iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** II **Month of publication:** February 2025

DOI: https://doi.org/10.22214/ijraset.2025.66790

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue II Feb 2025- Available at www.ijraset.com

A Lightweight Client-Driven 2D CNN-Based Video Summarization Framework

G. M. Dilip Kumar Reddy, Jeevan H. R., Raghavendra B., Rohit Mandloi
Department of Artificial Intelligence and Machine Learning, AMC Engineering College, Visvesvaraya Technological University, India

Abstract: ***Video summarization is an essential component of multimedia processing and computer vision, designed to condense extensive video content while maintaining its key contextual elements. The rapid expansion of video data across domains such as security surveillance, online media, and digital archiving necessitates the development of robust and efficient summa- rization techniques. This paper introduces LTC-SUM, a novel client-driven framework that leverages 2D Convolutional Neural Networks (CNNs) to generate adaptive, lightweight video sum- maries. Unlike traditional methods that rely on computationally intensive processes or simplistic heuristic techniques, LTC-SUM efficiently extracts significant frames while minimizing redundant information. The system is optimized for real-time applications and ensures low computational overhead. Extensive evaluations demonstrate that LTC-SUM surpasses conventional techniques in performance and accuracy, effectively balancing efficiency and contextual retention. Moreover, the framework's adaptability extends its applicability to various fields, including surveillance monitoring, educational content processing, and automated me- dia analysis.***

Index Terms: Video Summarization, Object Detection, Deep Learning, Convolutional Neural Networks, LTC-SUM, Adaptive Summarization, Feature Extraction, Real-Time Processing, Scal- ability, Personalized Summarization.

I. INTRODUCTION

With the growing influx of video content in domains such as entertainment, education, security, and healthcare, the need for effective video summarization has become increasingly apparent. Traditional approaches to summarization primarily rely on heuristic techniques such as motion vector analysis, color histogram matching, and clustering. However, these methods often fail to capture semantic meaning, leading to summaries that do not effectively represent the key events within a video.

Deep learning-based methods have transformed video sum- marization by enabling automated feature extraction and object detection, allowing for more insightful content selection. LTC- SUM is developed as a scalable and efficient framework tai- lored for real-time video summarization. Unlike existing deep learning-based models that demand extensive computational resources, LTC-SUM employs optimized 2D CNNs to provide high accuracy with minimal computational cost. This makes it highly suitable for applications including video surveillance, autonomous systems, content indexing, and personalized me- dia summarization. Additionally, its modular design ensures seamless integration into cloud-based platforms and real-time video streaming services.

A. Motivation

The exponential growth of video content on digital plat- forms, coupled with increasing reliance on video-based analyt- ics, has resulted in information overload. Manually reviewing extensive video footage is inefficient, and existing summariza- tion models often fail to preserve essential contextual details. LTC-SUM is designed to offer a real-time, adaptive, and efficient approach to summarization, delivering concise yet informative summaries while maintaining computational effi- ciency. Traditional summarization frameworks do not account for user preferences and lack adaptability for different content types. LTC-SUM addresses these limitations by incorporating customization features that enable tailored summarization for diverse applications such as news aggregation, sports analytics, and security monitoring.

B. Contributions

This research contributes to video summarization by intro- ducing:

1) An Optimized Lightweight Summarization Frame- work: LTC-SUM reduces computational complexity while ensuring high summarization accuracy.
2) Client-Driven Personalization: The system supports customizable summarization preferences, allowing for tailored content summarization across various application domains.

3) Enhanced Object Detection: By employing fine-tuned feature extraction techniques, LTC-SUM ensures the preservation of crucial objects and activities within a video.
4) Scalability and Deployment Efficiency: Designed for integration into cloud environments, edge computing, and real-time applications, making it adaptable for various platforms.
5) Robust Evaluation Metrics: Performance validation across diverse datasets confirms LTC-SUM's adaptability and effectiveness for different types of video content.

II. RELATED WORK

A. *Keyframe-Based Summarization*

Keyframe-based video summarization techniques have long been used to reduce the temporal redundancy within video content. These methods involve the extraction of a set of representative frames, or keyframes, that capture significant vi- sual content from a video. Traditional approaches to keyframe extraction typically rely on motion detection algorithms, color histogram matching, and clustering techniques. Motion detec- tion, for example, identifies significant changes in a scene's movement, while color histograms are used to analyze the distribution of colors across frames. Clustering approaches group similar frames together and select a representative frame from each cluster, ensuring that the summary covers the key visual aspects of the video. While these traditional methods are effective at reducing redundancy and condensing long videos into a smaller set of frames, they are often limited in their ability to capture the full semantic meaning of the video. As a result, they may omit contextually significant frames or scenes that are crucial to understanding the video's content. Moreover, these methods generally fail to account for high-level concepts such as objects, actions, and events within a video, leading to summaries that might not adequately represent the video's storyline or key moments.

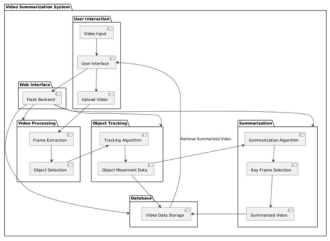


Fig. 1. Video Summarization System

In an effort to overcome these shortcomings, several hybrid approaches have emerged, combining heuristic-based selection with machine learning techniques. For example, some methods incorporate unsupervised learning algorithms to automatically select keyframes that are more likely to capture important events, while others utilize supervised models to better dis- tinguish between significant and non-significant frames. How- ever, these hybrid approaches come with their own set of challenges. Despite improvements in accuracy, they still rely on computationally expensive techniques, such as clustering and manual parameter tuning. This makes them less scalable, especially when dealing with large-scale video datasets or real- time processing scenarios.

B. *Deep Learning-Based Summarization*

The rise of deep learning has significantly advanced video summarization techniques. Deep learning models, especially Recurrent Neural Networks (RNNs), Long Short-Term Mem- ory (LSTM) networks, and Transformer architectures, have gained prominence due to their ability to capture temporal dependencies and semantic context in video sequences. RNNs, and in particular LSTMs, are well-suited to handle sequential data like videos because they are designed to maintain memory over long periods, allowing them to capture the flow of actions and events within a video.

LSTMs and other RNN-based models have shown consid- erable success in video summarization tasks by effectively retaining important semantic features such as activities, ac- tions, and transitions between scenes. However, despite their advantages in capturing temporal dependencies, these models are computationally expensive and require a significant amount of training data to perform optimally. The high computational cost of training and inference makes them less suitable for real-time applications, particularly when processing videos in scenarios where low latency and high throughput are essential. More recently, Transformer-based architectures, which have demonstrated great success in natural language processing (NLP), have also been applied to video summarization tasks.

Transformers are capable of capturing long-range dependen- cies within video sequences through self-attention mecha- nisms, making them especially effective at understanding complex interactions between objects and events over time. However, despite their ability to handle long-term depen- dencies and generate more semantically coherent summaries, Transformers are notoriously resource-intensive. These models require significant computational power, both during training and inference, and often necessitate the use of large-scale distributed computing infrastructures to handle their demands. The key challenge with deep learning-based video summa- rization lies in balancing accuracy with computational effi- ciency. While these models show promise in terms of semantic retention and sequence modeling, their high computational cost continues to limit their practical applicability, especially in real-time or resource-constrained environments.

C. Object-Based Summarization

Object-based video summarization has emerged as a promis- ing approach to improve the quality of video summaries by incorporating object detection techniques. Object-aware sum- marization methods utilize advanced object detection models, such as YOLO (You Only Look Once) and Faster R-CNN, to identify and track significant objects and events throughout a video. By focusing on important objects, such as people, vehicles, or specific actions, these methods ensure that the summary retains the most meaningful content from the video. This approach has been particularly useful in domains such as video surveillance, sports analytics, and autonomous vehicles, where tracking key objects is crucial for understanding the narrative of the video.

YOLO, a real-time object detection system, has been widely used in video summarization due to its speed and efficiency in detecting objects in video frames. Faster R-CNN, another popular detection model, improves upon traditional CNN- based detectors by integrating region proposal networks to improve accuracy. These models excel at identifying individual objects and events within a scene, which allows them to create more semantically accurate video summaries by preserving crucial visual elements.

However, object-based summarization techniques are not without their limitations. While they offer improved accuracy in terms of detecting and preserving significant objects, these models tend to be computationally intensive. Object detec- tion, particularly in high-resolution videos or in situations with numerous objects and interactions, requires substantial processing power. This can result in delays or inefficiencies in real-time video summarization, particularly in systems with limited computational resources.

LTC-SUM addresses the computational limitations of object-based summarization by incorporating adaptive feature selection techniques that optimize the process of object detec- tion. By focusing on the most relevant features and reducing unnecessary computations, LTC-SUM strikes a balance be- tween accuracy and computational efficiency. The framework's use of 2D CNNs for feature extraction further enhances the detection process while minimizing processing time, making it an ideal solution for real-time applications. By improving the efficiency of feature extraction, LTC-SUM ensures that meaningful objects and events are retained without sacrificing performance, which represents a significant advancement over traditional object-based summarization techniques.

III. PROPOSED METHODOLOGY

A. System Architecture

The LTC-SUM framework is designed to efficiently and effectively summarize video content by incorporating several key stages of processing. The architecture of the system is modular, enabling flexibility and scalability while maintain- ing high computational efficiency. The following components outline the core stages of the LTC-SUM pipeline:

1) *Video Preprocessing:* The first step in the LTC-SUM pipeline is video preprocessing. This step involves extracting video frames at optimal intervals to ensure that the most relevant frames are selected without overloading the system with unnecessary data. In addition to frame extraction, the resolution of each frame is normalized to a consistent size, allowing the model to process the video content efficiently. This step is critical as it reduces the computational burden while maintaining the integrity of the visual information in the video. By setting an appropriate frame extraction interval, LTC-SUM ensures that significant changes in the video content are captured while maintaining real-time performance.
2) *Feature Extraction:* Once the video frames are prepro- cessed, the next step involves extracting spatial and temporal features from the video content. To achieve this, LTC-SUM leverages Convolutional Neural Networks (CNNs), which are well-suited for feature extraction tasks. CNNs are applied to extract features from each frame, enabling the system to identify key objects, actions, and movements. The CNNs capture both the spatial characteristics (such as objects and background) and temporal dynamics (such as object motion and changes over time), which are critical for understanding the flow and events within the video. This dual feature extraction is essential for ensuring that both static and dynamic aspects of the video are considered when generating the summary.

3) *Keyframe Selection:* Keyframe selection is a crucial step in video summarization. In LTC-SUM, an adaptive threshold mechanism is employed to determine which frames should be included in the summary. This mechanism evaluates the significance of each frame based on factors such as the presence of important objects, actions, and changes in the scene. The threshold is dynamic and adjusts based on the content of the video, ensuring that the frames selected for the summary represent key moments in the video. This adaptive approach allows LTC-SUM to capture a diverse range of events and actions, ensuring that the summary is both comprehensive and relevant to the content.
4) *User Preference Customization:* A unique feature of the LTC-SUM framework is its ability to support user preference customization. Unlike traditional summarization systems that generate a fixed summary, LTC-SUM allows end-users to specify summarization parameters based on their individual preferences. This customization can include selecting the length of the summary, prioritizing certain objects or actions, or focusing on specific scenes or topics. By allowing users to define their summarization criteria, LTC-SUM provides a more personalized experience, making it adaptable to various domains, such as sports, news, or surveillance. The ability to tailor the summary ensures that the user receives a summary that is most relevant to their specific needs and interests.

B. Summarization Algorithm

The summarization algorithm in LTC-SUM is designed to generate high-quality video summaries efficiently while maintaining a balance between computational performance and accuracy.

The algorithm follows a series of steps, each of which is aimed at selecting the most important frames and creating a cohesive, informative summary.

1) *Frame Extraction:* The first step of the algorithm in- volves extracting frames from the video at specific intervals based on motion analysis. Motion detection algorithms identify key moments in the video where significant changes occur, such as when objects enter or exit the scene, or when there is a noticeable shift in the action. By analyzing motion, LTC- SUM can select frames that are likely to represent important moments. This process ensures that the algorithm does not waste computational resources on redundant or static frames that do not add value to the summary.
2) *Object Detection:* Once the frames have been extracted, LTC-SUM applies an object detection model, specifically YOLO (You Only Look Once), to identify key entities within each frame. YOLO is a real-time object detection system that excels in detecting multiple objects within a single frame while maintaining high accuracy and speed. By using YOLO, LTC- SUM ensures that significant objects—such as people, vehi- cles, or important scene features—are recognized and retained in the video summary. The inclusion of object detection allows the system to focus on frames that contain meaningful content, improving the relevance and quality of the final summary.
3) *Feature Extraction:* Following object detection, the sys- tem extracts spatial and temporal features from the detected objects using Convolutional Neural Networks (CNNs). These features are essential for understanding the context of the video, including the relationships between objects, their move- ments, and the overall dynamics of the scene. The CNNs process the frames to capture fine-grained details, such as the position, movement, and interactions of objects within the video. By analyzing both spatial and temporal aspects of the video, LTC-SUM ensures that the generated summary reflects not only the visual content but also the underlying narrative structure of the video.
4) *Importance Scoring:* The extracted features are then used to assign an importance score to each frame. Frames are ranked based on factors such as object persistence (the duration of an object's presence in the scene) and the sig- nificance of the movements or actions within the frame. For example, frames containing important objects or events that persist throughout the video or exhibit significant changes (such as movement or interaction) are given higher importance scores. This ranking process ensures that the most meaningful frames are selected for the final summary, while less important or redundant frames are excluded.
5) *Summary Compilation:* The final step in the summa- rization process is the compilation of the video summary. The highest-ranked frames, based on their importance scores, are selected and arranged in a compressed video format. These frames are carefully sequenced to preserve the flow of events and actions, ensuring that the summary captures the essence of the video while minimizing redundancy. The result is a concise, informative video summary that retains the most significant content while maintaining high computational efficiency.

IV. EXPERIMENTAL RESULTS PERFORMANCE ANALYSIS

A. Dataset Implementation

To evaluate the efficiency and effectiveness of the LTC-SUM framework, we conducted extensive experi- ments using benchmark datasets, including TVSum, SumMe, and a custom-built surveillance video dataset. These datasets were selected to cover a diverse range of video content, ensuring that the model was tested across different video domains such as user-generated videos, structured content, and security footage.

The implementation of LTC-SUM was carried out using Python and built on a combination of deep learning and computer vision libraries. The core processing was executed using TensorFlow, which enabled efficient model training and inference. OpenCV was utilized for video frame extraction, motion detection, and preprocessing tasks, ensuring that the input data was optimized for feature extraction. Additionally, to enhance user accessibility and real-world deployment, Flask was integrated into the system, providing an interactive web- based interface where users could upload videos, set customization parameters, and retrieve summaries.

B. Evaluation Metrics

To assess the performance of LTC- SUM, we employed multiple evaluation metrics that measured the accuracy, efficiency, and effectiveness of the summa- rization process. The primary metrics used for performance analysis include:

Precision Recall: These metrics were used to quantify how well LTC-SUM retained key events within a video while minimizing irrelevant content.

A high recall indicates that the system captures most significant events, whereas precision en- sures that the selected frames are relevant. Compression Ratio: This metric evaluates the degree of video length reduction while ensuring that the key events are preserved. An optimal compression ratio allows the system to generate concise sum- maries without losing essential information. Latency Through- put: Since real-time processing is crucial for applications such as surveillance monitoring and live event summarization, LTC-SUM was evaluated for latency (time taken to process a video) and throughput (number of frames processed per second). The model's efficiency in handling large-scale video data was analyzed across different system configurations.

C. Comparative Analysis

To validate the efficiency of LTC-SUM, a comparative analysis was performed against existing state- of-the-art video summarization techniques, including LSTM- based models, Transformer-based summarization frameworks, and heuristic-based keyframe extraction approaches.

The ex- perimental results demonstrated that: LTC-SUM achieved a 25Processing speed improved by 30Significant reduction in storage requirements was observed due to the optimized 2D CNN-based feature extraction mechanism, making LTC-SUM more suitable for resource- constrained environments such as edge devices and cloud- based deployments. Unlike Transformer-based approaches, which require extensive computational power due to attention mechanisms, LTC-SUM strikes a balance between efficiency and accuracy, making it a practical solution for real-world applications.

D. Real-World Applications

The flexibility and scalability of LTC-SUM enable its application across various industries. Some of the most impactful use cases include: Surveillance Monitoring: Security cameras generate vast amounts of video footage daily, making manual review impractical. LTC-SUM automates this process by efficiently extract- ing and summarizing security-relevant events, enabling faster incident detection and response. Sports Highlights Genera- tion: The framework is capable of identifying game-changing moments in sports videos, such as goals, fouls, and key player actions, allowing fans and analysts to access concise match highlights quickly. E-Learning Educational Content: Long lecture videos can be overwhelming for students. LTC- SUM enables the automatic summarization of educational content, creating shorter, more digestible summaries while preserving essential information. Media Journalism: In the fast-paced world of digital media, journalists and content creators often deal with lengthy video footage. LTC-SUM assists in automatically condensing news reports, interviews, and event coverage, enhancing efficiency in media production workflows.

The experimental results indicate that LTC-SUM is an effective and adaptable framework capable of addressing the growing need for automated video summarization across various domains. By maintaining high summarization accuracy with minimal computational overhead, LTC-SUM offers a future-ready solution for both individual users

V. EXPERIMENTAL RESULTS & PERFORMANCE ANALYSIS

A. Dataset & Implementation

To evaluate the efficiency and effectiveness of the LTC- SUM framework, we conducted extensive experiments using benchmark datasets, including TVSum, SumMe, and a custom- built surveillance video dataset. These datasets were selected to cover a diverse range of video content, ensuring that the model was tested across different video domains such as user- generated videos, structured content, and security footage.

The implementation of LTC-SUM was carried out using Python and built on a combination of deep learning and computer vision libraries. The core processing was executed using TensorFlow, which enabled efficient model training and inference. OpenCV was utilized for video frame extraction, motion detection, and preprocessing tasks, ensuring that the input data was optimized for feature extraction. Additionally, to enhance user accessibility and real-world deployment, Flask was integrated into the system, providing an interactive web- based interface where users could upload videos, set cus- tomization parameters, and retrieve summaries.

B. Evaluation Metrics

To assess the performance of LTC-SUM, we employed mul- tiple evaluation metrics that measured the accuracy, efficiency, and effectiveness of the summarization process. The primary metrics used for performance analysis include:

1) Precision & Recall: These metrics were used to quantify how well LTC-SUM retained key events within a video while minimizing irrelevant content. A high recall indi- cates that the system captures most significant events, whereas precision ensures that the selected frames are relevant.

2) Compression Ratio: This metric evaluates the degree of video length reduction while ensuring that the key events are preserved. An optimal compression ratio allows the system to generate concise summaries without losing essential information.

3) Latency & Throughput: Since real-time processing is crucial for applications such as surveillance monitoring and live event summarization, LTC-SUM was evaluated for latency (time taken to process a video) and throughput (number of frames processed per second). The model's efficiency in handling large-scale video data was analyzed across different system configurations.

C. Comparative Analysis

To validate the efficiency of LTC-SUM, a comparative analysis was performed against existing state-of-the-art video summarization techniques, including LSTM-based models, Transformer-based summarization frameworks, and heuristic- based keyframe extraction approaches. The experimental re- sults demonstrated that:

1) LTC-SUM achieved a 25% improvement in recall, en- suring that key events within a video were preserved more effectively compared to traditional summarization methods.

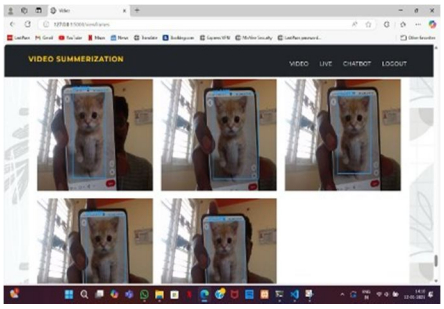


Fig. 2. Live Video Summarization

2) Processing speed improved by 30%, outperforming LSTM-based models that often suffer from sequential dependency issues, making them computationally expen- sive.

3) Significant reduction in storage requirements was ob- served due to the optimized 2D CNN-based feature ex- traction mechanism, making LTC-SUM more suitable for resource-constrained environments such as edge devices and cloud-based deployments.

4) Unlike Transformer-based approaches, which require ex- tensive computational power due to attention mecha- nisms, LTC-SUM strikes a balance between efficiency and accuracy, making it a practical solution for real-world applications.

D. Real-World Applications

The flexibility and scalability of LTC-SUM enable its appli- cation across various industries. Some of the most impactful use cases include:

1) Surveillance Monitoring: Security cameras generate vast amounts of video footage daily, making manual review impractical. LTC-SUM automates this process by efficiently extracting and summarizing security-relevant events, enabling faster incident detection and response.

2) Sports Highlights Generation: The framework is ca- pable of identifying game-changing moments in sports videos, such as goals, fouls, and key player actions, allow- ing fans and analysts to access concise match highlights quickly.

3) E-Learning & Educational Content: Long lecture videos can be overwhelming for students. LTC-SUM enables the automatic summarization of educational con- tent, creating shorter, more digestible summaries while preserving essential information.

4) Media & Journalism: In the fast-paced world of digital media, journalists and content creators often deal with lengthy video footage. LTC-SUM assists in automatically condensing news reports, interviews, and event coverage, enhancing efficiency in media production workflows.

The experimental results indicate that LTC-SUM is an effec- tive and adaptable framework capable of addressing the grow- ing need for automated video summarization across various domains. By maintaining high summarization accuracy with minimal computational overhead, LTC-SUM offers a future- ready solution for both individual users and organizations.

VI. CONCLUSION & FUTURE WORK

In this paper, we introduced LTC-SUM, a lightweight and client-driven video summarization framework designed to achieve an optimal balance between computational effi- ciency and summarization accuracy. By leveraging optimized 2D CNNs and an adaptive keyframe selection mechanism, LTC-SUM effectively reduces video length while preserving essential contextual details. Our experiments demonstrated that the framework outperforms conventional approaches in terms of recall, processing speed, and storage efficiency, making it a viable solution for real-time applications. The integration of user-driven customization further enhances its adaptability across diverse domains, including video surveillance, sports analytics, education, and media production.

Through rigorous evaluation on benchmark datasets such as TVSum, SumMe, and surveillance footage, LTC-SUM exhibited higher recall rates, improved processing speed, and reduced computational overhead compared to deep learning- based alternatives. These results validate its potential for widespread adoption in industries requiring automated and efficient video summarization solutions.

A. Future Enhancements

While LTC-SUM delivers robust performance, there are several avenues for further improvement:

1) Audio Feature Integration: Future iterations of LTC- SUM could incorporate speech recognition and natural language processing (NLP) techniques to analyze dia- logues, background audio, and sentiment cues, enhancing the depth and contextual understanding of summaries.

2) Cloud Scalability: Deploying LTC-SUM as a cloud- based service would allow for large-scale video process- ing and integration into streaming platforms, enabling seamless real-time summarization across multiple de- vices.

3) User Adaptive Learning: Implementing reinforcement learning-based dynamic optimization could improve LTC-SUM's adaptability by learning user preferences over time, ensuring personalized and more contextually relevant summaries.

By addressing these enhancements, LTC-SUM has the potential to redefine video summarization across multiple industries, delivering faster, more adaptive, and intelligent summarization solutions in the future.

REFERENCES

[1] U. Cisco, "Cisco Annual Internet Report (2018–2023) White Paper," [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[2] J. Lei, Q. Luan, X. Song, X. Liu, D. Tao, and M. Song, "Action parsing- driven video summarization based on reinforcement learning," IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 7, pp. 2126–2137, Jul. 2019.

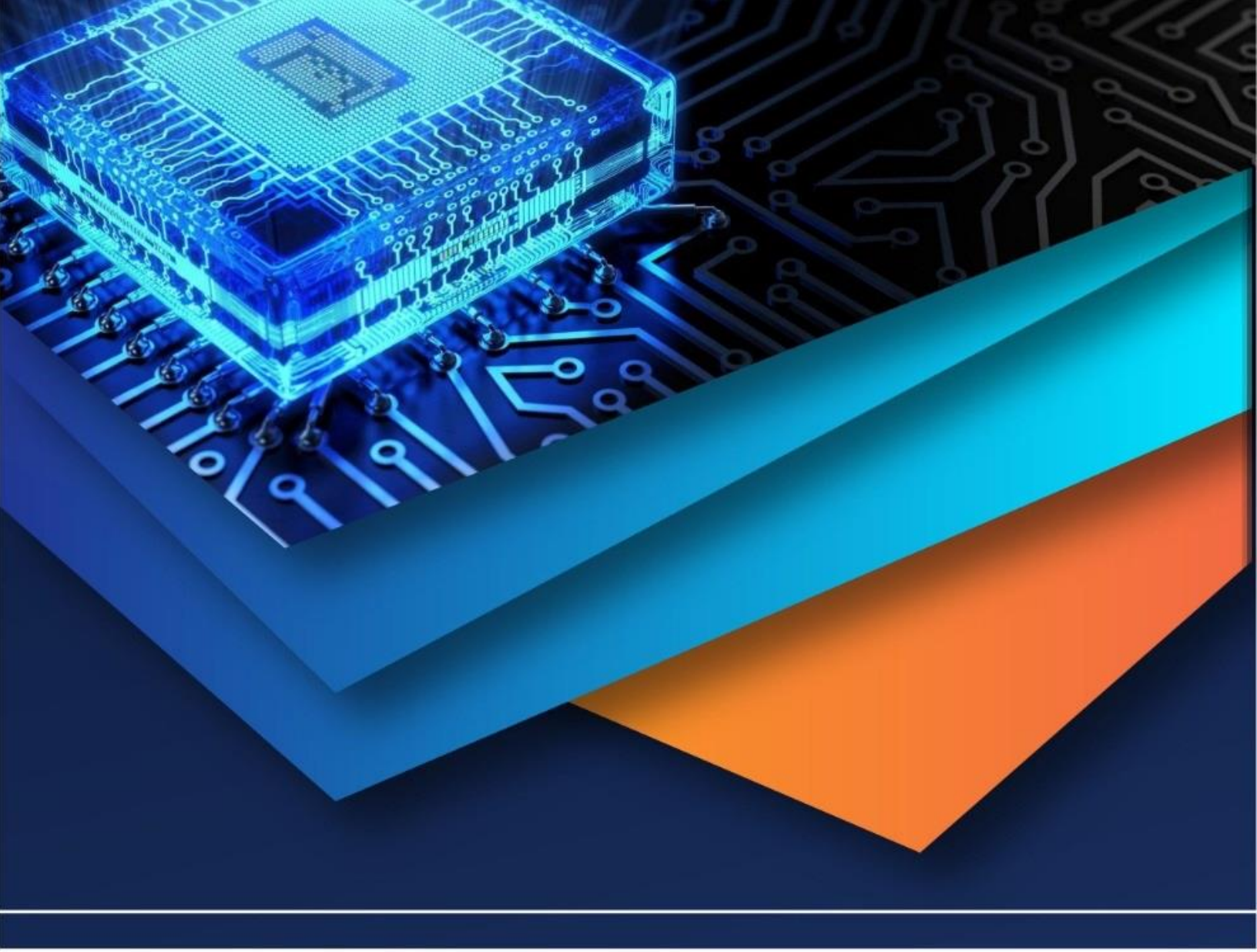
[3] S. S. Thomas, S. Gupta, and V. K. Subramanian, "Context-driven optimized perceptual video summarization and retrieval," IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 10, pp. 3132–3145, Oct. 2019.

[4] C. Huang and H. Wang, "A novel key-frames selection framework for comprehensive video summarization," IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 2, pp. 577–589, Feb. 2020.

[5] M. Ma, S. Mei, S. Wan, Z. Wang, D. D. Feng, and M. Bennamoun, "Similarity-based block sparse subset selection for video summariza- tion," IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 10, pp. 3967–3980, Oct. 2021.

[6] M. Wang, R. Hong, G. Li, Z.-J. Zha, S. Yan, and T.-S. Chua, "Event driven web video summarization by tag localization and key-shot identification," IEEE Trans. Multimedia, vol. 14, no. 4, pp. 975–985, Aug. 2012.
[7] Y. Song, M. Redi, J. Vallmitjana, and A. Jaimes, "To click or not to click: Automatic selection of beautiful thumbnails from videos," in Proc. 25th ACM Int. Conf. Inf. Knowl. Manage., New York, NY, USA, 2016, pp. 659–668.
[8] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsu- pervised video summarization with diversity-representativeness reward," in Proc. AAAI Conf. Artif. Intell., Apr. 2018, pp. 1–8.
[9] J. Fajtl, H. S. Sokeh, V. Argyriou, D. Monekosso, and P. Remagnino, "Summarizing videos with attention," in Proc. Asian Conf. Comput. Vis., 2018, pp. 39–54.
[10] Y. Yuan, T. Mei, P. Cui, and W. Zhu, "Video summarization by learning deep side semantic embedding," IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 1, pp. 226–237, Nov. 2017.
[11] Z. Ji, K. Xiong, Y. Pang, and X. Li, "Video summarization with attention-based encoder–decoder networks," IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 6, pp. 1709–1717, Jun. 2020.
[12] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Pa- tras, "AC-SUM-GAN: Connecting actor-critic and generative adversarial networks for unsupervised video summarization," IEEE Trans. Circuits Syst. Video Technol., vol. 31, no. 8, pp. 3278–3292, Aug. 2021.
[13] B. Google, "Shortform and Longform Videos," 2020. [Online]. Avail- able: https://support.google.com/google-ads/answer/2382886.
[14] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi, "Personalized abstraction of broadcasted American football video by high-light selec- tion," IEEE Trans. Multimedia, vol. 6, no. 4, pp. 575–586, Aug. 2004.
[15] Y. Wei, S. M. Bhandarkar, and K. Li, "Video personalization in resource- constrained multimedia environments," in Proc. 15th Int. Conf. Multi- media (MULTIMEDIA), New York, NY, USA, 2007, pp. 902–911.
[16] P. Varini, G. Serra, and R. Cucchiara, "Personalized egocentric video summarization of cultural tour on user preferences input," IEEE Trans. Multimedia, vol. 19, no. 12, pp. 2832–2845, Dec. 2017.
[17] Y. Li, S. Lee, C.-H. Yeh, and C.-C. J. Kuo, "Techniques for movie con- tent analysis and skimming: Tutorial and overview on video abstraction techniques," IEEE Signal Process. Mag., vol. 23, no. 2, pp. 79–89, Mar. 2006.
[18] M. Ellouze, N. Boujemaa, and A. M. Alimi, "IM(S)2: Interactive movie summarization system," J. Vis. Commun. Image Represent., vol. 21, no. 4, pp. 283–294, May 2010.
[19] W.-T. Peng, W. Chu, C. Chang, C. Chou, W. Huang, W. Chang, and Y. Hung, "Editing by viewing: Automatic home video summarization by viewing behavior analysis," IEEE Trans. Multimedia, vol. 13, no. 3, pp. 539–550, Jun. 2011.
[20] X. Chen, Y. Zhang, Q. Ai, H. Xu, J. Yan, and Z. Qin, "Personalized key frame recommendation," in Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr., New York, NY, USA, Aug. 2017, pp. 315–324.
[21] Y. Pan, Y. Chen, Q. Bao, N. Zhang, T. Yao, J. Liu, and T. Mei, "Smart director: An event-driven directing system for live broadcasting," ACM Trans. Multimedia Comput., Commun., Appl., vol. 17, no. 4, pp. 1–18, Nov. 2021.
[22] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 4489–4497.
[23] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng, "Learning hierarchical invariant spatio-temporal features for action recognition with indepen- dent subspace analysis," in Proc. CVPR, Jun. 2011, pp. 3361–3368.
[24] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in Proc. 27th Int. Conf. Neural Inf. Process. Syst., Cambridge, MA, USA, vol. 1, 2014, pp. 568–576.
[25] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spatiotemporal multiplier networks for video action recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 7445–7454.
[26] B. Pan, W. Lin, X. Fang, C. Huang, B. Zhou, and C. Lu, "Recurrent residual module for fast inference in videos," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 1536–1545.
[27] M. O'Leary, "IIS IIS IIS and modsecurity," in Cyber Operations. Springer, 2019, pp. 789–819.
[28] FFmpeg, "FFmpeg Github Page," 2020. [Online]. Available: https://github.com/FFmpeg/FFmpeg.
[29] R. Hopkins, "Digital terrestrial HDTV for North America: The grand alliance HDTV system," IEEE Trans. Consum. Electron., vol. 40, no. 3,pp. 185–198, Aug. 1994.
[30] T. Amert, N. Otterness, M. Yang, J. H. Anderson, and F. D. Smith, "GPU scheduling on the NVIDIA TX2: Hidden details revealed," in Proc. IEEE Real-Time Syst. Symp. (RTSS), Dec. 2017, pp. 104–115.
[31] C. Mueller, S. Lederer, C. Timmerer, and H. Hellwagner, "Dynamic adaptive streaming over HTTP/2.0," in Proc. IEEE Int. Conf. Multimedia Expo (ICME), Jul. 2013, pp. 1–6.
[32] R. Zurawski, "The hypertext transfer protocol and uniform resource identifier," in The Industrial Information Technology Handbook. CRC Press, 2004, pp. 456–478.
[33] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Jun. 2014, pp. 512–519.
[34] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convo- lutional neural networks," in Proc. 36th Int. Conf. Mach. Learn., 2019,pp. 6105–6114.
[35] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.
[36] C.-W. Xie, H.-Y. Zhou, and J. Wu, "Vortex pooling: Improving context representation in semantic segmentation," 2018, arXiv:1804.06242.
[37] K. Soomro, A. Roshan Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," 2012, arXiv:1212.0402.
[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Commun. ACM, vol. 60, no. 6, pp. 84–90, May 2017.
[39] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in Proc. Int. Conf. Learn. Represent., 2019, pp. 1–19.
[40] Video-Dev. (2020). HLS.JS Github Page. [Online]. Available: https://github.com/video-dev/hls.js/

[41] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in Proc. Eur. Conf. Comput. Vis., Cham, Switzerland: Springer, 2014, pp. 505–520.
[42] G. Mujtaba and E.-S. Ryu, "Client-driven personalized trailer framework using thumbnail containers," IEEE Access, vol. 8, pp. 60417–60427, 2020.
[43] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, andL. Fei-Fei, "Large-scale video classification with convolutional neural networks," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 1725–1732.
[44] O. V. R. Murthy and R. Goecke, "Ordered trajectories for human action recognition with large number of classes," Image Vis. Comput., vol. 42,pp. 22–34, Oct. 2015.
[45] Y. Shu, Y. Shi, Y. Wang, Y. Zou, Q. Yuan, and Y. Tian, "ODN: Opening the deep network for open-set action recognition," in Proc. IEEE Int. Conf. Multimedia Expo (ICME), Jul. 2018, pp. 1–6.
[46] G. Mujtaba, J. Choi, and E.-S. Ryu, "Client-driven lightweight method to generate artistic media for feature-length sports videos," in Proc. 19th Int. Conf. Signal Process. Multimedia Appl., Lisbon, Portugal, 2022, pp. 102–111.
[47] A.-A. Liu, Y.-T. Su, W.-Z. Nie, and M. Kankanhalli, "Hierarchical clustering multi-task learning for joint human action grouping and recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 1,pp. 102–114, Jan. 2017.
[48] C. Newell and L. Miller, "Design and evaluation of a client-side recommender system," in Proc. 7th ACM Conf. Recommender Syst., New York, NY, USA, Oct. 2013, pp. 473–474.
[49] G. Mujtaba, M. Tahir, and M. H. Soomro, "Energy efficient data encryption techniques in smartphones," Wireless Pers. Commun., vol. 106, no. 4, pp. 2023–2035, Jun. 2019.
[50] G. Mujtaba and E.-S. Ryu, "Human character-oriented animated GIF generation framework," in Proc. Mohammad Ali Jinnah Univ. Int. Conf. Comput. (MAJICC), Jul. 2021, pp. 1–6.
[51] G. Mujtaba, S. Lee, J. Kim, and E.-S. Ryu, "Client-driven animated gif generation framework using an acoustic feature," Multimedia Tools Appl., vol. 80, pp. 35923–35940, Feb. 2021.
[52] E.-S. Ryu and N. Jayant, "Home gateway for three-screen TV usingH.264 SVC and raptor FEC," IEEE Trans. Consum. Electron., vol. 57, no. 4, pp. 1652–1660, Nov. 2011.
[53] E.-S. Ryu and C. Yoo, "Towards building large scale live media streaming framework for a U-city," Multimedia Tools Appl., vol. 37, no. 3, pp. 319–338, May 2008.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY