Spatiotemporal Feature Fusion for Video Summarization

Shamal Kashid and Lalit K. Awasthi, *National Institute of Technology Uttarakhand, Srinagar, 246174, India*

Krishan Berwal, *Military College of Telecommunication Engineering, Mhow, 453441, India*

Parul Saini, *Dehradun Institute of Technology University, Dehradun, 248009, India*

Video summarization (VS) is crucial process for compacting video content into a concise and informative representation, enhancing accessibility and the user experience. This work introduces a new approach based on spatiotemporal features derived from long short-term memory and pretrained convolutional neural network (CNN) models for static VS. It utilizes dual-CNN to identify keyframes by extracting features from benchmark datasets that contain user-generated summaries as the ground truth. Additionally, the incorporation of self-organizing map clustering into the dual-CNN model is investigated for superior performance compared to alternative clustering strategies. This spatiotemporal-based VS method effectively selects the most representative frames from the extracted spatiotemporal features. Unlike traditional methods, it does not require training on specific VS datasets, eliminating the need for extensive labeled data. Compared to existing state-of-the-art techniques in the literature, the proposed approach demonstrates promising results, consistently generating high-quality video summaries across various content categories. It achieved average F-scores of 84.7%, 86.4%, 61.9%, and 53.6% on four benchmark Open Video, YouTube, TVSum, and SumMe datasets, respectively, showing its effectiveness in producing informative video summaries.

The rapid growth of video data necessitates efficient video summarization (VS) methods that enable users to browse and comprehend large amounts of video content quickly. VS provides a brief overview of the video's content, but users must spend time and effort searching through the vast amount of information. Images and videos have become popular multimedia formats due to their direct link to users. The Internet is overflowing with video data, accelerated by the widespread availability of high-speed Internet and storage options. Platforms like YouTube generate more than 10 hours of video every second.[1] Video consumption is higher than text and images and requires significant human resources.

To address this, efficient methods and technologies for VS, acquisition, compression, and presentation are crucial.[1] VS has been studied since the 1990s and can be categorized into static and dynamic VS. Static VS uses a storyboard with keyframes to represent video content, skipping audio messages. Dynamic VS incorporates image, text, and audio information. Static VS is more straightforward to browse and reduces computational complexity in video retrieval and analysis.[1]

These VS approaches leverage spatial and temporal clues to reduce long videos into short representations, improving rapid browsing, retrieval, and understanding of visual content. Spatial features extract the visual characteristics of individual frames, while temporal features emphasize motion dynamics,

1070-986X © 2024 IEEE
Digital Object Identifier 10.1109/MMUL.2024.3428933
Date of publication 17 July 2024; date of current version 23 September 2024.

scene transitions, and narrative advancement. Convolutional neural networks (CNNs) are critical in VS because they extract detailed spatial characteristics from each frame, improving a model's ability to distinguish between elements. Recurrent neural networks, specifically long short-term memory (LSTM) networks,[4] have transformed the representation of time-based patterns in video data by effectively capturing long-range dependencies and temporal interactions, resulting in more contextually aware video summaries.[1]

This study uses deep neural networks to enhance video summarizing by utilizing spatiotemporal features. It employs pretrained CNNs—ResNet-50[1] and GoogLeNet,[1] modified as M-ResGoogleNet—to extract detailed spatial representations from each video frame. The temporal patterns within sequences are also considered using LSTMs. Self-organizing map (SOM) clustering groups feature vectors into coherent clusters based on similarity. From each cluster, the representative keyframe is selected based on the Euclidean distance. This method efficiently compacts significant video content into a short static summary, preserving crucial visual information. The collection of frames corresponding to the input video is processed to reduce the number of redundant frames, as shown in Figure 1. The problem of VS is formulated as a combination of feature extraction and clustering tasks, where the objective is to select keyframes that effectively represent the content of the video.

This work introduces an innovative approach to spatiotemporal-based VS (STVS) (static VS), aiming to create summaries that are not dependent on the specific category of the video. The study aims to overcome some limits of the current methods by proposing a methodology that leverages features extracted from a modified pretrained dual-CNN and LSTM model

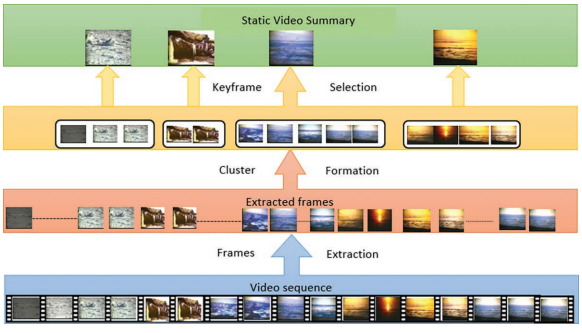

FIGURE 1. Basic workflow of VS.

to accurately identify the essential elements within a video. The primary contributions of this research are outlined as follows:

- An innovative method has been developed for creating precise video abstracts across various categories.
- The integration of deep-learning-based spatiotemporal feature extraction with the clustering approach summarization represents a significant advancement in VS.
- Extracting spatial and temporal deep features through the utilization of the dual-CNN (modified versions of GoogLeNet and ResNet-50) and LSTM models.
- A novel method for summarizing videos by combining the proposed feature extraction model with the Kohonen SOM.
- Overall, the proposed approach significantly advances VS by integrating spatiotemporal features, using deep learning for feature extraction with SOM clustering, and employing an efficient summarization methodology.

BACKGROUND STUDY

VS techniques aim to extract essential keyframes from videos. Various approaches have been employed, including feature-based methods, like color, motion, objects, gesture, speech, audiovisual cues, etc. Clustering-based techniques, such as *k*-means clustering, partitioned clustering, and spectral clustering, have also been utilized. Other approaches include shot-selection-based VS, event-based summarization, and trajectory-based VS.[1] Deep learning techniques have become effective in extracting informative features from video content, especially by integrating spatiotemporal representations. Mahesseni et al.[2] proposed a novel adversarial LSTM network, which addresses the model's limited coding capability but can cause model failure due to unstable training. Zhou et al.[3] presented an end-to-end LSTM-based unsupervised learning method for VS, combining diversity and representativeness with a novel feedback reward. The SUM-GDA[3] is a novel VS model leveraging a diverse global attention mechanism to model pairwise temporal relations and determine keyframes. The model employs determinantal point processes to choose an optimal subset of keyframes, enhancing diversity in the selected frames. The extension of SUM-GDA to an unsupervised scenario provides a cost-effective solution, eliminating the

need for human labeling and illustrating for practical applications.

The GL-RPE,[5] a method combining global and local input decomposition with relative position embedding, enhances the network's efficiency in capturing both local and global interdependencies for improved unsupervised VS. This approach is adaptable to different unsupervised methods. The 3DST-UNet-RL[6] framework for VS uses a 3D spatiotemporal U-Net and reinforcement learning to identify crucial video content and remove redundancy. The method demonstrates effectiveness in both unsupervised and supervised modes, with applications in general VS benchmarks and medical video tasks.

Considering clustering approaches,[14] equal-frames-partition-based VS utilizes an equal-partition-based clustering technique to cluster the entire video into keyframe groups. The SUM-GANdpp[2] is an unsupervised method for VS that uses a discrete similarity measure learned through the generative adversarial network discriminator. ERA[16] and VSS-Net,[17] two existing approaches to VS, prioritize modeling spatial and temporal features differently. While ERA emphasizes object detection for spatial feature modeling and constructs relationships among entities over time, VSS-Net focuses on temporal feature modeling using contextual graphs. However, the proposed STVS approach shows its functionality by utilizing trained features from LSTM and CNN models. STVS offers a distinct approach to capturing both spatial and temporal dynamics, showcasing promising results on benchmark datasets in comparison to existing methods.

Furthermore, some methods[18,19,20] may generate summaries that lack contextual understanding or capture the essence of the content. An efficient, secure technique for the keyframes-based VS model (ESKVS) proposes efficient and secure keyframe selection algorithms based on the probability of a frame being a keyframe and its contained information. The secret sharing strategy enhances the security of secret keyframes using blockwise and polynomial congruence encryption techniques.[14]

The discussed approaches face challenges like information overload, accessibility, and user experience while leveraging advancements in deep learning to improve the efficiency and quality of VS and redundancy in video summary. Overall, the motivation behind the proposed approach lies in its ability to address these challenges by offering concise and informative summaries. The approach aims to enhance the utility of video content for users across various domains. It aims to make things work better and be more useful by utilizing established evaluation metrics on benchmark datasets, adding spatiotemporal features from deep neural networks, and making the summarization process easier by using effective feature extraction and clustering algorithms.

PROPOSED MODEL

The proposed technique for creating static summaries for input videos is based on the idea that users will only find a summary engaging if it accurately captures the main points of the original video. A modified pretrained CNN processes the generated set to extract the deep features. Frame representations are built using the layer activations of a fully connected layer of pretrained models, and LSTM modules extract temporal features. Combining features extracted from both models improves the efficacy of feature vectors. Figure 2 displays the proposed STVS architecture.

Extraction of Frames

The videos consist of a series of images referred to as frames. Generally, the frames per second vary depending on the type of video. In this case, the video comprises 30 frames per second. We first converted these videos into frames and then input them into the pretrained network. This approach neither involves presampling to preserve the video's originality nor converts colored frames into gray-scale ones.

Extraction of Features

Deep features are derived from individual frames within the set of frames. In VS, we accomplish the feature extraction task using two scenarios: for spatial features, ResGoogleNet (GoogLeNet + Resnet50) and M-ResGoogleNet (modified GoogLeNet + Resnet50) are used, and the LSTM module is used for temporal features, which are explained in the "Spatial Features" and "Temporal Features" sections, respectively.

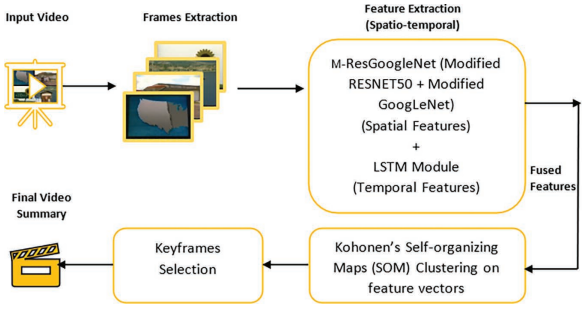

FIGURE 2. Proposed model architecture for static VS.

Spatial Features

The spatial features from input frames are extracted by using dual-CNN approaches: ResGoogleNet and M-ResGoogleNet. The details are given as follows:

- *ResGoogleNet*: This refers to a fusion approach where features extracted from both the ResNet-50 and GoogLeNet architectures are combined to create more robust features from frames. Feature fusion combines the features extracted from both networks. The feature vectors F1 (2048-D) and F2 (1024-D) extracted using ResNet-50 and GoogLeNet are concatenated to form a final vector F (3072-D). These fused features can capture broader information from the input frames.
- *M-ResGoogleNet*: In the proposed modified ResGoogleNet, four new dense layers with 1024, 512, 256, and 128 hidden nodes using dropout value ranging from 0.2 to 0.4 are added to the top layer of both networks (ResNet-50 and GoogLeNet), and, also, a dense layer having 80 hidden nodes with a softmax activation function is added to predict the final output, as shown in Figure 3.

Temporal Features

LSTM is employed to analyze sequential information and consider time series for extracting the temporal features, as shown in Figure 4. The network has gates to regulate information flow and determine the importance of data. The forget gate uses the sigmoid function to discard or label data. The input gate computes the cell state, while the output gate identifies the hidden state for predictions. The LSTM model architecture includes a sequence input layer, seven LSTM layers, and a classification output layer, as shown in Figure 4. The public dataset Common Objects in Context (COCO) is used for training the proposed M-ResGoogleNet and LSTM models together for feature extraction of the video frames with a split ratio of 70% training, 15% testing, and 15% validation. It consists of 330,000 images, each annotated with 80 object categories and five captions describing the scene for object detection, segmentation, and captioning tasks.

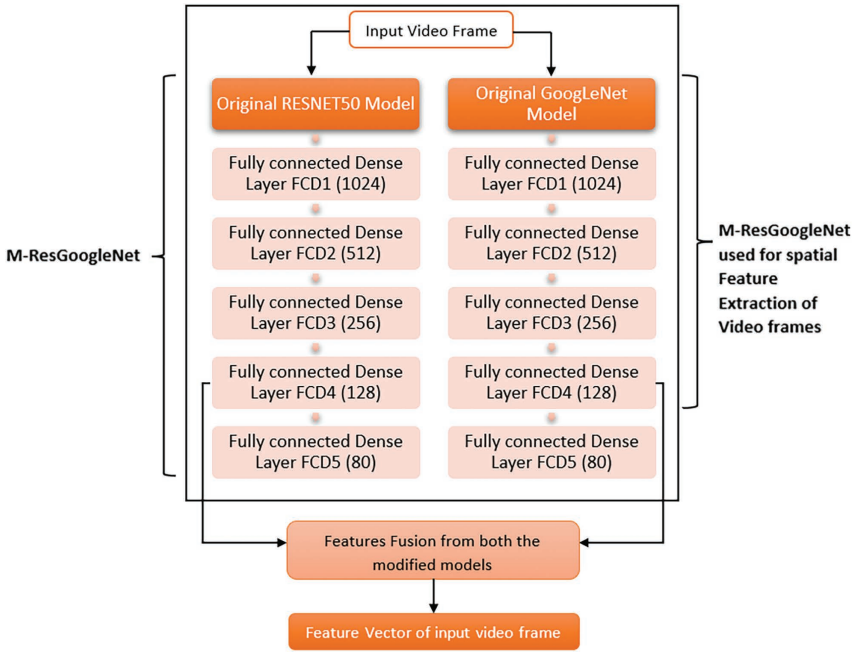

FIGURE 3. M-ResGoogleNet architecture.

FIGURE 4. LSTM module.

We perform rotation and scaling for data augmentation. Rotation ranges from –45° to +45°, and scaling ranges from 0.8 to 1.2. We use the transfer learning approach for feature extraction and fine-tuning after training. The original ResNet-50 and GoogLeNet architectures load the pretrained weights from the Imagenet dataset. We set the learning rate parameter to 0.001. The proposed framework trains and tests image classification using the proposed M-ResGoogleNet, which freezes the original ResNet-50 and GoogLeNet model layers. For VS, features are extracted from the last layer of the modified model, excluding the top classification layer. Adapting more layers of transfer learning enhances the accuracy of the model's predictions for the dataset. Finally, the features are fused for clustering.

SOM Clustering

The Kohonen SOM is an unsupervised learning neural network. Many conventional techniques are used for clustering, but most require prior knowledge of the data distribution, whereas the Kohonen SOM does not require this.[7] The Kohonen SOM is utilized to reduce dimensionality. Regarding frame clustering, the data are high dimensional, and we need to know how pixels are distributed among frames. A study by Arai[8] found that SOM-based clustering outperformed *k*-means clustering in terms of cluster separability, improving by 16%. SOM is more advanced than previous clustering approaches in terms of presentation because it displays the relationships between the clusters in a 2D space and groups the data points into clusters.

The extracted features are input to the Kohonen SOM clustering approach. Apply the batch SOM algorithm to train the SOM model. This process involves adjusting the model's weights to learn the topology and distribution of the input data. The training continues for a fixed number of epochs, refining the SOM's representation. After training, identify and retrieve clusters of frames that exhibit similar patterns on the SOM. The learning rate for SOM clustering is fixed at 0.01. These clusters represent groups of frames with similar characteristics, implying visual similarity.

Algorithm 1 proposes a method for clustering frames from a video using SOM clustering. Initially, we extract deep-level features from the frames, capturing their essential characteristics. Then, these features are fed into the SOM model. The model is trained using the batch SOM algorithm to learn the structure and distribution of the data over a certain number of iterations. As a result, the SOM organizes itself so that neighboring nodes represent similar features, effectively clustering the frames. This approach can be highly effective in identifying and grouping visually similar frames, which is valuable for VS.

Detection of Keyframes

Algorithm 2 shows the method for selecting representative frames or keyframes from clusters obtained through the SOM clustering process. The purpose of this algorithm is to identify frames that are significantly different from the cluster's centroid frame. The algorithm operates iteratively over each cluster, first calculating the centroid frame by averaging the features of all frames within the cluster. Subsequently, it measures the dissimilarity between each frame and the centroid, selecting the frame with the maximum dissimilarity to the representative frame. By selecting frames with the maximum dissimilarity, the algorithm aims to ensure that the representative frames are distinct from other frames within the same cluster and, thus, provide a meaningful summary of the cluster's content.

PERFORMANCE ANALYSIS

This section introduces the VS evaluation technique to analyze the performance of the proposed framework.

Algorithm 1. Clustering Using the Kohonen SOM.

Input: $F = (f_1, f_2, f_3, f_4, \ldots. f_n)$, the collection of n frames of the input video.
Output: Frame clusters k that share similarities.
Step 1: Extract deep-level features from frames using scenarios as explained in the "Extraction of Features" section.
Step 2: Input the extracted feature vector to the Kohonen SOM model.
Step 3: Utilize the batch SOM algorithm to train the model, allowing it to learn the input's topology and distribution over a fixed number of epochs.
Step 4: Return a cluster comprising frames that share similarities.

Algorithm 2. Representative Frames Selection.

Input: k clusters formed by SOM clustering
Output: Representative frames
For $i = 1$ to k
Step 1: For each frame in a cluster, calculate the centroid frame.
Step 2: Calculate the dissimilarity between each frame and the centroid frame.
Step 3: Select the frame with maximum dissimilarity to the representative frame.

VS Evaluation Technique

In video summary analysis, the summary of the videos is examined depending on individual interests and opinions. For qualitative analysis, keyframes are analyzed visually and compared with the ground-truth summaries. Ground-truth summaries are the ideal expected summary provided by humans or users. Three performance evaluation metrics are utilized for quantitative analysis: the precision (P), recall (R), and F-measure metrics are described as follows. The marked reference summary is used as a frame unit to estimate the R, P, and F-measure,[1] which is referred to as the harmonic mean of R and P and calculated as

$$\text{F-Measure} = \frac{2 \times P \times R}{(P + R)}. \quad (1)$$

The scores are calculated by either taking the average or choosing the maximum value. Therefore, there are two possible ways to aggregate the F-measure scores: one is to compute an average of F-measure over all reference summaries, and the other is to use the maximum score.[9]

EXPERIMENTAL RESULTS AND DISCUSSION

This section assesses the proposed model's performance using the performance analysis detailed in the previous section. Four datasets are employed to measure the effectiveness of the proposed VS models. The comprehensive description of the datasets is provided in section the "Datasets for VS" section. The proposed VS is evaluated through quantitative and qualitative analysis, as explained in the "Quantitative Analysis" and "Qualitative Analysis" sections, respectively. All experiments were conducted in Google Colab using a T4 GPU running at 51 GB of random-access memory, an Intel Xeon 2.20 GHz processor, and 166.8 GB of disk storage. Furthermore, an ablation study for dual-CNN feature extraction selection is presented in the "Ablation Study" section.

Datasets for VS

In this section, we evaluate our algorithm on four diverse datasets: Open Video (OV),[14] YouTube (YT),[14] TVSum,[1] and SumMe,[1] as shown in Table 1. The OV dataset includes 50 videos. The video frame size is 352×240 pixels. Every video in MPEG-1 forms in color and has sound. The total size of the OV dataset is 763 MB. These videos include various genres (documentary, lecture instructive, ephemeral, and historical), span from 1 to 4 min, and total around 75 min. The YT dataset comprises 50 videos, and every video in MPEG-1 in this dataset is also in color with sound. The total size of the YT dataset is 468 MB. These video elements are shared into several categories, including cartoons, sports, news, marketing, a series collection for TV, and homemade videos. The SumMe dataset comprises 25 videos captured from first-person or third-person perspectives, varying from 1 to 6 min. TVSum, a dataset consisting of 50 videos sourced from YouTube, offers diverse content categorized into 10 distinct categories.

Quantitative Analysis

In the VS dataset scenario (OV and YT), keyframes extracted from videos undergo comparison with ground-truth summaries generated by users. Performance evaluation of the proposed methods employs the F-measure metric. Table 2 displays the results obtained on a per-category basis, illustrating that the dual-CNN-based feature extraction VS method achieves commendable F-scores across various categories. To evaluate the effectiveness of the proposed VS model, we conduct a comparative analysis of STVS-I and STVS-II results with other keyframe-selection-based VS techniques. Table 3 presents the evaluation and comparison of the proposed STVS approach with existing methodologies on the OV, YT, TVSum, and SumMe datasets. The F-measure metric demonstrates the superior performance of STVS over previous approaches. Table 4 represents the F-measure for the SumMe and TVSum datasets, where the average denotes the

TABLE 1. Description of the VS datasets.

Datasets	Number of videos	Number of annotations	Categories	Duration (Min)
OV[1]	50	5	Documentary, educational ephemeral, historical, and lecture	1–4
UT[1]	50	5	Cartoons, sports, news, home, and user-generated content	1–4
TVSum[9]	50	20	News, documentaries, and user-generated content	1–11
SumMe[9]	25	15–18	Holidays, events, home, and user-generated content	1–6

TABLE 2. VS analysis (%) of the YT dataset according to category.

Category	Model	Precision	Recall	F-measure
Cartoons	GoogLeNet	75.5	77.8	76.6
	ResNet50	78.0	82.7	80.3
	MultiCNN[13]	80.0	83.0	81.0
	Proposed STVS-I (ResGoogleNet)	79.8	84.2	81.9
	Proposed STVS-II (M-ResGoogleNet)	**81.9**	**86.3**	**84.0**
Sports	GoogLeNet	81.2	77.4	79.2
	ResNet50	83.6	85.2	84.4
	MultiCNN[13]	85.0	89.0	87.0
	Proposed STVS-I (ResGoogleNet)	85.2	88.7	86.9
	Proposed STVS-II (M-ResGoogleNet)	**86.3**	**90.2**	**88.2**
News	GoogLeNet	77.3	79.4	78.3
	ResNet50	79.4	80.1	79.7
	MultiCNN[13]	78.0	81.0	79.0
	Proposed STVS-I (ResGoogleNet)	81.2	82.6	81.8
	Proposed STVS-II (M-ResGoogleNet)	**82.7**	**86.7**	**84.7**

Best results are shown in bold.

TABLE 3. Comparative analysis F-measure (%) of different approaches on the OV, YT, TVSum, and SumMe datasets.

Model	OV	YT	TVSum	SumMe
SUM-GANdpp[2]	72.0	62.1	51.7	39.1
MultiCNN[13]	82.0	83.0	—	—
ESKVS[14]	76.0	81.2	—	—
SUM-GDAunsup[3]	—	—	59.6	50.0
CSNet+GL+RPE[5]	—	—	59.1	50.2
Park et al.[15]	—	—	59.3	49.8
Liu et al.[6]	—	—	58.1	44.6
Proposed STVS-I ResGoogleNet	81.5	84.5	60.8	51.2
Proposed STVS-II (M-ResGoogleNet)	**84.7**	**86.4**	**61.9**	**53.6**

Best results are shown in bold.

TABLE 4. F-measure for SumMe and TVSum datasets (F-Avg and F-Max).*

	TVSum		SumMe	
Model	**F-Avg**	**F-Max**	**F-Avg**	**F-Max**
VS-LMM[10]	—	—	—	0.40
dppLSTM[11]	0.60	—	—	0.43
SASUM[12]	0.58	—	—	0.45
Proposed STVS-I ResGoogleNet	0.60	0.77	0.51	0.66
Proposed STVS-II M-ResGoogleNet	**0.61**	**0.79**	**0.53**	**0.68**

*F-Avg: the average of F-measure scores over all ground-truth summaries. F-Max: the highest F-measure score within the reference summaries.

average of F-measure scores over all ground-truth summaries, and the maximum denotes the highest F-measure score within the reference summaries.[10]

Qualitative Analysis

Ground-truth summaries generated by five users for the 50 videos in the OV and YT dataset are available in the dataset itself. The proposed STVS method is compared with user-generated summaries. For our experiment, the result of the sample video in comparison with the other algorithms is shown in Figure 5. The video frames are first resampled to eliminate the meaningless frames in the Video SUMMarization (VSUMM) technique, and then SOM clustering is done based on the color histograms of the remaining frames. VSUMM1 and VSUMM2 also skip some keyframes compared to the ground-truth summaries of five users. Most of these approaches failed to extract the second and last keyframes compared to the ground-truth summaries of five users available from the VSUMM dataset. Unlike the other approaches, our proposed method, STVS-I (ResGoogleNet) and STVS-II (M-ResGoogleNet), selects the optimum number of keyframes compared to other approaches, and its summary is very close to the ground-truth summaries.

Ablation Study

We have conducted ablation studies on the video datasets, as in the "Datasets for VS" section, to investigate

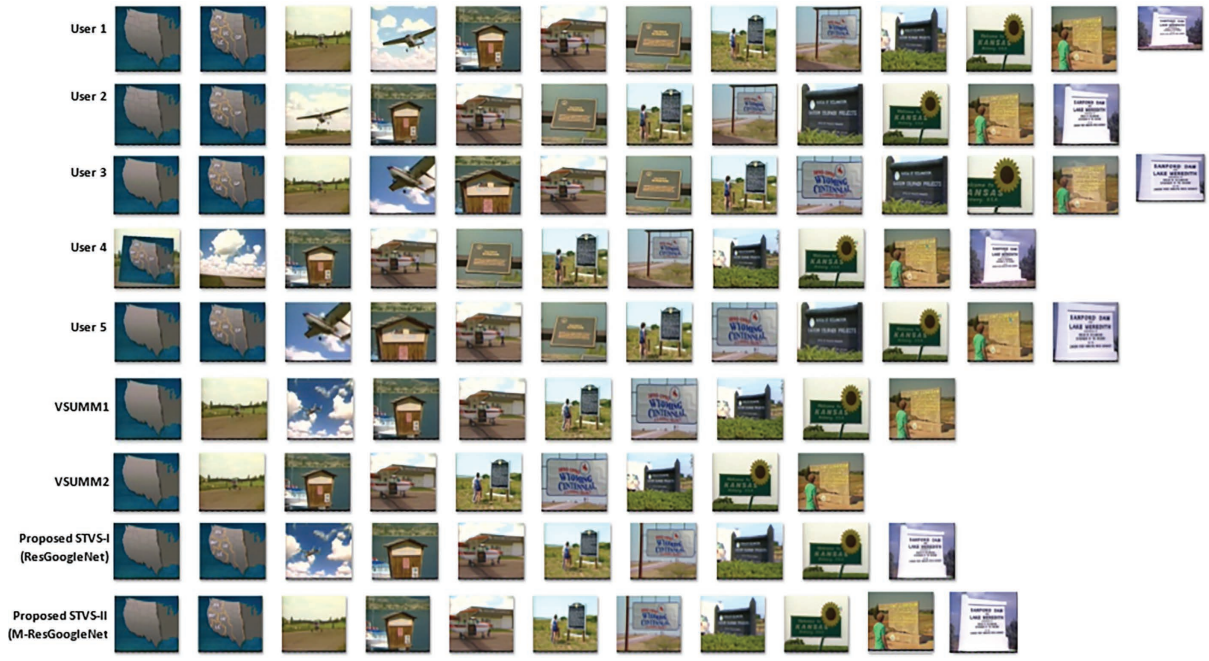

FIGURE 5. Sample video results of the proposed STVS-I and STVS-II in comparison with five user ground-truth summaries, VSUMM1, and VSUMM2 algorithms on the OV dataset.

TABLE 5. Ablation study: Comparing the proposed STVS on F-measure (%) with baselines VGG16, GoogLeNet, and ResNet-50.*

Model	OV	YT	TVSum	SumMe
VGG-16	72	71	50	42
GoogLeNet	77	77	52	46
ResNet-50	82	81	56	47
VGG-16 + GoogLeNet	80	79	54	48
VGG-16 + ResNet-50	81	80	55	49
ResGoogleNet (ours)	81	84	60	51
VGG-16 + GoogLeNet + ResNet-50	80	79	55	50
M-ResGoogleNet (ours)	**84**	**86**	**61**	**53**

*The best results are shown in bold.

1) the effectiveness of the proposed STVS-I (ResGoogleNet) and STVS-II (M-ResGoogleNet) based on feature extraction approaches and 2) the performance of the proposed algorithms compared to other related approaches. The ablation study makes use of three pretrained CNN models–namely, VGG-16, GoogLeNet, and ResNet-50[1]–and their combinations for the fusion of the features, as shown in Table 5. GoogleNet and ResNet-50 are selected for the fusion of features, as they give better performance as compared to other combinations.

CONCLUSION

This work focused on static summarization methods for videos due to the rapid growth of vast video collections based on spatiotemporal features derived from LSTM and pretrained CNN models. According to experimental analysis of the TVSum, SumMe, OV, and YT datasets, it effectively recognizes keyframes by two distinct feature extraction deep learning scenarios with SOM clustering. SOM is employed to cluster frames following fusion, and the most representative frames within each cluster are chosen based on centroid calculations. Utilizing different combinations of additional pretrained CNN models can further enhance the results. Future research will focus on identifying the optimal number and combination of pretrained CNN models to enhance the overall quality of video summaries.

REFERENCES

1. P. Saini, K. Kumar, S. Kashid, A. Saini, and A. Negi, "Video summarization using deep learning techniques: A detailed analysis and investigation," *Artif. Intell. Rev.*, vol. 56, no. 11, pp. 12,347–12,385, 2023, doi: 10.1007/s10462-023-10444-0.

2. B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial LSTM networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 2982–2991, doi: 10.1109/CVPR.2017.318.
3. K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 7582–7589, doi: 10.1609/aaai.v32i1.12255.
4. K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proc. Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer-Verlag, Oct. 2016, pp. 766–782.
5. Y. Jung, D. Cho, S. Woo, and S. Kweon, "Global-and-local relative position embedding for unsupervised video summarization," in *Proc. Eur. Conf. Comput. Vis.*, Cham, Switzerland: Springer-Verlag, 2020, pp. 167–183.
6. T. Liu, Q. Meng, J.-J. Huang, A. Vlontzos, D. Rueckert, and B. Kainz, "Video summarization through reinforcement learning with a 3D spatio-temporal U-net," *IEEE Trans. Image Process.*, vol. 31, pp. 1573–1586, 2022, doi: 10.1109/TIP.2022.3143699.
7. S. Rani and M. Kumar, "Social media video summarization using multi-Visual features and Kohnen's Self Organizing Map," *Inf. Process. Manage.*, vol. 57, no. 3, 2020, Art. no. 102190, doi: 10.1016/j.ipm.2019.102190.
8. K. Arai, "Image clustering method based on density maps derived from self-organizing mapping: SOM," *Int. J. Adv. Comput. Sci. Appl.*, vol. 3, no. 7, pp. 102–107, 2012, doi: 10.14569/IJACSA.2012.030714.
9. M. Otani et al., "Rethinking the evaluation of video summaries," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 7588–7596, doi: 10.1109/CVPR.2019.00778.
10. M. Gygli, H. Grabner, and L. Van Gool, "Video summarization by learning submodular mixtures of objectives," In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, 2015, pp. 3090–3098, doi: 10.1109/CVPR.2015.7298928.
11. K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, "Video summarization with long short-term memory," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Cham, Switzerland: Springer-Verlag, May 2016, pp. 766–782.
12. H. Wei, B. Ni, Y. Yan, H. Yu, X. Yang, and C. Yao, "Video summarization via semantic attended networks," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 216–223, doi:10.1609/aaai.v32i1.11297.
13. M. S. Nair and J. Mohan, "Static video summarization using multi-CNN with sparse autoencoder and random forest classifier," *Signal Image Video Process.*, vol. 15, no. 4, pp. 735–742, 2021, doi: 10.1007/s11760-020-01791-4.
14. P. Saini and K. Berwal, "ESKVS: Efficient and secure approach for keyframes-based video summarization framework," *Multimedia Tools Appl.*, pp. 1–29, Feb. 17, 2024, doi: 10.1007/s11042-024-18405-7.
15. J. Park, J. Lee, I.-J. Kim, and K. Sohn, "SumGraph: Video summarization via recursive graph modeling," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*, Glasgow, UK, 2020, pp. 647–663.
16. G. Wu, J. Lin, and C. T. Silva, "ERA: Entity relationship aware video summarization with Wasserstein GAN," 2021, *arXiv:2109.02625*.
17. Y. Zhang, Y. Liu, W. Kang, and R. Tao, "VSS-net: Visual semantic self-mining network for video summarization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 4, pp. 2775–2788, Apr. 2023, doi: 10.1109/TCSVT.2023.3312325.
18. J. Xie, X. Chen, S.-P. Lu, and Y. Yang, "A knowledge augmented and multimodal-based framework for video summarization," in *Proc. 30th ACM Int. Conf. Multimedia*, 2022, pp. 740–749, doi: 10.1145/3503161.3548089.
19. J. Xie et al., "Multimodal-based and aesthetic-guided narrative video summarization," *IEEE Trans. Multimedia*, vol. 25, pp. 4894–4908, 2022, doi: 10.1109/TMM.2022.3183394.
20. T.-C. Hsu, Y.-S. Liao, and C.-R. Huang, "Video summarization with spatiotemporal vision transformer," *IEEE Trans. Image Process.*, vol. 32, pp. 3013–3026, 2023, doi: 10.1109/TIP.2023.3275069.

SHAMAL KASHID is pursuing her Ph.D. degree in computer science and engineering (CSE) from the National Institute of Technology (NIT) Uttarakhand, Srinagar, 246174, India. Her research interests include image processing, deep learning, and video processing. Kashid received her M.Tech. degree in CSE from Savitribai Phule Pune University. Contact her at kashid.shamalphd2021@nituk.ac.in.

LALIT K. AWASTHI is a director with NIT Uttarakhand, Srinagar, 246174, India. His research interests include mobile computing, sensor networks, and P2P networks. Awasthi received his Ph.D. degree in CSE from the Indian Institute of Technology Roorkee, Roorkee, India. He is a Senior Member of IEEE. Contact him at lalitdec@gmail.com.

KRISHAN BERWAL is an associate professor at the Military College of Telecommunication Engineering, Mhow, 453441,

India. His research interests include machine learning, deep learning, and video processing. Berwal received his Ph.D. degree in CSE from Visvesvaraya National Institute of Technology Nagpur in March 2019. He is a Senior Member of IEEE, a life member of the International Society for Technology in Education, Institution of Electronics and Telecommunication Engineers, and Indian Unit International Association of Pattern Recognition, and a member of the Institution of Engineering and Technology and Association for Computing Machinery. Contact him at k2b@ieee.org.

PARUL SAINI is as an assistant professor with DIT University, Dehradun, 248009, India. Her research interests include image processing, deep learning, and video processing. Saini received her Ph.D. degree from NIT Uttarakhand, India, in 2023. Contact her at parul.saini@dituniversity.edu.in.