Video Summarization using Attention Mechanisms and CNN

Ajinkya Somawanshi, Devang Shirodkar, Vinayak Yadav, Krushna Tawri, Prof. Rakhi Punwatkar

[1234] *UG Student, Dept. of computer Engineering, Zeal college of engineering, Maharashtra, India*

[5]*Professor, Dept. of computer Engineering, Zeal college of engineering, Maharashtra, India*

---***---

Abstract - This Video summarization is a vital task in multimedia analysis, especially given the vast volume of video data in the digital age. Although deep learning methods have been extensively studied for this purpose, they often face challenges in efficiently processing long-duration videos. This paper tackles the issue of unsupervised video summarization by introducing a novel approach that selects a sparse subset of video frames to optimally represent the original video. The core concept involves training a deep summarizer network within a generative adversarial framework, which includes an autoencoder LSTM network as the summarizer and another LSTM network as the discriminator. The summarizer LSTM is designed to select key video frames and reconstruct the input video from these selected frames. Concurrently, the discriminator LSTM's role is to differentiate between the original video and its reconstruction. Through adversarial training between the summarizer and discriminator, combined with sparsity regularization, the network learns to produce optimal video summaries without requiring labeled data. Evaluations on several benchmark datasets indicate that this method delivers competitive performance compared to fully supervised state-of-the-art techniques, highlighting its effectiveness in unsupervised video summarization.

Key Words: Event summarization, Critical information in videos, Surveillance systems ,Video analysis, Multimedia analysis, Deep learning, Unsupervised learning, Autoencoder LSTM, Long short-term memory network (LSTM)

1. INTRODUCTION

Video summarization (VS) has emerged as a crucial technique to address these challenges by condensing lengthy videos into concise representations while preserving key information. The main goal of VS is to streamline video analysis by eliminating unnecessary frames and retaining keyframes, thus facilitating efficient browsing and structured access to video content. Automatic VS (AVS) powered by Artificial Intelligence (AI) is a rapidly growing research area, enabling the summarization of lengthy videos without human intervention.

VS has applications in various domains, including surveillance, education, entertainment, and medical diagnostics. Its practical uses range from monitoring and tracking to creating movie trailers and enabling video search engines. Additionally, VS significantly reduces frame redundancy, optimizing storage requirements and computational time.

This paper addresses the problem of unsupervised video summarization, focusing on selecting a sparse subset of frames that minimizes the representation error between the original video and its summary. We propose a novel approach using a generative adversarial framework, which combines an autoencoder LSTM network as the summarizer and another LSTM network as the discriminator. By training these networks adversarially, our aim is to produce optimal video summaries without the need for labeled data.

We provide an overview of our proposed approach to unsupervised video summarization and discuss its applications in various domains. Additionally, we delve into the technical details of our methodology, including the use of deep learning architectures such as CNNs and LSTMs for feature extraction and the implementation of a generative adversarial network for optimization. Through experimental evaluation on benchmark datasets, we demonstrate the effectiveness of our approach in generating high-quality video summaries.

Overall, this paper contributes to ongoing research in video summarization by introducing a novel unsupervised approach that leverages deep learning and generative adversarial techniques to produce compact and informative video summaries across diverse domains.

2. Related work

The body of the paper consists of numbered sections that present the main findings. These sections should be organized to best present the material.

(i) Problem Formulations

Traditional video summarization methods such as video synopsis and montages compress video content by tracking moving objects or merging keyframes into summary images. However, these methods often fail to maintain the temporal motion layouts. Alternative techniques like hyper-lapses focus on temporal manipulation. Recent advancements have been directed toward storyboard generation, which extracts a subset of representative video frames. Despite this progress, deep learning methods have not been extensively applied in this area.

(ii) Supervised vs. Unsupervised Summarization

Supervised summarization methods depend on human-annotated keyframes for training, aiming to optimize frame selection to minimize loss relative to ground truth annotations. In contrast, unsupervised methods rely on heuristic criteria for keyframe selection. Although transfer learning has shown potential, it presents challenges in ensuring domain correlations. Unsupervised methods are particularly effective in scenarios where human annotations are difficult to obtain, such as military or nursing home settings.

(iii) Deep Learning Approaches

Deep learning techniques, especially Long Short-Term Memory (LSTM) networks, have been used for keyframe selection in both forward and reverse temporal directions. Recurrent auto-encoders are also employed to represent annotated temporal intervals in highlights. LSTM-based models like vsLSTM and dppLSTM focus on structured prediction and diversity enhancement. Additionally, unsupervised generative adversarial learning models like SUM-GAN introduce a novel approach by integrating variational auto-encoder LSTMs and regularization techniques for video summarization.

(iv) Generative Adversarial Networks (GANs)

GANs, initially popular in image processing, have been adapted for video summarization. These models enhance prior methods by incorporating a variational auto-encoder LSTM and appropriate regularization for frame selection. Unlike earlier approaches that primarily rely on discriminators for learning signals, GAN-based models integrate frame selectors, thereby improving the summarization process.

(v) Unsupervised Approaches

Unsupervised techniques are prominent in the field, with clustering-based methods and dictionary learning being common for identifying keyframes. Clustering algorithms group visually similar frames or shots, using cluster centers as representative keyframes. Dictionary learning employs base vectors in the model to reconstruct the original video content, effectively pinpointing keyframes or shots.

(vi) Attention-Based Approaches

Attention-based LSTM frameworks use low-level features like motion and face detection to capture user attention, enabling a deeper understanding of viewer engagement. By modeling attention cues derived from user interactions, these frameworks can identify key shots that align with user preferences, leading to more effective video summarization strategies.

3. Deep learning based approach

Deep learning (DL) has emerged as a powerful paradigm within machine learning, offering various network structures and applications across domains such as cybersecurity, natural language processing, bioinformatics, robotics, and medical information processing. In the context of video summarization (VS), DL methods haveshown remarkable effectiveness and versatility, encompassing supervised, weakly supervised, unsupervised, and reinforcement learning approaches.

3.1 Supervised Learning-Based Video Summarization

Supervised learning techniques in video summarization (VS) involve training models using labeled data to predict future outcomes. However, acquiring well-defined datasets is often expensive and difficult due to the need for domain expertise and the vast diversity of online content. Supervised models are generally divided into classification and regression models, employing algorithms such as linear classifiers, k-nearest neighbors, support vector machines, decision trees, and random forests. Prominent deep learning (DL) techniques used in supervised video summarization include deep belief networks (DBNs), deep neural networks (DNNs), and convolutional neural networks (CNNs), each offering distinct strengths in feature extraction and classification.

DBNs utilize a deep architecture composed of stacked restricted Boltzmann machines (RBMs) to perform feature extraction and classification. DNNs improve model accuracy by incorporating multiple hidden layers. CNNs, renowned for their ability to extract high-level features from video frames, use convolutional and pooling layers to process visual information.

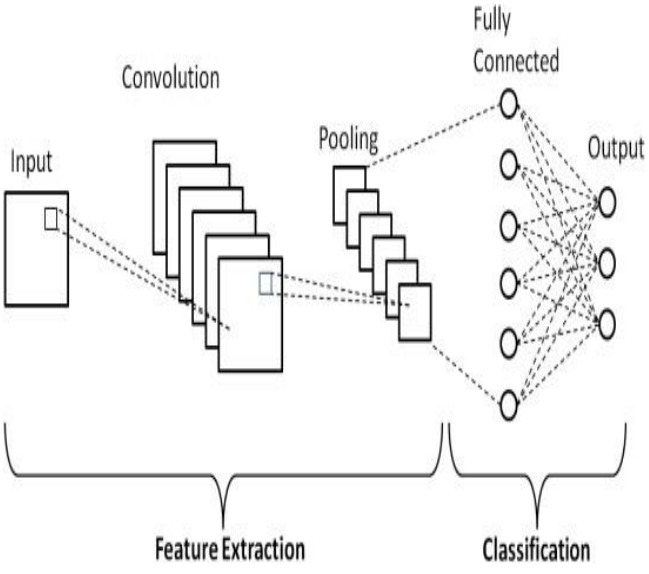


Fig. 3.1 Basic Architecture of CNN

3.2 Unsupervised Learning-Base Video Summarization

Unsupervised learning techniques in VS operate without labeled data, relying on clustering, association, and dimensionality reduction methods like principal component analysis (PCA), k-means clustering, and singular value decomposition (SVD). Generative adversarial networks (GANs) have emerged as a robust unsupervised learning framework for video summarization, enabling the generation of informative summaries through adversarial training between a generator and discriminator network.

3.4 Reinforcement Learning-Based Video Summarization

Reinforcement learning (RL) approaches in video-summarization (VS) involve sequential decision-making processes, where an agent learns to maximize rewards through trial and error. RL-based methods employ hierarchical LSTM networks, 3D spatiotemporal U-Nets, and diverse reward functions to produce comprehensive and representative video summaries, effectively adapting to different video content and lengths

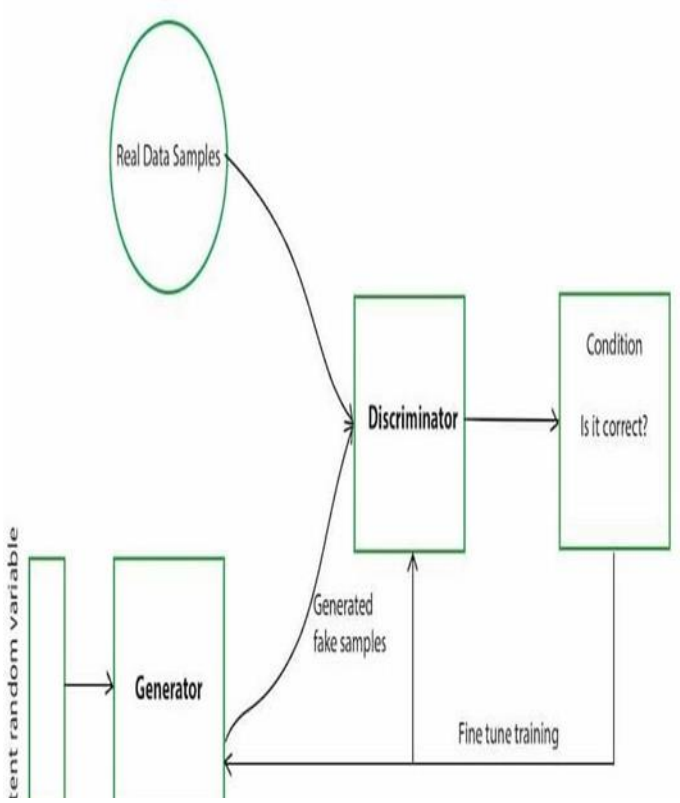


Fig. 3.3 General Working of Generative A

4. REVIEW OF VAE AND GAN

Variational Autoencoder (VAE) is a powerful generative model that creates a distribution over observed data given latent variables. It comprises an encoder, which maps input data to a latent space, and a decoder, which reconstructs the input data from this space. The VAE's learning objective involves minimizing the negative log-likelihood of the data distribution, comprising the reconstruction loss and the Kullback-Leibler (KL) divergence term. On the other hand, a Generative Adversarial Network (GAN) is an innovative neural network framework consisting of two competing networks: a generator (G) and a discriminator (D). The generator produces data resembling an unknown distribution, while the discriminator distinguishes between generated samples and real data. GANs aim to optimize the generator to produce data indistinguishable from real data, formulated as a minimax optimization problem. The discriminator maximizes the probability of correctly classifying real and generated samples, while the generator minimizes the probability of the discriminator correctly identifying generated samples, aiming to minimize log(1 − D(G(z))).

5.MAIN COMPONENT OF OUR APPROACH

Our approach consists of two primary components: the summarizer and the discriminator recurrent networks. The summarizer uses a selector LSTM (sLSTM) to choose a subset of frames from the input video, which have been preprocessed using deep features extracted by a CNN. These selected frames are then encoded into a deep feature representation using an encoder LSTM (eLSTM). The system assigns importance scores to guide the frame selection process, which are used to weigh the frame features before they are fed into the eLSTM. Finally, a decoder LSTM (dLSTM) takes the encoded deep features and reconstructs a sequence of features representing the original video.

The discriminator's role is to differentiate between the original video and the reconstructed video, classifying them as 'original' or 'summary,' respectively. It uses a classifier LSTM (cLSTM) with a binary classification output. During training, adversarial techniques are applied, where the cLSTM learns to accurately identify reconstructed sequences as 'summary' while being misled to confuse these sequences with 'original' ones.

Our training strategy employs four loss functions: LGAN, which represents the augmented GAN loss, and Lreconst, the reconstruction loss for the recurrent encoder-decoder. Additionally, an extra frame selector, guided by a prior distribution, generates the encoded representation and the reconstructed feature sequence. The adversarial training of the cLSTM is regulated to ensure it accurately recognizes the reconstructed summary as 'summary' while also being confused between the reconstructed summary and 'original' videos.

In conclusion, our method involves adversarial training between the summarizer (comprising sLSTM, eLSTM, and dLSTM) and the discriminator (cLSTM) until the discriminator can no longer distinguish between the reconstructed summaries and the original videos.

6.SYSTEM REQUIREMENTS

6.1 Software Requirements

Development Environments:

1. Visual Studio Code or appropriate IDE for desktop application development.

2. Flutter for cross-platform compatibility in the desktop application.

3. Appropriate server-side frameworks (e.g., Node.js, Python) for backend services.

Libraries and Resources:

1. Required: Implementation of attention mechanisms for video processing and summarization.

2. Optional: Leveraging external video processing libraries such as PyTorch and datasets like SumMe and TVSum for enhanced feature extraction and training.

6.2 Hardware Requirements

1. Minimum i5 processor for efficient video processing.

2. Minimum 8GB RAM to handle video summarization tasks effectively.

3. High-speed internet connectivity to ensure seamless access to cloud-based resources and models.

4. Adequate storage, either SSD or HDD, to store video files, datasets, and system data.

7.ALGORITHMS

1) KTS (Kernel Temporal Segmentation) :

The Kernel Temporal Segmentation (KTS) algorithm is a robust method for video summarization that segments a video into distinct temporal parts based on changes in visual content. It begins by extracting features from each video frame using techniques like Convolutional Neural Networks (CNNs) to capture visual details. These features are used to construct a kernel matrix, which represents the similarity between frame pairs. By computing the cumulative sum of the kernel matrix, the algorithm identifies significant change points where the content shifts. These change points are used to divide the video into coherent segments, each representing a continuous part with similar content. The optimal segmentation minimizes within-segment variance while maximizing between-segment variance, ensuring meaningful divisions. Key segments or frames are then selected from each segment to generate a concise and representative video summary, making KTS an effective tool for creating highlights, scene detection, and content-based retrieval.

2) Bi-directional LSTM(Bi-lstm) :

Bidirectional Long Short-Term Memory networks (Bi-LSTMs) play a significant role in video summarization by leveraging their ability to capture temporal dependencies in both forward and backward directions. This dual perspective allows Bi-LSTMs to understand the context of each frame relative to its preceding and succeeding frames, leading to more accurate and coherent summaries. In video summarization, Bi-LSTMs process sequences of video frames to extract features that encapsulate the video's temporal dynamics. By considering the entire sequence bidirectionally, these networks can better identify important segments and transitions, ensuring that the generated summaries are both comprehensive and contextually relevant. This makes Bi-LSTMs particularly effective in applications requiring a nuanced understanding of temporal patterns, such as highlight extraction, scene segmentation, and content summarization.’

3) Luong and Bahadanau attention algorithms:

Luong and Bahdanau's attention mechanisms significantly enhance video summarization by allowing models to focus on different parts of the video frames selectively. The Bahdanau attention, often referred to as additive attention, computes alignment scores by combining the hidden states of both the encoder and decoder through a feedforward neural network, enabling the model to focus on specific temporal segments dynamically. Luong attention, known as multiplicative or dot-product attention, calculates alignment scores directly as the dot product between the encoder's hidden states and the decoder's current state, which is computationally efficient. In video summarization, these attention mechanisms enable models to capture and emphasize keyframes and important segments by dynamically weighing the relevance of each frame, thus producing more accurate and contextually rich video summaries.

8. DATA SETS USED IN VS

This section provides an overview of various datasets commonly used for evaluating video summarization (VS) methods, along with different evaluation methodologies. The following datasets are typically used for VS evaluation:

1. TVSum: This dataset comprises 50 videos across various categories, including news, tutorials, user-generated content, and documentaries. Each video, ranging from 2 to 11 minutes, is annotated by 20 individuals based on frame relevance and ratings.

2. SumMe: This dataset includes 25 videos with durations between 1 and 6 minutes, covering diverse topics such as holidays, events, and games. Each video has annotations from 15 to 18 users for identifying key portions.

3. CoSum: This dataset consists of 51 videos with a total length of 4444 minutes, each ranging from 11 to 25 minutes. The videos cover a variety of topics, with annotations to highlight critical segments.

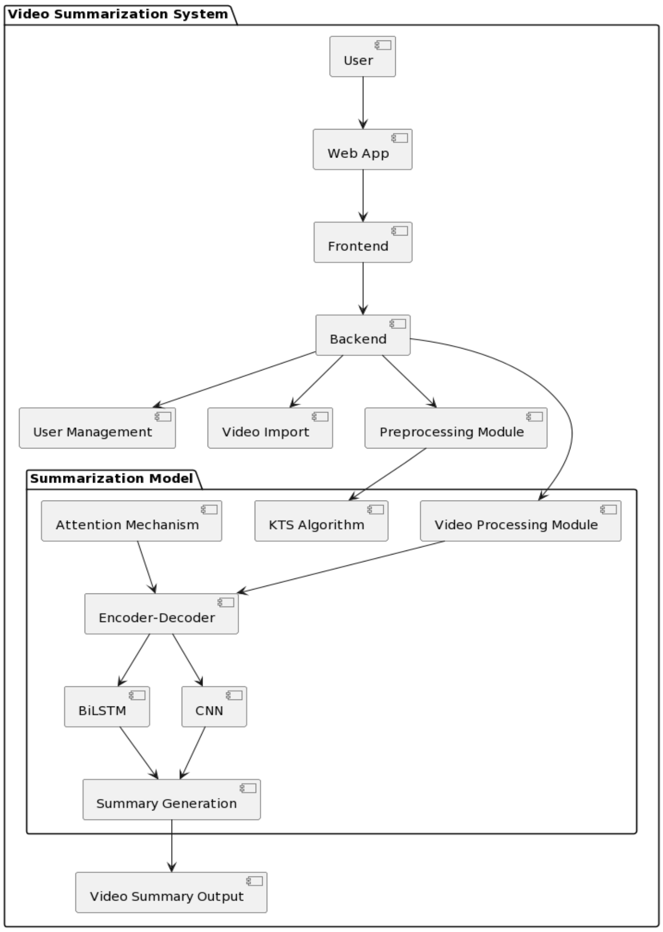
4. Thumk1K: This dataset includes a collection of YouTube

videos on topics such as skydiving, bridge crossings, sports, and cultural landmarks.

5. Open Video Project (OVP) : This dataset contains 50 videos annotated with five different user keyframe sets, providing a basis for evaluating video summarization techniques.

9. SYSTEM DESIGN

9.1 System Architecture



Project Workflow:

User Interface (Web App/Frontend):

- User Registration/Login: Enable secure user registration and login functionalities.
- Input Interface: Provide an intuitive interface for users to upload video files or input video URLs.
- Settings Configuration: Allow users to customize summary parameters such as desired length or percentage.
- Summarization Trigger: Implement a button to initiate the summarization process.
- Display Summary: Showcase the generated video summary effectively.

Backend:

- User Management:
Authentication System: Implement secure user authentication mechanisms.
User Database: Store user credentials and session data securely.

- Video Import:
Upload and Retrieval: Manage video uploads and fetch videos from user-provided URLs.

- Preprocessing Module:
Data Preparation: Preprocess video data for analysis, including segmentation and frame extraction.

- Summarization Model (Attention Mechanism):
Attention Mechanism Implementation: Deploy attention-based models tailored for video summarization tasks.

- KTS Algorithm: Apply the Keyframe Extraction using the Temporal Summarization (KTS) algorithm for frame selection.

- Video Processing Module:
Data Analysis: Implement algorithms for motion detection, object tracking, and scene segmentation.

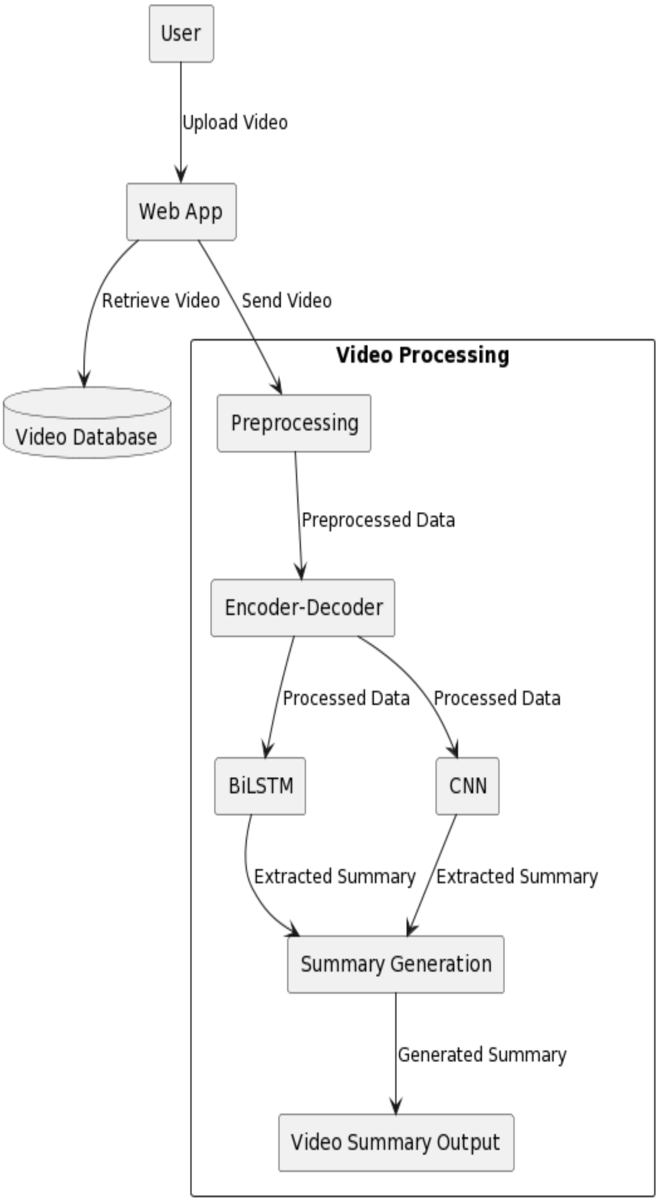
- Encoder-Decoder Module (LSTM, CNN):
Bi-LSTM (Bidirectional LSTM): Utilize bidirectional LSTM networks for sequence modeling in summarization.
CNN (Convolutional Neural Network): Apply CNNs for feature extraction from video frames.

- Summary Generation:
Attention Mechanism for Summary: Utilize attention weights to generate summaries from selected keyframes.
Video Summary Output: Compile selected keyframes into concise and coherent video summaries based on attention and relevance scores.

- Summary Output and User Interface:
Summary Display: Present the generated video summaries on the front end.
Playback Control: Implement functionalities for playing, pausing, and navigating through the summaries.
Download/Share Options: Provide users with options to download or share the summarized videos.

-Deployment and Integration:
Cloud Deployment: Host backend services on cloud platforms like AWS, Google Cloud, or Azure.
Module Integration: Ensure seamless communication and integration between frontend and backend components. Testing and Quality Assurance:
System Validation: Conduct rigorous testing to ensure the accuracy of generated video summaries and a smooth user experience.

9.2 Data Flow Diagram



10. PERFORMANCE MEASURE

The following sections provide an overview of various methods utilized for assessing Video Summarization (VS):

10.1 Static VS Evaluation:

Initially, evaluations relied on detailed criteria like frame relevance and instructional importance, but these were time-consuming and lacked reproducibility. To address this, recent studies have employed objective metrics such as commitment and image reconstruction capacity. For instance, Chasanis et al. assessed summary quality using the commitment standard, while Liu et al. considered image reconstruction capacity. Additionally, comparative analyses of user summaries have been conducted, with F-score evaluations on datasets like TVSum, SumMe, OVP, YouTube, and VSUMM presented.

10.2 Dynamic VS Evaluation:

Initially, user-created datasets were used for evaluation, followed by F-score assessments. A concurrent evaluation process introduced with the SumMe dataset employed predefined criteria similar to BBC. Other methods utilize metrics like the Matthews correlation coefficient or rely on a single ground-truth summary instead of multiple user summaries for evaluation. F-score evaluations for dynamic summaries on datasets like TVSum, SumMe, and YouTube are also provided.

10.3 Observations from F-score evaluations:

- The TVSum dataset is widely utilized, with the Convolutional Neural Network Bi-Convolutional Long Short Term Memory Generative Adversarial Network method achieving a static summary F-score of 69.0%.

- SumMe follows as the second most used dataset, with the Deep Attentive Preserving method achieving a static summary F-score of 45.5%.

- The Multi Convolutional Neural Network excelled on the Open Video Project dataset, achieving a static summary F-score of 82.0%.

- The Multi-edge optimized LSTM RNN method attained the highest static summary F-score of 85.8% on the YouTube dataset.

- For static summaries on the VSUMM dataset, the Multi-edge optimized LSTM RNN method achieved the highest F-score of 92.4%.

- Regarding dynamic summaries, the Convolutional Neural Network Bi-Convolutional Long Short-Term Memory Generative Adversarial Network method reached an F-score of 72.0% on the TVSum dataset.

- The Dilated Temporal Relational-Generative Adversarial Network method performed well on the SumMe dataset, achieving an F-score of 51.4% for dynamic summaries.

- The unsupervised learning-based Cycle-SUM method outperformed, achieving an F-score of 77.3% for generating dynamic summaries.

11. RESULTS

We assess our methodology on four datasets: SumMe, TVSum, Open Video Project (OVP), and YouTube.

1) SumMe: This dataset contains 25 user videos showcasing various events such as cooking and sports, with durations ranging from 1.5 to 6.5 minutes, accompanied by frame-level importance scores.

2) TVSum: Comprising 50 YouTube videos spanning 10 categories, the TVSum dataset exhibits diverse content with video lengths varying from 1 to 5 minutes.

3) OVP: We utilize the same 50 videos as in prior studies, covering various genres and ranging from 1 to 4 minutes in length.

4) YouTube: This dataset consists of 50 videos sourced from online platforms, featuring content like cartoons, news, and sports, with durations ranging from 1 to 10 minutes.

Evaluation Setup: We employ the key shot-based metric proposed in previous research, defining precision and recall based on the temporal overlap between generated and user-annotated key shots. The F-score, serving as the evaluation metric, is the harmonic mean of precision and recall. We follow established procedures to convert frame-level scores to keyframes and key shot summaries across all datasets.

Implementation Details: To ensure fairness in comparison, we utilize the output of the pool5 layer of the GoogLeNet network as the feature descriptor for each video frame. Our framework incorporates a two-layer LSTM with 1024 hidden units for the discriminator LSTM (cLSTM) and two two-layer LSTMs with 2048 hidden units each for eLSTM and dLSTM. We employ a decoder LSTM that reconstructs the feature sequence in reverse order. Parameters of eLSTM and dLSTM are initialized with those of a pre-trained recurrent autoencoder model on original video feature sequences. We utilize the Adam optimizer with default parameters for training.

Baselines: Due to the generative structure of our approach, we cannot entirely replace subnetworks with baselines. Thus, alongside variations of our approach defined in Section 6, we also evaluate other baselines.

Quantitative Results: The model incorporating additional frame-level supervision, SUM-GANsup, outperforms unsupervised variants. Variations with explicit regularization, like SUM-GANdpp and SUM-GANrep, exhibit slightly better performance than SUM-GAN. Generally, SUM-GANdpp outperforms SUM-GANrep. Training with combined losses from VAE and GAN enhances accuracy.

Comparison with State of the Art: Our unsupervised SUM-GANdpp model surpasses all unsupervised approaches across all datasets, achieving nearly 5% improvement over state-of-the-art unsupervised methods on SumMe. SUM-GANsup outperforms supervised methods in all datasets except OVP.

Comparison with Shallow Features: We assess our model using shallow features utilized in prior studies and find that our model consistently outperforms the state of the art, even when shallow features perform better than deep features in some instances.

Qualitative Results: We demonstrate the temporal selection pattern of different approaches using an example video, depicting selected frames and frame-level importance scores. Despite minor variations, all approaches cover temporal regions with high frame-level scores, with most failure cases occurring in videos with very slow motions and no scene changes.

12. CONCLUSION

In conclusion, the exploration of various techniques for video summarization has unveiled a rich array of approaches, each presenting its own set of advantages and drawbacks. From traditional methods like keyframe extraction and clustering to modern deep learning-based approaches, researchers have made significant progress in improving the efficiency and efficacy of video summarization processes. The choice of technique often hinges on the specific needs of the application, whether it be real-time processing, content comprehension, or user preferences.

While traditional methods offer simplicity and computational efficiency, deep learning approaches, particularly those harnessing convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have showcased superior performance in capturing intricate temporal dependencies and semantic nuances. Nonetheless, these methods may encounter challenges concerning computational complexity and the demand for extensive labeled data.

As the field progresses, future research endeavors could concentrate on hybrid methodologies that amalgamate the strengths of both traditional and deep learning techniques. Furthermore, addressing issues related to interpretability, scalability, and the establishment of standardized benchmarks will bolster the broader adoption and evaluation of video summarization methodologies. To sum up, the exploration of diverse techniques for video summarization underscores the significance of tailoring approaches to the specific requirements and constraints of the application. By amalgamating the strengths of various methodologies, researchers can chart a course toward more resilient and adaptable video summarization systems in the days ahead.

REFERENCES

1. Chern C. I. Serrano, V. Shah, and M. D. Abràmoff, “A Semantic Text Summarization of Long Videos” Int. J. Telemed. Appl., vol. 2018, pp. 1–14, Oct. 2021.
2. M. Islam, A. V. Dinh, and K. A. Wahid, ’Video summarization using deep learning techniques’J. Biomed. Sci. Eng., vol. 10, no. 05, pp. 86–96, 2022.
3. Yuan, F. E. Tay, P. Li, L. Zhou, and J. Feng, ‘‘Cycle-sum: Cycleconsistent adversarial lstm networks for unsupervised video summarization,’’ in Proc. AAAI Conf. Artif. Intell., vol. 33, 2019, pp. 9143–9150.
4. García, J. Gallardo, A. Mauricio, J. López, and C. Del Carpio, “The Vid2Seq: LargeScale Pretraining of a Visual Language Model for Dense Video Captioning” in Proc. Int. Conf. Artif. Neural Netw. Cham, Switzerland: Springer, Sep. 2023 , pp. 635–642.
5. Li, Q. Ye, L. Zhang, L. Yuan, X. Xu, and L. Shao, ‘‘Exploring global diverse attention via pairwise temporal relation for video summarization,’’ Pattern Recognit., vol. 111, Mar. 2021, Art. no. 107677.

6. B. Zhao, M. Gong, and X. Li, ‘‘Hierarchical multimodal transformer to summarize videos,’’ Neurocomputing, vol. 468, pp. 360–369, Jan. 2022.
7. B. Zhao, X. Li, and X. Lu, ‘‘HSA-RNN: Hierarchical structure-adaptive RNN for video summarization,’’ in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7405–7414.
8. Mahasseni, M. Lam, and S. Todorovic, ‘‘Unsupervised video summarization with adversarial LSTM networks,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 202–211.
9. D C. L. Giles, G. M. Kuhn, and R. J. Williams, ‘‘Dynamic recurrent neural networks: Theory and applications,’’ IEEE Trans. Neural Netw., vol. 5, no. 2, pp. 153–156, Mar. 1994.