic, Stage One uses the text description of a video as a label to supervise the semantic features $\boldsymbol{A}$ derived from the integration of video concepts. We employs a pretrained CLIP model as the encoder for both video frames and text descriptions so that the resulting frame features $\{\boldsymbol{x}_i|1 \le i \le N\}$ and the text description features $\boldsymbol{T}$ are mapped to the same latent space. The concept encoder then extracts video concepts from $\{\boldsymbol{x}_i|1 \le i \le N\}$, yielding video concepts $\boldsymbol{C}$, which are integrated using an aggregator to obtain the semantic features $\boldsymbol{A}$. The concept encoder and aggregator are supervised using a cosine distance (CD) loss, as shown in Eq. (16). The range of $\mathcal{L}_{CD}$ is [0,1], and the closer $\mathcal{L}_{CD}$ is to 0, the closer the integrated semantics are to the video caption. The process is described in Algorithm 1.

$$\mathcal{L}_{CD} = 1 - \frac{\boldsymbol{A} \cdot \boldsymbol{T}}{\|\boldsymbol{A}\| \, \|\boldsymbol{T}\|}. \tag{16}$$

Algorithm 1 Stage One

```
Input:
  Video frames V = {f_i | 1 ≤ i ≤ N}
  Text description
for epoch = 1 to 60 do
  for batch in training data do
    Get video frames V and text description
    X = {x_i | 1 ≤ i ≤ N} ← Pretrained CLIP model on frames
    T ← Pretrained CLIP model on text description
    for granularity ∈ {α, β, γ} do
      C^(G) ← Global concept features as detailed in Sec 3.1.1
      C^(L) ← Local concept features as detailed in Sec 3.1.2
    end for
    A ← Aggregator integrates the video concepts C^(G) and C^(L)
    L_CD ← CD loss as defined in Eq. (16)
    Backpropagate L_CD and update model parameters
  end for
end for
```

3.3.2. Stage two

The outputs from the semantic enhancer, $\hat{\boldsymbol{X}}$, are fed into a regression network to generate importance scores $P = \{p_i|1 \le i \le N\}$ for all video frames. During this stage, the mean squared error (MSE) loss is used as the loss function to compare P with the ground truth (GT), as shown in Eq. (17). This MSE loss supervises the training of both the semantic enhancer and the scoring network.

$$\mathcal{L}_{sup} = \frac{1}{n}\sum_{k=1}^{n}\left(p_i - GT_i\right)^2. \tag{17}$$

On the other hand, the larger value of p_i for each frame f_i indicates greater semantic alignment with the general semantics of the video. To ensure that the summary better aligns with the general semantics of original video, we use p_i as the weight for the corresponding frame. A reconstruction loss is then computed between the general semantics of the weighted sequence $\{p_i\boldsymbol{x}_i|1 \le i \le N\}$ and $\boldsymbol{A}$, as shown in Eq. (18).

$$\mathcal{L}_{rec} = \left|\frac{1}{n}\sum_{i=1}^{n} p_i\boldsymbol{x}_i - \boldsymbol{A}\right|. \tag{18}$$

In Stage Two, the overall loss function is given by Eq. (19), where λ_1 and λ_2 are the weight coefficients that balance these two terms. The process is described in Algorithm 2.

$$\mathcal{L} = \lambda_1\mathcal{L}_{sup} + \lambda_2\mathcal{L}_{rec}. \tag{19}$$

4. Experiments

To assess the effectiveness of the proposed MTGC-VS method, we conduct extensive experiments. This section begins by introducing the experimental setup, including the datasets used, evaluation metrics, and implementation details. The performance of the proposed method is demonstrated through a comparison of its F-score with those of existing methods. Additionally, ablation experiments are performed to analyze the contributions of the key modules in the model. Finally, we present the visual results of the video summaries generated by the MTGC-VS method.

Algorithm 2 Stage two

Input:
  Video frames $V = \{f_i | 1 \le i \le N\}$
  Ground truth $GT = \{GT_i | 1 \le i \le N\}$
for epoch = 1 to 60 **do**
  for batch in training data **do**
    Get video frames V and GT
    $\boldsymbol{X} = \{\boldsymbol{x}_i | 1 \le i \le N\} \leftarrow$ Pretrained CLIP model on frames
    for granularity $\in \{\alpha, \beta, \gamma\}$ **do**
      $\boldsymbol{C}^{(G)} \leftarrow$ Global concept features as detailed in Sec 3.1.1
      $\boldsymbol{C}^{(L)} \leftarrow$ Local concept features as detailed in Sec 3.1.2
    end for
    $\boldsymbol{A} \leftarrow$ Aggregator integrates the video concepts $\boldsymbol{C}^{(G)}$ and $\boldsymbol{C}^{(L)}$
    $\hat{\boldsymbol{X}} \leftarrow \boldsymbol{X}$ is enhanced via $\boldsymbol{A}$, as defined in Eq. (15).
    $\mathrm{FC1}_{out} = nn.linear1(X)$
    $\mathrm{ReLU}_{out} = nn.relu(\mathrm{FC1}_{out})$
    $\mathrm{LN}_{out} = nn.layernorm(\mathrm{ReLU}_{out})$
    $P = nn.linear2(\mathrm{LN}_{out})$
    $\mathcal{L}_{sup} \leftarrow$ MSE loss as defined in Eq. (17)
    $\mathcal{L}_{rec} \leftarrow$ Reconstruction loss as defined in Eq. (18)
    $\mathcal{L} \leftarrow$ total loss as defined in Eq. (19)
    Backpropagate $\mathcal{L}$ and update model parameters
  end for
end for

Table 1
Summary of the datasets used (Annots: Number of annotations).

Dataset	Videos	Annots	Duration	Purpose
TVSum (Song et al., 2015)	50	20	2–11 min	Train & Test
SumMe (Gygli et al., 2014)	25	15–18	1–6 min	Train & Test
OVP (De Avila et al., 2011)	45	5	1–4 min	Train

4.1. Dataset

The training procedure in Stage One requires multi-modal video and description data. MSR-VTT (Xu et al., 2016) is a widely-used dataset in video retrieval (Hao et al., 2023; Huang et al., 2023; Kim et al., 2023; Pei et al., 2023; Wu, Luo et al., 2023) and video caption generation (Chen et al., 2023; Munusamy, 2023; Tang et al., 2022; Wu, Liu et al., 2023; Yang et al., 2023). It contains 10,000 video clips, each ranging from 10 to 30 s, with 20 different descriptions per video. The dataset covers a broad range of themes and activities, ensuring strong diversity. For this study, we select 1000 video clips and their corresponding descriptions from MSR-VTT. Each video is downsampled to 3 frames per second (fps) to construct the training dataset for this stage.

Building on the foundation established in Stage One, Stage Two utilizes TVSum (Song et al., 2015) and SumMe (Gygli et al., 2014) datasets for training, with performance evaluations conducted on both. To further augment the training data for the transfer and augmentation experiments, we include the original videos from the OVP (De Avila et al., 2011) dataset. All videos in these datasets are downsampled to 2 frames per second. In the canonical experiment, 80% of the data is used for training, while the remaining 20% is reserved for testing. Consistent with prior work, all OVP data from the augmented and transfer experiments are employed for training. Table 1 summarizes key details of the three datasets. To ensure robust results, evaluation is performed across five random splits of the dataset, and the average values are reported.

4.2. Evaluation metrics

To ensure a fair comparison with other existing methods, we evaluate the performance of the tested models using the F-score (F). The F-score balances the precision (P) and recall (R) metrics, as defined in Eq. (20). Precision indicates the proportion of frames in the generated summary video V_S that also appear in the ground-truth summary V_T; a higher precision value means that V_S contains more key frames. Recall, on the other hand, quantifies the proportion of frames in V_T that are accurately selected in V_S; a higher recall value signifies that V_S captures more of the important content from the original video.

$$P = \frac{|V_S \cap V_T|}{|V_S|}, \quad R = \frac{|V_S \cap V_T|}{|V_T|}, \tag{20a}$$

$$F = 2 \times \frac{PR}{P+R} \times 100\%. \tag{20b}$$

A representative summary should capture more key content while minimizing redundancy, achieving a balance between high recall and high precision. The F-score ensures that the evaluation is not biased toward either metric, offering a comprehensive and balanced assessment.

Table 2
Comparisons with state-of-the-art methods in Canonical setting by F-score (%).

Methods	SumMe	TVSum	Average
dppLSTM (Zhang et al., 2016)	38.6	54.7	46.6
SUM-GAN (Mahasseni et al., 2017)	41.7	56.3	49.0
H-RNN (Zhao et al., 2017)	42.1	60.2	51.1
Cycle-SUM (Yuan et al., 2019)	44.8	58.1	51.4
SASUM (Wei et al., 2018)	45.3	58.2	51.7
HSA-RNN (Zhao et al., 2018)	44.1	59.8	51.9
SUM-FCN (Rochan et al., 2018)	47.5	56.8	52.1
TTH-RNN (Zhao et al., 2020)	44.3	60.2	52.2
DHAVS (Lin et al., 2022)	45.6	60.8	53.2
GAN-VS (Zhong et al., 2021)	51.7	59.6	55.6
SUM-GDA (Li et al., 2021)	52.8	58.9	55.8
SABTNet (Fu & Wang, 2021)	50.7	61.0	55.8
Adv-Ptr-Der-SUM (Lan & Ye, 2021)	47.7	64.5	56.1
MPFN (Khan et al., 2024)	51.9	62.4	57.1
DN-VSN (Zang et al., 2023)	52.0	62.8	57.4
MC-VSA (Liu et al., 2020)	51.6	63.7	57.6
MIMRN (Wu et al., 2024)	48.3	59.1	53.7
VJMHT (Li et al., 2022)	50.6	60.9	55.7
STVT (Hsu et al., 2023)	55.1	67.1	61.1
AMFM (Zhang et al., 2024)	51.8	61.0	56.4
MTGC-VS (Ours)	**57.1**	**65.8**	**61.4**

4.3. Implementation details

In the experiment, we use the pretrained CLIP-VIT-B/32 model as the backbone for encoding features derived from both text descriptions and video frames. To extract features at multiple granularities from the encoded sequences, we set the group lengths to [4, 7, 13], which enhances the ability to generalize across videos of varying lengths. For groups that are too short, the features of the last frame are used as padding. In the concept encoder, both the global concept (GC) and local concept (LC) branches consist of three encoding layers, with four learnable global and local concepts for each. The cross-encoding layer depth in the semantic enhancer is set to one. To reduce the risk of overfitting, dropout rates of 0.3 and 0.5 are applied to the global and local concept branches, respectively. The model is trained for 100 epochs, with an initial learning rate of 10^{-5} and a weight decay rate of 10^{-4}. A cosine annealing schedule is used to dynamically adjust the learning rate, helping to prevent overfitting during training.

4.4. Results comparison

To demonstrate the effectiveness of the proposed method, this paper compares its performance with that of some of the current state-of-the-art methods on the TVSum and SumMe datasets. The comparison results are shown in Table 2. For the sake of conducting a fair comparison, the results of all the above methods used for comparison purposes are taken from their original papers.

Table 3
With state-of-the-art methods under the augmented and transfer settings.

Methods	SumMe		TVSum	
	Aug	Transfer	Aug	Transfer
dppLSTM (Zhang et al., 2016)	42.9	41.8	59.6	58.7
SUM-GAN (Mahasseni et al., 2017)	43.6	–	61.2	–
DR-DSN (Zhou et al., 2018)	43.9	42.6	59.8	58.9
DASP (Ji et al., 2020)	47.0	–	64.5	–
M-AVS (Ji et al., 2019)	46.1	–	61.8	–
DHAVS (Lin et al., 2022)	46.5	43.5	61.2	57.5
VASNet (Fajtl et al., 2019)	51.1	43.3	63.3	56.7
VJMHT (Li et al., 2022)	51.7	46.4	61.9	58.9
LHMA (Zhu et al., 2022)	52.8	45.4	62.8	55.1
SUM-GDA (Li et al., 2021)	54.4	46.9	60.1	59.0
DN-VSN (Zang et al., 2023)	53.2	50.6	61.8	60.9
MC-VSA (Liu et al., 2020)	53.0	48.1	64.0	59.5
STVT (Hsu et al., 2023)	55.9	48.2	67.7	59.9
MTGC-VS (Ours)	**58.3**	**52.5**	**66.3**	**60.5**

Table 4
Ablation study of proposed Module by F-score (%).

Text-modal	Multi-group	SumMe	TVSum
–	–	54.2	63.7
✓	–	56.7	64.8
–	✓	55.8	65.0
✓	✓	**57.1**	**65.8**

Table 5
Different group lengths settings by F-score (%).

Group sizes	SumMe	TvSum
4, 7	55.5	64.3
4, 7, 13	**57.1**	**65.8**
4, 7, 11, 13	56.1	65.8
7, 9	55.8	64.4
7, 9, 11	56.3	65.4
7, 9, 11, 13	56.5	65.8
9, 11	53.8	64.8
9, 11, 13	54.7	65.2
9, 11, 13, 17	54.4	65.4

Table 6
Different Num of m_G and m_L settings by F-score (%).

m_G	m_L	SumMe	TvSum
2	2	54.6	62.9
	4	55.1	63.3
	6	54.8	63.8
4	2	56.7	65.1
	4	**57.1**	**65.8**
	6	56.3	65.9
6	2	54.7	65.2
	4	56.0	65.8
	6	55.2	65.6

As shown in Table 2, the MTGC-VS method achieves the best performance to date in terms of the average measure on both the SumMe and TVSum datasets. MTGC-VS outperforms all previously developed methods, demonstrating its robustness and effectiveness in generating high-quality video summarizations. Notably, the performance of most of the previously developed methods varies significantly between the two datasets, with an average difference of 12.9%. In contrast, MTGC-VS reduces this difference to 8.7%, significantly narrowing the performance gap between the two datasets. This improvement is due to the concept encoder's ability to extract key semantic objects, which provides clearer guidance for keyframe selection and leads to more consistent and accurate summaries across datasets of varying lengths and characteristics.

When we examine the results in Table 3, where the MTGC-VS method is evaluated under the augmented and transfer settings, the performance improvement becomes even more evident. In the transfer setting, MTGC-VS achieves a 4.3% improvement on SumMe and a 0.6% improvement on TVSum. These results are significant, particularly given the limited size of the training dataset compared to that used by other methods. The transfer experiments utilize the OVP dataset, which has shorter videos (1–6 min) compared to the TVSum dataset (2–11 min). The relatively smaller dataset size and shorter video lengths in the OVP and SumMe datasets may initially limit the ability to learn the temporal dependencies required for longer videos. However, by expanding the dataset with TVSum and OVP, the model's ability to handle both long and short videos improves, resulting in noticeable performance gains in validation.

4.5. Ablation study

In this section, we conduct an ablation study to assess the contribution of various components of the proposed model. The study focuses on the text modality encoder, the multi-group module design, the choice of group sizes in the multi-group module, and the number of encoding layers in the concept encoder. These experiments aim to understand how these different factors influence the performance and generalization capabilities on the video summarization task.

For models that do not include the text modality, we only uses the CLIP image encoder to extract features from the given video frames, skipping stage one and proceeding directly to stage two of the model training process. This experimental modification does not result in a change in the number of model parameters. For models that do not include the multi-group module, we use 7 as the single fixed group size and follow the two-stage training process described in Section 3.3, which also does not affect the number of model parameters.

From Table 4, it is evident that both the text modality encoder and the multi-group module contribute significantly to the performance of the model. When neither of these components is included, the model achieves an F-score of 54.2% on the SumMe dataset and 63.7% on TVSum. The inclusion of the text modality encoder improves the F-scores to 56.7% and 64.8%, respectively, indicating that text descriptions provide valuable semantic guidance to the model, allowing it to capture more meaningful video concepts. Similarly, when the multi-group module is introduced, the F-scores improve to 55.8% and 65.0%, suggesting that temporal dependencies are better captured, which aids in understanding the sequential nature of the video. Finally, when both components are used together, the performance reaches its peak with an F-score of 57.1% on SumMe and 65.8% on TVSum, demonstrating the synergistic effect of the two modules in improving the ability to capture both semantic and temporal information effectively.

The results in Table 5 show that the choice of group sizes in the multi-group module has an impact on performance. Smaller group sizes (e.g., [4, 7]) work well for shorter videos, as they allow better capture of finer temporal dependencies. However, as the size of the group increases, as in [4, 7, 13], the model better captures long-range dependencies, which is especially beneficial for longer videos like those in the TVSum dataset. The results show that the optimal group length configuration for both datasets is [4, 7, 13], as this setup consistently produces the highest F-scores. Moreover, combining additional group lengths such as [4, 7, 11, 13] does not lead to substantial improvement and may slightly degrade performance in some cases, indicating that excessive fragmentation of the video into smaller temporal chunks may interfere with the capture of global dependencies.

As shown in Table 6, the analysis of the learnable embeddings m_G and m_L reveals that increasing the number of embeddings does not always lead to a proportional increase in performance. For instance, when m_G is set to 4 and m_L to 4, the model achieves the highest F-score of 57.1% on SumMe and 65.8% on TVSum. However, increasing the number further (e.g., $m_G = 6$, $m_L = 6$) results in slight performance degradation on SumMe and no significant improvement on TVSum,

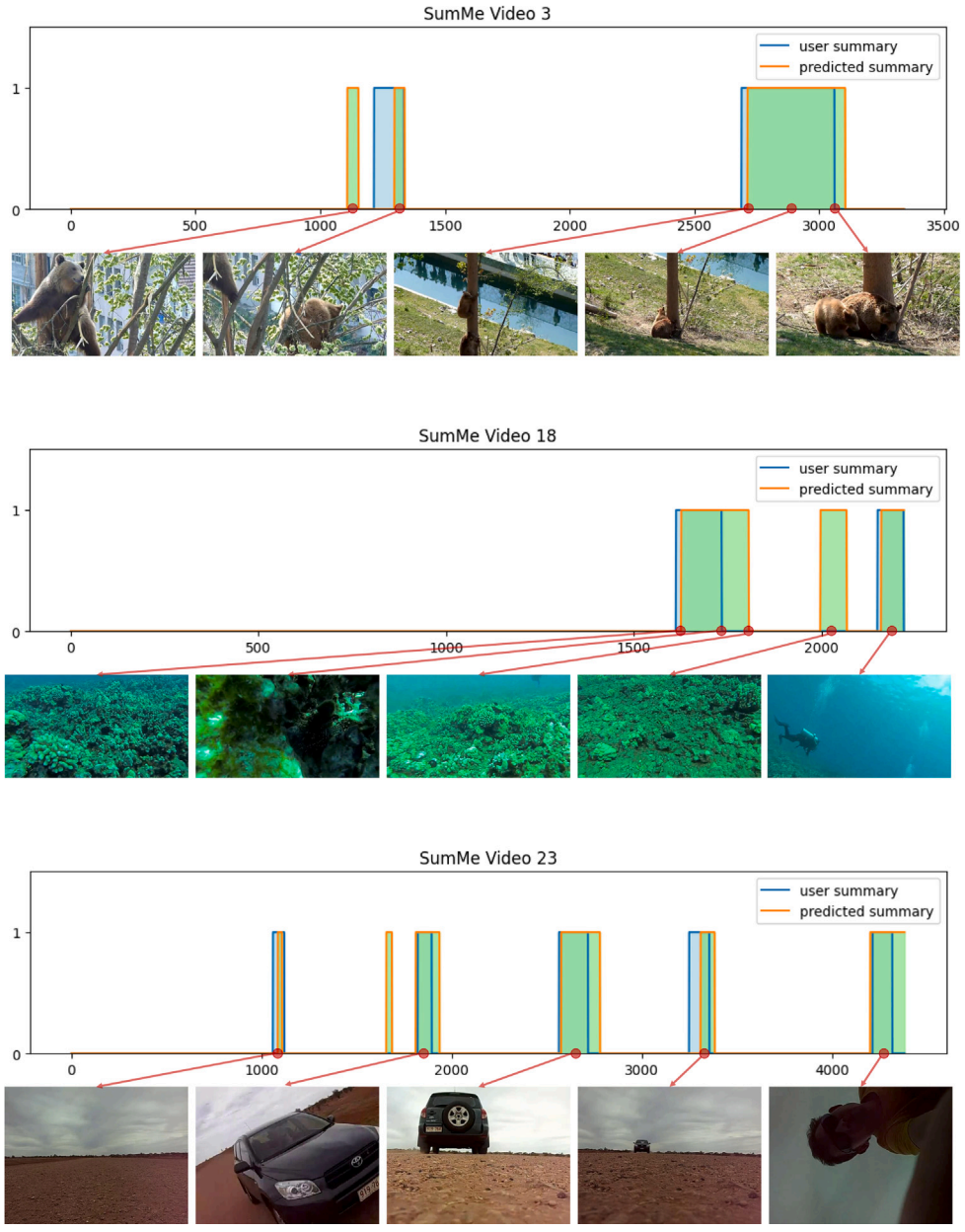


Fig. 3. The visualization results of videos 3, 18, and 23 in SumMe, where the blue area represents the user summary, and the yellow area represents the summary we predicted. A larger overlap between the two areas indicates a closer match between our predicted summary and the user summary.

suggesting that the model might already have sufficient capacity with 4 embeddings. This finding underscores that the number of learnable embeddings in the Transformer structure should be chosen based on the complexity of task and data characteristics. Simply increasing the quantity does not guarantee better performance unless supported by sufficient training data and task-specific design.

The experimental findings emphasize the importance of balancing the complexity of the model (in terms of both group size and number of encoding layers) with the nature of the data (e.g., video length, diversity, and complexity). For instance, smaller group sizes and fewer layers may be more suitable for datasets with shorter and simpler videos like SumMe, while larger group sizes and deeper models are better suited for more complex and longer videos like those in TVSum. Thus, the choice of group sizes and model depth should be carefully tuned to maximize both efficiency and effectiveness.

4.6. Result discussion

The experimental results demonstrate the effectiveness of the proposed MTGC-VS framework in addressing key challenges of video summarization, particularly in handling datasets with varying video lengths and characteristics. The superior performance across both SumMe and TVSum datasets, coupled with reduced inter-dataset performance variance, highlights the robustness of the framework. This success can be attributed to three critical design elements: (1) the integration of cross-modal semantic guidance through text-visual alignment, (2) the hierarchical temporal modeling enabled by multi-group configurations, and (3) the adaptive concept encoding mechanism that balances local and global context aggregation. To visually demonstrate the effectiveness of our method, we present qualitative results in Figs. 3 and 4. Fig. 3 compares the predicted selection results (yellow area) with user

Fig. 4. The visualization results of videos 24, 30, and 38 in TVSum. To better distinguish the overlapping areas, we use blue to represent the user summary and red to represent the predicted summary. Based on the predicted summary, we selected 5 frames from the original video.

summaries (blue area) for videos 3, 18, and 23 from the SumMe dataset. The spatial distribution of user summaries indicates that content selection is primarily driven by semantic relevance rather than temporal patterns. Through cross-modal semantic alignment, our method effectively captures key semantic features from temporal content, generating summaries that preserve the original semantic integrity while eliminating redundant information. This enables our approach to accurately identify and include content segments that align with user summaries. Fig. 4 showcases the generation results (red area) for videos 24, 30, and 38, demonstrating our method's versatility across diverse content. The evaluated videos encompass various themes including animals, vehicles, and human activities, captured from both first-person and third-person perspectives. The generated summaries consistently maintain high semantic correspondence with user-provided references, effectively capturing relevant themes and perspectives across different video types.

The ablation study offers practical insights for model configuration. While larger group sizes (13 frames) prove essential for modeling long-range dependencies in TVSum videos (2–11 min), smaller groups (4 frames) effectively capture rapid scene changes prevalent in SumMe videos (1–4 min). This explains why the hybrid configuration [4,7,13] achieves optimal performance. The layer depth experiments further reveal that four encoding layers provide an optimal balance between model capacity and generalization.

The augmented and transfer learning experiments underscore the framework adaptability, where MTGC-VS maintains strong performance when trained on OVP (1–6 min videos) and tested on TVSum. This cross-dataset generalization capability, evidenced by the 52.5% and 60.5% F-score in transfer mode, suggests that the learned semantic-temporal representations are not dataset-specific.

Two fundamental limitations warrant discussion. First, the current text modality processing relies on prepared video captions, which may not fully capture domain-specific semantics in specialized videos (e.g., medical procedures). Second, the fixed group size combinations require manual configuration based on dataset statistics. Future work could explore dynamic group size prediction and domain-adaptive concept encoding to address these limitations.

5. Conclusion

This paper presents MTGC-VS, a novel video summarization method that integrates multi-granular concept encoding and semantic enhancement to improve the quality of video summaries. The framework primarily consists of a concept encoder and a semantic enhancer. The concept encoder models both global and local semantics of the video, enriching the overall semantic representation. The semantic enhancer filters key content by evaluating the relevance of each frame to the overall semantics, ensuring that the summary aligns with the essential content of the original video.

Through extensive experiments on benchmark datasets, we demonstrate that MTGC-VS outperforms existing methods by generating summaries that retain more important content while minimizing redundancy. The approach effectively addresses the challenge of balancing the preservation of video semantics and the reduction of superfluous information.

Despite these achievements, there are areas for further exploration. The integration of multimodal information, such as audio and text, could further enhance the robustness of the summarization process. Additionally, we plan to investigate the optimization of reward functions within the reinforcement learning framework to improve summary quality and explore methods to enhance model interpretability. Our future work will also focus on improving the efficiency and scalability of the model to better suit real-world applications.

CRediT authorship contribution statement

Junren Huang: Conceptualization, Methodology, Software, Validation, Writing – original draft. **Yu Xin:** Methodology, Writing – review & editing, Supervision. **Jiangbo Qian:** Supervision. **Yihong Dong:** Supervision.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Yu Xin reports financial support was provided by Natural Science Foundation of Zhejiang Province. Yu Xin reports financial support was provided by the 3315 Plan Foundation of Ningbo. Yu Xin reports financial support was provided by China Natural Science Foundation. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We acknowledge the support of the Natural Science Foundation of Zhejiang Province, China (Grant No. LY22F020001, No. LZ20F020001), the 3315 Plan Foundation of Ningbo (Grant No. 2019B-18-G), China Natural Science Foundation under Grant 62271274 and the support of Research and Application of A Multi Billion Parameter Monitoring Video Model for domestic full stack AI infrastructure, China (Grant No. 2024Z004).

Data availability

Data will be made available on request.

References

Aner, A., & Kender, J. R. (2002). Video summaries through mosaic-based shot and scene clustering. In *Computer vision—ECCV 2002: 7th European conference on computer vision copenhagen, Denmark, May 28–31, 2002 proceedings, part iV 7* (pp. 388–402). Springer.

Apostolidis, E., Adamantidou, E., Metsai, A. I., Mezaris, V., & Patras, I. (2020). AC-SUM-GAN: Connecting actor-critic and generative adversarial networks for unsupervised video summarization. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(8), 3278–3292.

Apostolidis, E., Balaouras, G., Mezaris, V., & Patras, I. (2021). Combining global and local attention with positional encoding for video summarization. In *2021 IEEE international symposium on multimedia* (pp. 226–234). IEEE.

Basavarajaiah, M., & Sharma, P. (2021). GVSUM: generic video summarization using deep visual features. *Multimedia Tools and Applications*, *80*(9), 14459–14476.

Chen, J., Pan, Y., Li, Y., Yao, T., Chao, H., & Mei, T. (2023). Retrieval augmented convolutional encoder-decoder networks for video captioning. *ACM Transactions on Multimedia Computing, Communications and Applications*, *19*(1s), 1–24.

De Avila, S. E. F., Lopes, A. P. B., da Luz Jr, A., & de Albuquerque Araújo, A. (2011). VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern Recognition Letters*, *32*(1), 56–68.

Fajtl, J., Sokeh, H. S., Argyriou, V., Monekosso, D., & Remagnino, P. (2019). Summarizing videos with attention. In *Computer vision–ACCV 2018 workshops: 14th Asian conference on computer vision, perth, Australia, December 2–6, 2018, revised selected papers 14* (pp. 39–54). Springer.

Fu, H., & Wang, H. (2021). Self-attention binary neural tree for video summarization. *Pattern Recognition Letters*, *143*, 19–26.

Gong, B., Chao, W.-L., Grauman, K., & Sha, F. (2014). Diverse sequential subset selection for supervised video summarization. *Advances in Neural Information Processing Systems*, *27*.

Gygli, M., Grabner, H., Riemenschneider, H., & Van Gool, L. (2014). Creating summaries from user videos. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VII 13* (pp. 505–520). Springer.

Hao, X., Zhang, W., Wu, D., Zhu, F., & Li, B. (2023). Dual alignment unsupervised domain adaptation for video-text retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 18962–18972).

Hsu, T.-C., Liao, Y.-S., & Huang, C.-R. (2023). Video summarization with spatiotemporal vision transformer. *IEEE Transactions on Image Processing*.

Huang, S., Gong, B., Pan, Y., Jiang, J., Lv, Y., Li, Y., & Wang, D. (2023). Vop: Text-video co-operative prompt tuning for cross-modal retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6565–6574).

Ji, Z., Jiao, F., Pang, Y., & Shao, L. (2020). Deep attentive and semantic preserving video summarization. *Neurocomputing*, *405*, 200–207.

Ji, Z., Xiong, K., Pang, Y., & Li, X. (2019). Video summarization with attention-based encoder–decoder networks. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*(6), 1709–1717.

Khan, H., Hussain, T., Khan, S. U., Khan, Z. A., & Baik, S. W. (2024). Deep multi-scale pyramidal features network for supervised video summarization. *Expert Systems with Applications*, *237*, Article 121288.

Kim, J., Lee, Y., & Moon, J. (2023). T2V2T: Text-to-video-to-text fusion for text-to-video retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5612–5617).

Lan, L., & Ye, C. (2021). Recurrent generative adversarial networks for unsupervised WCE video summarization. *Knowledge-Based Systems*, *222*, Article 106971.

Lei, J., Luan, Q., Song, X., Liu, X., Tao, D., & Song, M. (2018). Action parsing-driven video summarization based on reinforcement learning. *IEEE Transactions on Circuits and Systems for Video Technology*, *29*(7), 2126–2137.

Li, H., Ke, Q., Gong, M., & Zhang, R. (2022). Video joint modelling based on hierarchical transformer for co-summarization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(3), 3904–3917.

Li, P., Ye, Q., Zhang, L., Yuan, L., Xu, X., & Shao, L. (2021). Exploring global diverse attention via pairwise temporal relation for video summarization. *Pattern Recognition*, *111*, Article 107677.

Lin, J., Zhong, S.-h., & Fares, A. (2022). Deep hierarchical LSTM networks with attention for video summarization. *Computers & Electrical Engineering*, *97*, Article 107618.

Liu, Y.-T., Li, Y.-J., & Wang, Y.-C. F. (2020). Transforming multi-concept attention into video summarization. In *Proceedings of the Asian conference on computer vision*.

Mahasseni, B., Lam, M., & Todorovic, S. (2017). Unsupervised video summarization with adversarial lstm networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 202–211).

Money, A. G., & Agius, H. (2008). Video summarisation: A conceptual framework and survey of the state of the art. *Journal of Visual Communication and Image Representation*, *19*(2), 121–143.

Munusamy, H. (2023). Multimodal attention-based transformer for video captioning. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, *53*(20), 23349–23368.

Pei, R., Liu, J., Li, W., Shao, B., Xu, S., Dai, P., Lu, J., & Yan, Y. (2023). Clipping: Distilling clip-based models with a student base for video-language retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 18983–18992).

Rochan, M., Ye, L., & Wang, Y. (2018). Video summarization using fully convolutional sequence networks. In *Proceedings of the European conference on computer vision* (pp. 347–363).

Song, Y., Vallmitjana, J., Stent, A., & Jaimes, A. (2015). Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5179–5187).

Tang, M., Wang, Z., Zeng, Z., Li, X., & Zhou, L. (2022). Stay in grid: Improving video captioning via fully grid-level representation. *IEEE Transactions on Circuits and Systems for Video Technology*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Wei, H., Ni, B., Yan, Y., Yu, H., Yang, X., & Yao, C. (2018). Video summarization via semantic attended networks. *Vol. 32*, In *Proceedings of the AAAI conference on artificial intelligence*.

Wu, B., Liu, B., Huang, P., Bao, J., Peng, X., & Yu, J. (2023). Concept parser with multi-modal graph learning for video captioning. *IEEE Transactions on Circuits and Systems for Video Technology*.

Wu, W., Luo, H., Fang, B., Wang, J., & Ouyang, W. (2023). Cap4video: What can auxiliary captions do for text-video retrieval? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10704–10713).

Wu, G., Song, S., Wang, X., & Zhang, J. (2024). Reconstructive network under contrastive graph rewards for video summarization. *Expert Systems with Applications*, *250*, Article 123860.

Wu, J., Zhong, S.-h., Jiang, J., & Yang, Y. (2017). A novel clustering method for static video summarization. *Multimedia Tools and Applications*, *76*, 9625–9641.

Xu, J., Mei, T., Yao, T., & Rui, Y. (2016). Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5288–5296).

Yang, B., Cao, M., & Zou, Y. (2023). Concept-aware video captioning: Describing videos with effective prior information. *IEEE Transactions on Image Processing*.

Yuan, L., Tay, F. E. H., Li, P., & Feng, J. (2019). Unsupervised video summarization with cycle-consistent adversarial lstm networks. *IEEE Transactions on Multimedia*, *22*(10), 2711–2722.

Zang, S.-S., Yu, H., Song, Y., & Zeng, R. (2023). Unsupervised video summarization using deep non-local video summarization networks. *Neurocomputing*, *519*, 26–35.

Zhang, K., Chao, W.-L., Sha, F., & Grauman, K. (2016). Video summarization with long short-term memory. In *Computer vision–ECCV 2016: 14th European conference, amsterdam, the netherlands, October 11–14, 2016, proceedings, part VII 14* (pp. 766–782). Springer.

Zhang, Y., Liu, Y., & Wu, C. (2024). Attention-guided multi-granularity fusion model for video summarization. *Expert Systems with Applications*, *249*, Article 123568.

Zhao, B., Li, X., & Lu, X. (2017). Hierarchical recurrent neural network for video summarization. In *Proceedings of the 25th ACM international conference on multimedia* (pp. 863–871).

Zhao, B., Li, X., & Lu, X. (2018). Hsa-rnn: Hierarchical structure-adaptive rnn for video summarization. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7405–7414).

Zhao, B., Li, X., & Lu, X. (2020). TTH-RNN: Tensor-train hierarchical recurrent neural network for video summarization. *IEEE Transactions on Industrial Electronics*, *68*(4), 3629–3637.

Zhong, R., Wang, R., Zou, Y., Hong, Z., & Hu, M. (2021). Graph attention networks adjusted bi-LSTM for video summarization. *IEEE Signal Processing Letters*, *28*, 663–667.

Zhou, K., Qiao, Y., & Xiang, T. (2018). Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. *32*, In *Proceedings of the AAAI conference on artificial intelligence*. (1).

Zhu, W., Lu, J., Han, Y., & Zhou, J. (2022). Learning multiscale hierarchical attention for video summarization. *Pattern Recognition*, *122*, Article 108312.