Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 7453744, 25 pages
https://doi.org/10.1155/2022/7453744

Research Article

An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning

Hafiz Burhan Ul Haq,[1] Muhammad Asif,[1] Maaz Bin Ahmad,[2] Rehan Ashraf,[3] and Toqeer Mahmood[3]

[1]*Faculty of Computer Science, Lahore Garrison University, Lahore, Pakistan*
[2]*College of Computing and Information Science, Karachi Institute of Economics and Technology, Karachi, Pakistan*
[3]*Faculty of Computer Science, National Textile University, Faisalabad, Pakistan*

Correspondence should be addressed to Toqeer Mahmood; toqeer.mahmood@yahoo.com

Received 9 July 2021; Revised 11 December 2021; Accepted 25 March 2022; Published 12 May 2022

Academic Editor: Nadeem Qazi

The advancements in digital video technology have empowered video surveillance to play a vital role in ensuring security and safety. Public and private enterprises use surveillance systems to monitor and analyze daily activities. Consequently, a massive volume of data is generated in videos that require further processing to achieve security protocol. Analyzing video content is tedious and a time-consuming task. Moreover, it also requires high-speed computing hardware. The video summarization concept has emerged to overcome these limitations. This paper presents a customized video summarization framework based on deep learning. The proposed framework enables a user to summarize the videos according to the Object of Interest (OoI), for example, person, airplane, mobile phone, bike, and car. Various experiments are conducted to evaluate the performance of the proposed framework on the video summarization (VSUMM) dataset, title-based video summarization (TVSum) dataset, and own dataset. The accuracy of VSUMM, TVSum, and own dataset is 99.6%, 99.9%, and 99.2%, respectively. A desktop application is also developed to help the user summarize the video based on the OoI.

1. Introduction

Security is the primary concern of the entire world. Besides taking a few other security measures, video surveillance cameras have been installed on private and public premises to cope with this challenge. Several kinds of security surveillance cameras (i.e., static and moveable) have been installed in public places, homes, shops, airports, banks, and so on. These cameras play a vital role in real-time monitoring and detecting suspicious behavior. They are also helpful in investigating events or crime scenes, for example, road accidents, robbery, murder, and terrorist activity [1].

Furthermore, the global estimated number of currently operational cameras is more than 770 million [2]. These cameras usually remain active round the clock and generate more than 2,500 petabytes of video data per day [3]. Figure 1 exhibits daily statistics of the real-world data produced by the video surveillance cameras.

Considerable progress has already been made in developing video analytic tools that automatically perform content-based video interpretation, including motion detection [4, 5], facial recognition [6, 7], people counting [8–10], and license plate recognition [11–13]. However, the issue is that manual (security guards, police officers, etc.) interventions are still required to analyze the recorded videos. Visual analysis of video content to extract meaningful information is complex and time-consuming because visual analysis needs to concentrate and watch the whole video [14]. It may also result in false negatives, especially in the case of long videos. Therefore, there is an ultimate need for a solution that helps in reducing human efforts and time for manual analysis. Multiple efforts are being made towards video summarization to address this concern and generate a video summary that quickly provides the whole video's gist [15]. The video summarization (VS) is creating a summary of extensive video content by detecting and presenting relevant

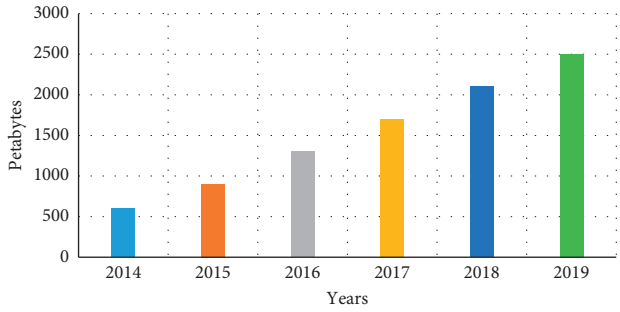


Figure 1: Daily data generated by surveillance cameras.

material to the potential users that are most informative and contain up-to-date information. VS is being used in security surveillance systems to detect and analyze suspicious or anomalous activities. Personal VS is used to share occasional videos on social media, generate sports highlights, make trailers of movies and serials and video content indexing to facilitate fast browsing of huge video through a video search engine, and so on [16–19].

The researchers have made several efforts to propose automatic VS. Most of the VS techniques generate a summary based on selecting keyframes representing the video through the skimming process [8–12, 15–17, 20–37]. Feature-based approaches for VS produce a generalized video summary rather than focusing on a specific object [20–37]. The shot boundary detection approaches are also well known for video summarization [38–45]. These approaches show limitations in detecting the object precisely, hence failing to fulfill the user's requirements. Clustering [20, 24, 46–49] and trajectory-based [17, 28] techniques summarize the video by focusing on similar activities, events, and objects. However, these approaches do not summarize any video containing information according to the user's interest. Consequently, these techniques limit the use of retrieval tasks and do not help enhance the users' observing experience.

This study presents an effective VS framework based on the OoI to cope with the issues of video summarization. The OoI refers to the objects such as person, car, mobile, and bike that a user selects to summarize the video by collecting all frames where the selected object appears. The proposed VS framework works in three steps: (i) the combination of the OoI selection phase, (ii) the object localization or detection phase, and (iii) the video summarization phase. Initially, the OoI selection is performed from the dictionary (a database of objects) to ignore the unnecessary noisy objects (other than the OoI) that are imperative for the segmentation of objects. After that, You Look Only Once (YOLOv3) detector is applied to localize the OoI. After localizing the OoI, the proposed VS algorithm summarizes the video based on the OoI.

Based on the above discussion, the contributions of the proposed work can be summarized as follows:

(i) In the OoI selection step, the proposed algorithm selects the object from the dictionary and ignores all the unnecessary objects automatically. After that, the YOLOv3 is used to detect the desired object.

(ii) The proposed VS framework can detect single and multiple objects present in the video.

(iii) The proposed VS algorithm effectively summarizes the video and overwhelms all the challenges shown in the VSUMM [50], TVSum [51], and own dataset.

(iv) The experimental finding highlights that the proposed VS framework performs tremendously as compared to state-of-the-art methods in the area of video summarization.

The rest of the paper is organized as follows: Section 2 describes the literature review for exiting techniques. Section 3 presents the proposed VS model that explains the video summarization method. Section 4 discusses the results and comparison with other techniques, and finally, the conclusion is discussed in Section 5.

2. Literature Review

Several techniques have been proposed. Uchihachi et al. [42] have presented the packing algorithm to define the excellent layout. This algorithm has packed the selected frame to produce the best sequence in the block, and the algorithm organizes several shots and produces videos concisely. However, this approach is useful only when the camera is moving. Chong-Wah Ngo et al. [34] presented a method that depends on the perceptual quality and redundancy reduction to maintain the content of the video summary. The video clusters are generated through temporal slices coherency to partition the video into shots and subshots. After that, the authors adopted a motion attention framework presented in [52] to analyze the clusters and quality of the shots. The temporal graph is also formed to prescribe the importance of clusters. The graph's attention values are used to select the appropriate scenes to create a video summary. The summary generated using the proposed approach is only about 10–25% of the entire video.

Rav-Acha et al. [32] have also presented the video's abstraction, where all events and activities of the video are consolidated to generate a video summary. The work has been performed by detecting the moving objects directly and then applying video optimization using detected objects. However, this approach cannot join the parts of different scenes because of discontinuity. Damnjanovic et al. [35] introduced an event-based video précising technique. Initially, the technique calculates each frame's energy by summating the absolute difference between the current frame and the reference frame pixel values. In this way, all the existing events in frames are determined. Later on, the video summarization algorithm is applied to extract keyframes. The suggested approach is suitable in a static environment. However, in the case of a dynamic or changing background, the system's performance is unacceptable. Almeida et al. [30] have presented a video summarization method that performs three steps to summarize the video content: extraction of visual features, summarizing the video, and its filtration. First, the color histogram's extraction of visual features labels the visual contents by manipulating the color. Second, a speedy and straightforward algorithm is

implemented for condensing the video. The purpose of the algorithm is to detect similar content and select the relevant frames. Finally, filtration is performed on selected frames to remove noise and redundant data from the video to generate the video summary. Furthermore, the proposed approach is highly hardware-dependent and requires high computational speed. Moreover, the system's performance is unacceptable for objects summarization in lengthy videos. Wang and Ngo [20] have discussed a method in which the hierarchical hidden Markov model recognizes the motion features. It classifies low-level features to a high level using the semantic concept. After identifying the object's features, the most representative clips are selected from the video to generate a video summary. The generated summary is 50 times faster than the original video. The inclusion of similar shots in the video is a major limitation of this approach that causes redundancy. Another limitation is the ignorance of the objects that are in a moving state.

Miniakhmetova and Zymbler [40] have described the method of personalized video summarization that works in two stages. The first stage is video structuring, where various scene detection techniques are performed and a video summary is generated. In the second stage, objects are detected from the subset of video scenes using the detection bank. The video summary is generated that consists of the most influencing scenes in which objects are detected, which in turn become a region of the user's interest. The authors only proposed an idea to implement such a system, not any built prototype. Varghese and Nair [38] proposed a method in which video can be summarized by performing three main steps: shot boundaries detection, redundant frame elimination, and stroboscopic imaging. The shot boundary detection compares the current frame with the neighboring frame. Repetitive frames are removed using the structural similarity index (SSI). The stroboscopic is also used to understand the common background and show the existing activities in the video. Compared to the original video, the presented technique reduces the volume by 55% in the summarized video. Lai et al. [45] presented a frame recomposition-based approach using a clustering algorithm, optical flow, and background subtraction to detect foreground objects. The foreground object has been detected by fusing a group of pixels. After detecting the objects/activity, a sliding window has been used to combine the detected objects in consecutive frames to build a spatiotemporal trajectory. The video summary is generated by combining the entire spatiotemporal trajectory, and the algorithm has achieved an accuracy of 97%. However, this technique is suitable only when the camera position remains static. Srinivas et al. [21] discussed how video could be summarized by computing three factors. First, it assigns the score to each frame based on various features such as quality, color, hue, statically attention, temporal segment, demonstration, and uniformity. Second, it assigns weights to each score based on feature importance for getting keyframes. The standard deviation is used for the assignment of weightage. Finally, the redundancy is eliminated by eliminating repetitive frames, which are collected based on their ascending order score. The presented method showed slightly better performance than the improved frame-blocks features method (IFBFM). However, the comparison with state of the art is not presented.

Davila and Zanibbi [33] have discussed the frame selection in lecture videos based on segmentation by diminishing the conflicts between content regions, removing objects, and rebuilding each frame to generate a video summary. The compression rate of the approach is not specified while discussing the video summarization, and the approach is tested on lecture videos only. Ajmal et al. [29] have discussed a method in which human motion has been tracked with the Kalman filter's help to find the trajectory. The color features are helpful for video, where the color histogram is used for shots-detection and generates a video summary. However, this approach is designed for surveillance videos only. Ma et al. [39] have presented a collaborative representation of the adjacent frame technique to detect an abnormal frame and remove noisy content from the video. Keyframes are selected using minimum sparse reconstruction to remove the noisy data and prevent the loss of important information. The frame having high collaborative representation error is considered a keyframe. A greedy iterative algorithm is utilized for model optimization that controls the count of keyframes with the help of the average percentage of reconstruction (APOR) and the sparse boundary. However, this approach is ineffective for videos with different frames. Sridevi and Kharde [53] performed video summarization by detecting highlights. In this method, two-stream architecture is used that consists of a deep convolutional neural network (DCNN). The two-dimensional convolutional neural network (2D-CNN) is used to exploit spatial information, and the three-dimensional convolutional neural network (3D-CNN) is used to exploit temporal information to score the video segment highlights. This method achieved a 43.9 precision rate.

Meyer et al. [54] presented a cloud-based system known as HOMER for the generation of video highlights. In this system, the video summary can be generated by detecting the user's emotions. Two different datasets are used for experimental analysis: a dataset filmed through a dual-camera setup and a home video randomly selected from Microsoft's video titles in the wild (VTW) dataset. Resultantly, HOMER achieved 38% improvement from baseline. Afzal and Tahir [55] described a video summarization by combining ResNet 152 and gated recurrent unit (GRU). In this method, ResNet 152 is used to extract deep features that existed in the video. Similarly, a gated recurrent unit (GRU) is used to improve the method's robustness and performance. The experimental analysis was performed on the SumMe dataset, and F-measure was 43.7. Gunawardena et al. [56] performed OoI-based video summarization by generating features from the video according to different scenes with the help of VGG16-1. The technique generates features of OoI using VGG16-2 by taking the frame (containing objects) from the selected video. The accuracy of the proposed method is 88%. However, only three objects are used in the experiments and the computational time is high. Meng et al. [57] proposed a technique that summarizes a video into several key objects through representative object proposals generated from

video frames. The proposed technique is tested only on a few objects such as a clock, microphone, and signs. The overall accuracy of the technique is not given. Fataliyev et al. [58] proposed a method to summarize a video with the help of object motion pattern analysis. The method is based on key positions extraction and index frame generation. The Gaussian mixtures are used for object extraction and adaptive background subtraction. For noise reduction, morphological opening and closing operations are adopted. The overall accuracy of the method is 82%. However, the method is tested only for a single object, like a person.

Table 1 summarizes some of the existing VS techniques discussed in the current section.

In public places, many surveillance cameras have been installed to monitor suspicious activities such as mobile snatching, terrorism, and robbery, where the information contained by every single frame is essential. Most of the existing techniques work on the principle of keyframe selection by eliminating the redundant frames that may result in the loss of important information related to a user's interest. Due to the limitation (disappearance of object and event), these techniques cannot produce significant results. Though some techniques summarize the video based on OoI, the main limitation of these techniques is their high computational power requirements and low accuracy. So, there is a need for a framework that should provide robustness, high accuracy, support for multiple static and dynamic objects, and provision for investigating numerous scenarios. The framework should be able enough to accommodate a wide range of OoI.

3. Proposed Framework

The proposed VS framework takes video and OoI as input. After that, the frames having OoI are detected using an object detection module. Finally, only the detected frames are combined to produce a video summary as an output. The architecture of the proposed framework is shown in Figure 2. It comprises the following main modules:

(i) Selection of input: it takes the video and OoI

(ii) OoI detection module: it detects the OoI from the videos using a deep learning technique

(iii) The video summarization module takes the frames that contain an OoI and generates the video summary as an output

The description of each module is given in the following subsections.

3.1. Selection of Inputs. In this work, a desktop application is developed using Python that provides an interactive user interface for selecting input video and OoI. The detailed working of the application is as follows.

3.1.1. Input Video Selection. The front-end of the application developed for selecting input video is shown in Figure 3. It contains the information related to the input and performs a video format validation check. The application only supports MP4 and AVI standard formats.

3.1.2. OoI Selection. After selecting the video, the next step is to choose the object type (i.e., OoI) to be detected from the input video. The user may select the OoI from the dropdown menu, as shown in Figure 4. A dictionary has been developed using the MS COCO dataset. The dataset contains 330 thousand images in which more than 200 thousand images are labeled. Moreover, it has 15 million object instances of 80 object categories of car, person, suitcase, and so on. The 11 supercategories of MS COCO datasets are person, animal, outdoor objects, indoor objects, vehicle, sports, kitchenware, food, appliance, furniture, and electronics [59]. The pistol dataset contains 2986 images with a single annotation class known as the pistol. The pistol dataset images contain cartoon and staged studio quality images of guns and pistols in hand [60]. A sample set of images from the MS COCO and pistol datasets is shown in Figures 5 and 6.

3.2. OoI Detection Module. In the proposed framework, YOLOv3 (You Look Only Once) [42] is used to detect the OoI. This module determines the scene, event, and frame where the desired object is located. YOLOv3 uses a variant of Darknet containing 53 layers that are trained on Imagenet. Furthermore, 53 more layers have been added for task detection that provides the fully convolutional underlying architecture for YOLOv3, consisting of 106 layers. There is no pooling layer in YOLOv3. A convolutional layer with stride 2 is used to downsample the feature maps to prevent the loss of low-level features [61]. It applies a single neural network to the full video, where the network divides the frames into regions, and it predicts probabilities and bounding boxes [62]. The architecture of the YOLOv3 is presented in Figure 7.

In YOLOv3, each class score is predicted with the help of logistic regression, and the prediction of multiple labels of the object can be performed using a threshold. However, classes with scores higher than a threshold are assigned to the box [59]. In the proposed framework, object detection is performed using a bounding box to demarcate the OoI. In case of multiple objects in a frame, this method helps to describe the spatial location of an OoI. The prediction of the bounding box is described in Figure 8.

In Figure 8, (b_x, b_y) are the x-y dimensions of the bounding box. However, for each bounding box, YOLO v3 predicts four coordinates (t_x, t_y, t_w, t_h). If the cell is offset from the top left corner of the image by (C_x, C_y) and the bounding box prior has width and height p_w, p_h, then the predictions are presented as follows:

$$\begin{aligned} b_x &= \sigma t_x) + c_x, \\ b_y &= \sigma\left(t_y\right) + c_y, \\ b_w &= p_w e^{t_w}, \\ b_h &= p_h e^{t_h}. \end{aligned} \tag{1}$$

Table 1: Summary of the existing techniques for video summarization.

Sr. no.	Author's name	Dataset	Approach	Targeted domain	Observations/limitations
1	Varghese and Nair [38]	Random documentary, football, home video, and advertisement videos	Detect the shot boundaries for selecting keyframes, eliminate redundant frames by using SSI, and then apply stroboscopic effect for checking the common background frames.	Documentary, sports, home video, advertisement videos, and news	55% reduction of video volume.
2	Srinivas et al. [21]	Random video sequences	Compute the score of each frame and ranking frames and then eliminate the repetitive frames.	All kinds of videos such as documentary, sports, home video, advertisement videos, and news	Overall there is a 1.8% improvement in the IFBFM.
3	Almeida et al. [30]	Randomly videos selected from open video project	Extract the key feature by using macroblock and also apply color histogram that selects the most representative frames.	All kinds of videos such as compressed videos	Highly hardware-based, required high computational speed, also missing the object in lengthy videos.
4	Wang and Ngo [20]	TRECVID BCC	Summarize the rushes videos using the two-level hierarchical hidden Markov model (HHMM); it is discussed that precise videos are based on the event and object.	All kinds of rushes videos	Events or objects are ignored in a moving state and unable to remove the similar shots far away.
5	Lai et al. [45]	Randomly selected video sequences	It removes the spatiotemporal segments that are irrelevant by using frame recomposition. Extracted objects are rejoined in the spatiotemporal trajectory for generating a video summary.	All kinds of videos such as documentary, sports, home video, advertisement videos, and news	Objects can only be detected through a fixed camera. Extract 97% activity from the original video.
6	Ma et al. [39]	VSUMM and TVSum	Optimize the model based on the adjacent frame using the iteration algorithm that takes the average percentage of frame reconstruction.	Documentary, sports, home video, advertisement videos, and news	Only focused on fixed-size frames.
7	Rav-Acha et al. [32]	Randomly selected surveillance activity	Video synopsis approaches that reduce spatiotemporal redundancy and express the dynamic appearance of an object reduce spatiotemporal redundancy and express an object's dynamic appearance.	Surveillance activities and movies	Due to discontinuity, this algorithm will not join different parts of the scenes.
8	Davila and Zanibbi [33]	Lecture videos	Summarize the video by reducing the ambiguity between the content regions and navigating the lecture video on hand-written content existing on the whiteboard.	Lecture videos	96.28% recall with a better compression rate.
9	Uchihachi et al. [42]	Staff meeting video	Video can be summarized by measuring a shot's importance and eliminating redundant scenes.	All kinds of videos such as documentary, sports, home video, advertisement videos, and news	The algorithm efficiently packed and demonstrated the vary-sized keyframes.
10	Damnjanovic et al. [35]	Surveillance videos	Detect and cluster the events that existed in surveillance video. Also, create two types of summary static and dynamic.	Surveillance videos	The major limitation is the false detection of events in case of changes in the background.
11	Chong-Wah Ngo et al. [34]	Randomly selected video sequences	For video summarization, this approach captured attention values and video structure. For removing redundancy, video can be structured based on scenes, clusters, and so on in a hierarchical tree.	Song's videos and home-based videos	Generating 10% to 25% video summary.
12	Miniakhmetova and Zymbler [40]	YouTube videos	Construct a video summary based on user remarks that include like, dislike, and neutral based on influencing scene like object appearance.	YouTube videos	There is no prototype of the system to implement a video summarization model.
13	Ajmal et al. [29]	Hostel video	Person can be detected by using a histogram of oriented gradient (HOG) with the help of the support vector machine (SVM) classifier, and motion is tracked with the Kalman filter's help.	Surveillance videos	The system reduces video storage and saves time by making browsing fast.

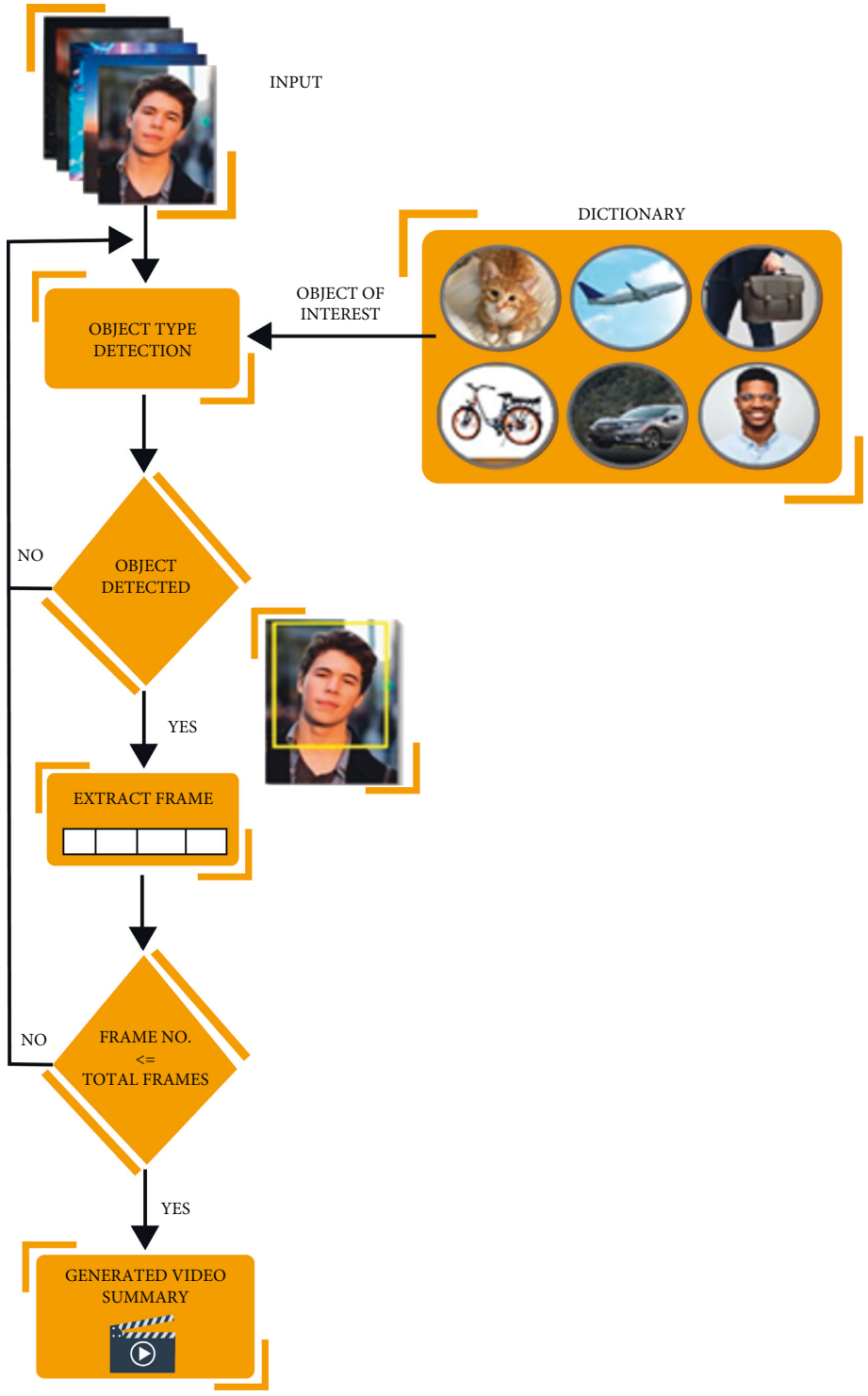


Figure 2: Architecture of the proposed framework.

In this work, YOLOv3 is used for OoI detection because it is much faster than its competitors [63]. Figure 9 shows the comparison of YOLO v3 with other object detection models in terms of speed. The processing speed of YOLO v3 is 45 fps that is quite impressive compared to Single Shot Detectors (SSD), Faster-RCNN, and R-FCN [63, 64]. However, the accuracy of YOLOv3 is less than F − RCNN*, but the processing speed is much higher; that is, it processes 45 fps, while the Faster-RCNN family processes only 5 fps [63, 64]. Figure 10 shows the comparison of YOLOv3 with its competitors.

3.3. Video Summarization Module. VS module takes the frames that contain an OoI as input and generates the video

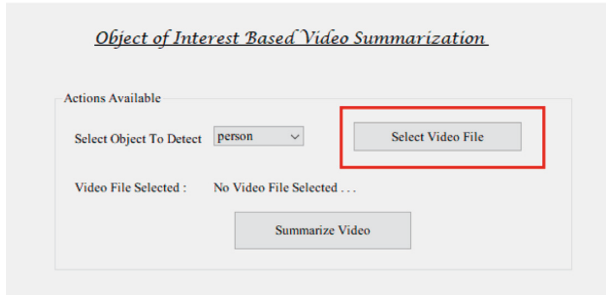


Figure 3: Input video selection.

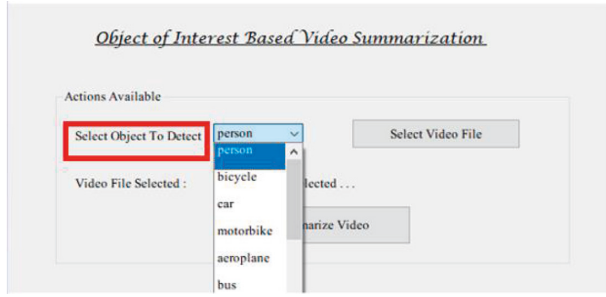


Figure 4: OoI selection.

Figure 5: Sample images of the MS COCO dataset.

Figure 6: Sample images of the pistol dataset.

summary as an output. The steps of the VS process are described as follows:

(1) Read the current frame from the input video.

(2) Perform OoI detection in the current frame using YOLOv3.

(3) If OoI is found, save the current frame in the buffer. Otherwise, discard it.

(4) If the current frame is the last, go to step 5. Otherwise, go to step 1 for the next frame.

(5) Finalize the video summarization process by combining all the buffered frames having OoI.

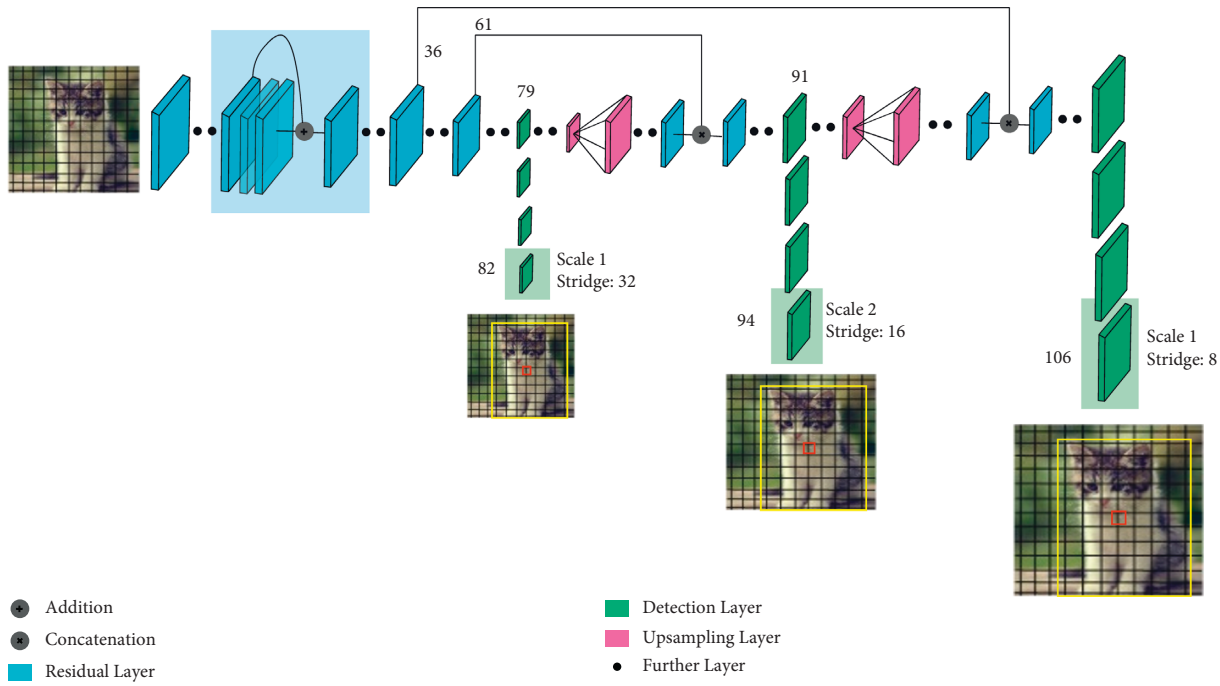


Figure 7: YOLOv3 architecture [61].

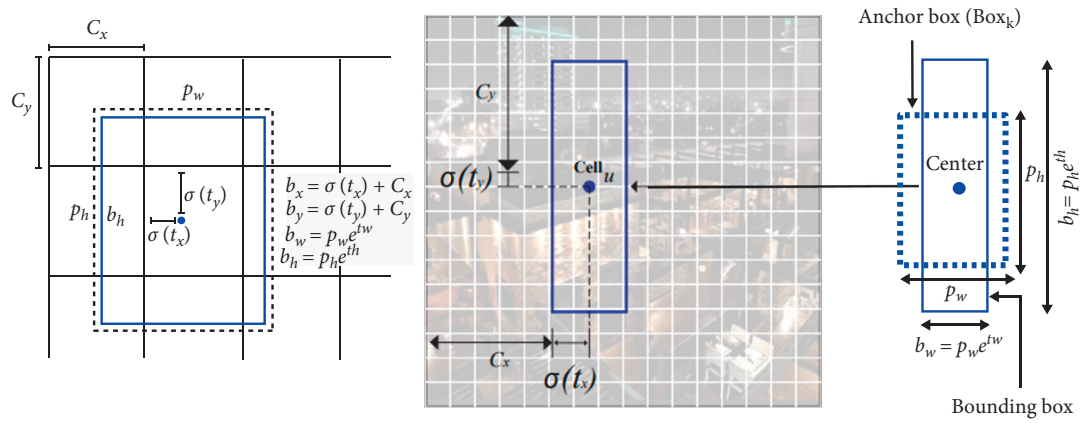


Figure 8: Prediction of the bounding box.

The algorithmic flow of the proposed video summarization framework is also given in Algorithm 1.

4. Experimental Analysis

All the experiments were performed on a machine equipped with an Intel Core i5-6200U processor (running at 2.4 GHz) and 8 gigabytes (GB) of RAM, and Python was used as the programming language.

In this work, a subjective method is used to evaluate the performance of the proposed framework. For each test stream, summarized video is generated manually (with the help of a video editing tool, "Filmora") and automatically through the proposed framework. The performance of the proposed framework is evaluated based on precision, recall, F1-score, and accuracy. The mathematical expressions for these evaluation parameters are given in the following equations:

$$\begin{aligned} \text{precision } (P) &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ \text{recall } (R) &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ F1 - \text{score} &= \frac{2 \times P \times R}{P + R}, \\ \text{accuracy} &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}. \end{aligned} \tag{2}$$

Three different datasets are used, the VSUMM dataset, the TVSum dataset, and own dataset, to compare and validate the efficiency of the proposed framework with the

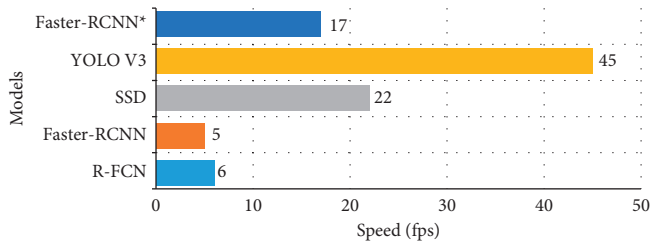


Figure 9: Speed-based comparison of YOLOv3 with its competitors.

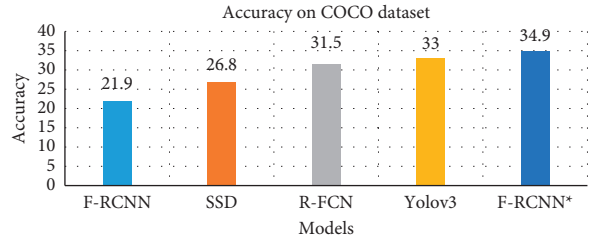


Figure 10: Accuracy-based comparison of YOLOv3 with its competitors.

Algorithm: Video summary generation
Algorithm 1: Process of VS
Inputs: Video X, OoI O
Output: Summarized video Y
Start process: VS(X, O)
$N \leftarrow$ No. of frames (X)
for $i = 0$ to $N - 1$, do
Read the current frame $F[i]$
Status OoI detection ($f[i]$, OoI)
  if (status = = 1) **then**{
    $Y[j] = F[i]$
}
  Else
    Discard the frame
end for x
Saved Y

Algorithm 1: Video summary generation.

Table 2: VSUMM dataset sequences.

Sr. no.	Sequence	Duration	Resolution	No. of frames	Types of objects
1	Lectures	58 seconds	352 × 240	3,480	3
2	Clinic video	1.08 minutes	352 × 240	4,080	2
3	Documentary 1	3.17 minutes	352 × 240	11,820	4
4	News	9 seconds	320 × 240	540	2
5	Documentary 2	2.58 minutes	320 × 240	10,680	5

Table 3: TVSum dataset video sequences.

Sr. no.	Sequence	Duration	Resolution	No. of frames	Types of objects
1	Documentary 1	1.38 minutes	640 × 360	5880	2
2	Truck accident	5.22 minutes	640 × 360	19,320	5
3	Festival	1.50 minutes	640 × 360	6600	5
4	News	2.18 minutes	480 × 360	8280	2
5	Documentary 2	2.10 minutes	640 × 360	7800	3

manual method. The VSUMM dataset consists of 50 videos from the open video project (OVP). All VSUMM videos are in MPEG-1 form with 30 fps, 352×240 pixels resolutions. However, videos contained by the VSUMM dataset belong to several categories (educational, documentary, historical, ephemeral, and lecture), ranging from 1 to 4 minutes. The TVSum contains 50 videos taken from different video websites that belong to several genres such as news, how-to, documentary, vlog, and egocentric. The own dataset contains the videos taken from multiple sources. The video is in AVI and MP4 format with different resolutions such as 320×240, 352×240, 640×360, 854×480, and 1280×720. Tables 2–4 list the sample test video sequences taken from VSUMM datasets, TVSum datasets, and own datasets, along with their specifications.

Extensive experiments are performed to evaluate the performance of the proposed framework on videos with different durations and resolutions. Some of the scenarios are discussed in the subsequent sections.

4.1. Evaluation of the VSUMM Dataset

4.1.1. Scenario 1. The video sequence consists of captured scenes from the lecture video in this scenario. The video has a duration of 58 seconds and 352×256 resolution. In this video, the person is considered an OoI. Hence, the video summarization is performed based on the object person. This video has been summarized by using both user-based and the proposed automated model.

Figure 11 shows frame-level (few frames) comparisons of the proposed method with the manual method. It reveals that the frames captured by both methods are the same in number, and there is no missing frame in this scenario.

The confusion matrix for the person as an OoI is shown in Table 5. It shows that all frames containing the person have been successfully detected by the proposed method. There was no error in the detection.

4.1.2. Scenario 2. In this scenario, the video sequence contains the captured scenes from the clinic. The video has a duration of 1.08 minutes, and its resolution is 352×256. It contains several objects such as a person, glasses, pen, and clock. In the video, the person is considered as OoI.

Figure 12 shows frame-level comparisons of the proposed method with the manual method. It shows that all the frames captured by both methods are the same in number, and there is no missing frame that cannot be detected by the proposed method.

The confusion matrix for the person taken as an OoI is shown in Table 6. It shows that, out of 420 frames containing the person, all frames have been successfully detected by the proposed method.

4.1.3. Scenario 3. In this scenario, the video sequence contains the captured scenes from the documentary on farmer living style. In the video, a person is considered an OoI. The video has a duration of 3.17 minutes, and its resolution is 352×256.

Figure 13 shows frame-level (few frames) comparisons of the proposed method with the manual method. It reflects that the first four frames are captured in both methods, while the fifth frame is wrongly predicted by the proposed method.

The confusion matrix for the person as an OoI is shown in Table 7. It shows that, out of 3780 frames containing the person, 3778 frames are detected by the proposed method, while 2 of the frames are wrongly predicted.

4.1.4. Scenario 4. In this scenario, the video sequence contains the captured scenes from the news. In the video, the person is considered an OoI. The video has a duration of 9 seconds with a resolution of 352×256.

Figure 14 shows frame-level comparisons of the proposed method with the manual method. It shows that all the frames captured by both methods are the same in number, and there is no missing frame that cannot be detected by the proposed method.

The confusion matrix for the person taken as an OoI is shown in Table 8. It shows that, out of 360 frames containing the person, the proposed method has successfully detected all of the frames.

4.1.5. Scenario 5. In this scenario, the video sequence contains the captured scenes from the documentary on GYM workouts. The video has a duration of 2.58 minutes, and its resolution is 352×256. It comprises several objects such as person and car. In the video, the car is considered an OoI. Thus, the VS is performed using the object "car."

Figure 15 shows frame-level (few frames) comparisons of the proposed method with the manual method. It reflects that the first three frames are captured in both methods, while the fourth and fifth frames are missed by the proposed method. The missing frame is the size of persons in that frame, that is, too small, which can be visualized through naked eyes but not by the proposed method.

The confusion matrix for the car as an OoI is shown in Table 9. It indicates that, out of 300 total frames containing the car, 120 frames are detected by the proposed method. There are 180 frames in which a car exists, but the proposed framework did not detect them.

4.1.6. Results Summary. Table 10 presents the experimental results of the proposed framework. Several scenarios have been taken from different scenes or locations such as lectures, documentaries, news, and GYM containing various objects such as person and cars.

The object types and scenarios provide details of objects considered an OoI regarding the specific scenario. The duration of the summarized video is recorded that describes the duration detail before and after the processing. In best cases (lecture, documentary, and news), the recall and precision are 100%, showing that the proposed

Table 4: Test video sequences.

Sr. no.	Sequence	Duration	Resolution	No. of frames	Types of objects
1	Airport	24 seconds	1280 × 720	1440	5
2	Street video	3.06 minutes	1280 × 720	11,160	3
3	Car parking	5.16 minutes	854 × 480	18,960	3
4	Mobile snatching clips	2.49 minutes	640 × 360	10,140	5
5	Bike snatching	30 seconds	640 × 360	1800	3
6	Pistol testing	3.03 minutes	640 × 360	10980	3

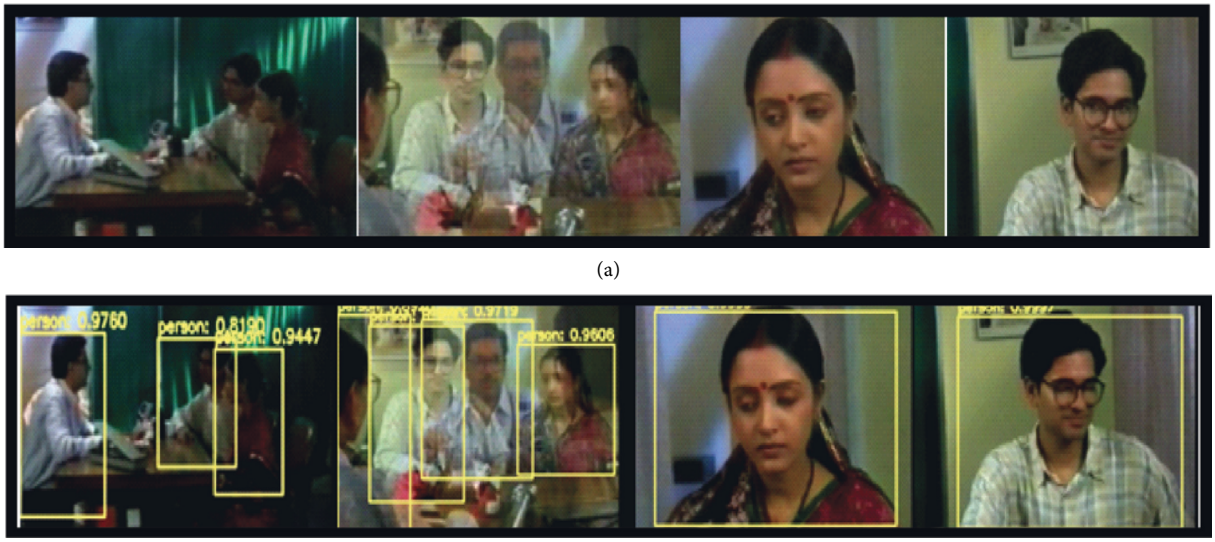
(a)

(b)

Figure 11: Detection of person: (a) manual method and (b) proposed method.

Table 5: Confusion matrix based on the person.

		Predicted label	
		OoI	Outlier
Actual label	OoI	540	0
	Outlier	0	2940



(a)

(b)

Figure 12: Detection of person: (a) manual method and (b) proposed method.

Table 6: Confusion matrix based on the person.

		Predicted label	
		OoI	Outlier
Actual label	OoI	420	0
	Outlier	0	3660

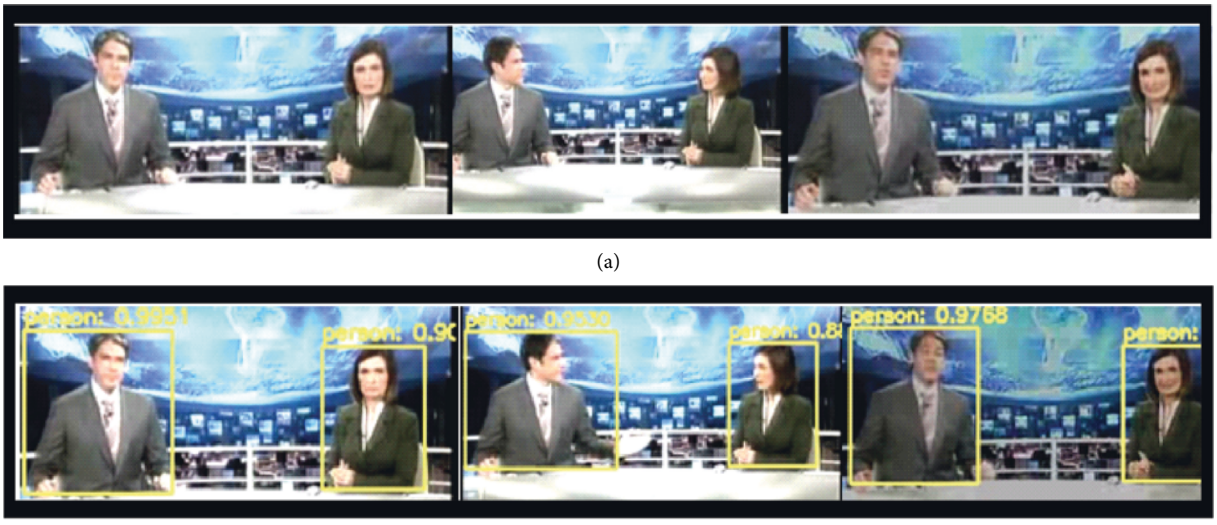
(a)

(b)

Figure 13: Detection of person: (a) manual method and (b) proposed method.

Table 7: Confusion matrix based on the person.

		Predicted label	
		OoI	Outlier
Actual label	OoI	3778	0
	Outlier	0	8038



(a)

(b)

Figure 14: Detection of person: (a) manual method and (b) proposed method.

Table 8: Confusion matrix based on person.

		Predicted label	
		OoI	Outlier
Actual label	OoI	360	0
	Outlier	0	180

(a)

(b)

Figure 15: Detection of car: (a) manual method and (b) proposed method.

Table 9: Confusion matrix based on car.

		Predicted label	
		OoI	Outlier
Actual label	OoI	120	0
	Outlier	180	10377

Table 10: Experimental results of the proposed framework on the VSUMM dataset.

Sr. no.	Object type	Scenarios	Video duration	Summarized duration	Saved time (%)	P (%)	R (%)	$F1$-score (%)	Accuracy (%)
1	Person	Lecture	58 sec.	9 sec.	84.4%	100	100	100	100
2	Person	Documentary 1	1.08 min.	7 sec.	89.7%	100	100	100	100
3	Person	Documentary 2	3.17 min.	1.3 min.	68%	99.9	100	99.4	99.9
4	Person	News	9 sec.	6 sec.	33.3%	100	100	100	100
5	Car	GYM	2.58 min.	2 sec.	98.8%	97	40	56	98.2
			Total time 8.30 min.	Total summarized time = 1.27 min.	Total saved time 82.84%				Overall accuracy 99.6%

framework accurately identifies the object and generates a full video summary. Similarly, the recall is less in the worst cases (documentary 2 and GYM). The reason is that the size of objects present in the frame is too small, and the video quality is not good. Therefore, the proposed method could not detect the object. The overall accuracy of the proposed framework is 99.6%, and the total saved time is 82.84%.

4.2. Evaluation of the TVSum Dataset

4.2.1. Scenario 1. In this scenario, the video sequence consists of captured scenes from the documentary on the honey bee. The video has a duration of 1.38 minutes and 640×360 resolution. In this video, the person is considered an OoI.

Figure 16 shows frame-level (few frames) comparisons of the proposed method with the manual method. It reflects that the first three frames are captured in both methods, while the fourth frame is wrongly predicted by the proposed method.

The confusion matrix for the person as an OoI is shown in Table 11. It shows that all the frames captured by both methods are the same in number, while the proposed method wrongly predicted 21 frames.

4.2.2. Scenario 2. In this scenario, the video sequence consists of captured scenes from the news. The video has a duration of 2.18 minutes and 480×360 resolution. In this video, the person is considered an OoI.

Figure 17 shows frame-level (few frames) comparisons of the proposed method with the manual method. It reflects that the first four frames are captured in both methods, while the fourth frame is wrongly predicted by the proposed method.

The confusion matrix for the person as an OoI is shown in Table 12. It shows that all the frames captured by both methods are the same in number, while only one frame is wrongly predicted by the proposed method in this video.

4.2.3. Scenario 3. This video sequence consists of captured scenes from the truck accident video. In this video, the truck is considered an OoI. It comprises several objects such as a person, car, chair, and truck. It has a duration of 5.22 minutes and 640×360 resolution.

Figure 18 shows frame-level comparisons of the proposed method with the manual method. It shows that all the frames captured by both methods are the same in number, and there is no single frame that is missed by the proposed methods in the detection.

The confusion matrix for the truck as an OoI is shown in Table 13. It shows that all 2940 frames containing the truck have been successfully detected by the proposed method.

4.2.4. Scenario 4. In this scenario, the video sequence consists of captured scenes from the festival. In this video, the truck is considered an OoI. It comprises several objects such as a person, truck, and balloons. The video has a duration of 1.50 minutes and 640×360 resolution.

Figure 19 shows frame-level comparisons of the proposed method with the manual method. It shows that all the frames captured by both methods are the same in number, and there is no single frame that is missed by the proposed methods in the detection.

The confusion matrix for the truck as an OoI is shown in Table 14. It shows that the proposed method has successfully detected all the 60 frames containing the truck.

4.2.5. Scenario 5. In this scenario, the video sequence monitors pet dog behavior scenes. The video has a duration of 2.10 minutes and 640×360 resolution. It comprises several objects such as a person and a clock. In this video, the dog is considered an OoI.

Figure 20 shows frame-level comparisons of the proposed method with the manual method. It shows that all the frames captured by both methods are the same in number, and there is no single frame that is missed by the proposed methods in the detection.

The confusion matrix for the dog as an OoI is shown in Table 15. It shows that all 1740 frames containing the dog have been successfully detected by the proposed method.

4.2.6. Results Summary. Table 16 presents the experimental results of the proposed framework. In experimental analysis, several scenarios have been taken from different scenes or locations such as documentaries, festivals, and news containing the various type of objects such as a person, dog, and truck.

In best cases, such as truck accidents, festivals, and news, the recall and precision of the proposed methods is 100%, which shows that the proposed framework is capable of identifying the object precisely and generates a full video summary. Similarly, in worst cases, such as news and honey bee documentary, the precision is less. The overall accuracy of the proposed framework is 99.9%, and the total saved time is 78.82%.

4.3. Evaluation of Own Dataset

4.3.1. Scenario 1. In this video, the VS is performed based on the airplane as an object. It comprises several objects, for example, person, airplane, tree, and mountain. The video sequence consists of captured scenes from the airport environment in this scenario. This video has been summarized by using both user-based and the proposed automated model. The video has a duration of 24 seconds and 1280×738 resolution.

Figure 21 presents the frame-level comparisons of the proposed method with the manual method. It reveals that the frames captured by both methods are the same in number, and there is no missing frame in this scenario.

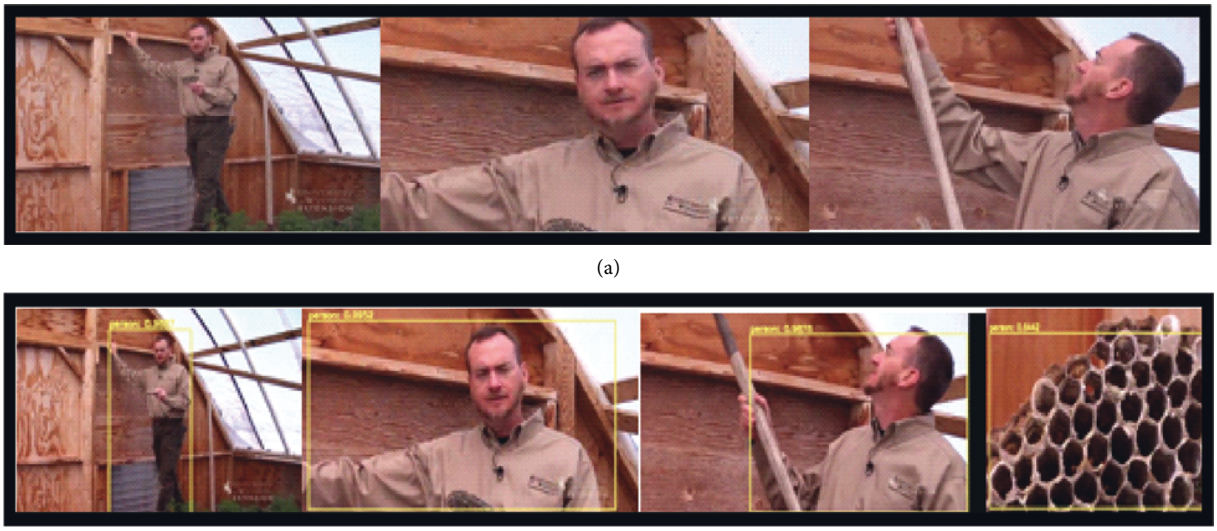
The confusion matrix for the airplane as an OoI is shown in Table 17. It shows that all 360 frames containing the airplane have been successfully detected by the proposed method. There was no error in the detection.

4.3.2. Scenario 2. The video sequence contains the captured scenes from the roadside/street environment in this scenario. It captures several objects such as a person, car, and bike. In the video, the car is considered an OoI. The video has a duration of 3.06 minutes with a resolution of 1280×738.

Figure 22 shows frame-level comparisons of the proposed method with the manual method. It reveals that all the frames captured by both methods are the same in number, and there is no missing frame that cannot be detected by the proposed method.

The confusion matrix for the car as an OoI is shown in Table 18. It shows that the proposed method has successfully detected all of the 120 frames containing the car.

4.3.3. Scenario 3. The video sequence contains the captured scenes from the parking environment in this scenario. It



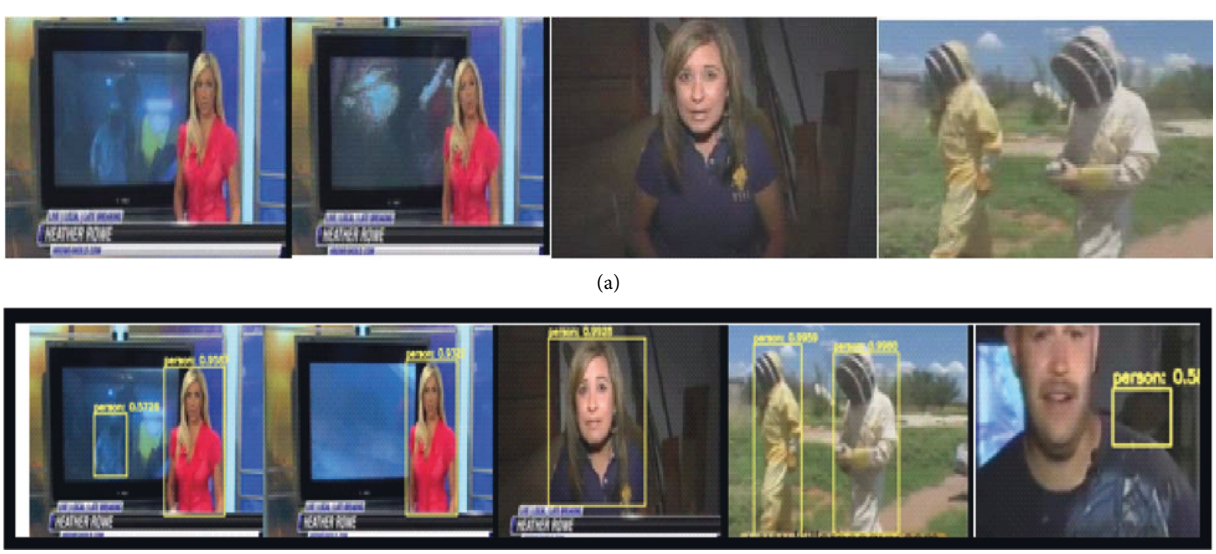
(a)

(b)

Figure 16: Detection of person: (a) manual method and (b) proposed method.

Table 11: Confusion matrix based on person.

		Predicted label	
		OoI	Outlier
Actual label	OoI	1599	210
	Outlier	0	4260



(a)

(b)

Figure 17: Detection of person: (a) manual method and (b) proposed method.

Table 12: Confusion matrix based on person.

		Predicted label	
		OoI	Outlier
Actual label	OoI	2579	1
	Outlier	0	5700

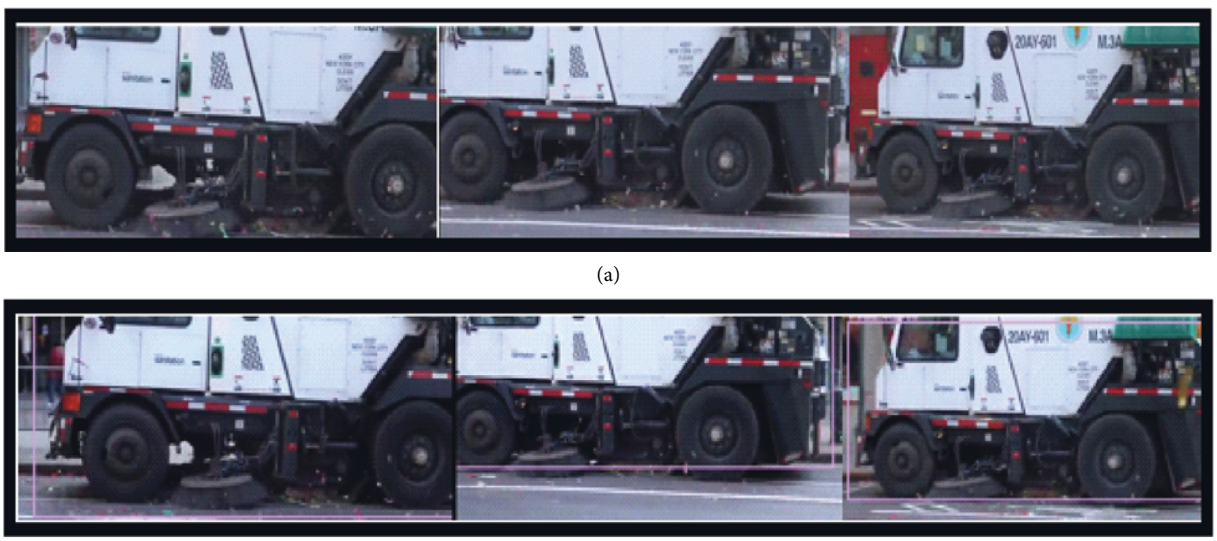
(a)

(b)

Figure 18: Detection of person: (a) manual method and (b) proposed method.

Table 13: Confusion matrix based on the truck.

		Predicted label	
		OoI	Outlier
Actual label	OoI	2940	0
	Outlier	0	16380



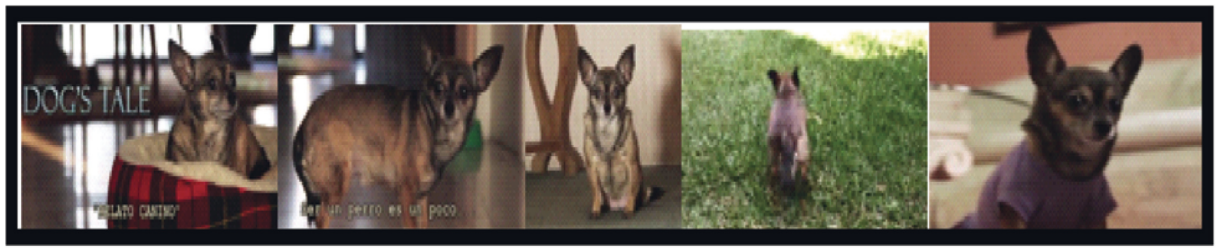
(a)

(b)

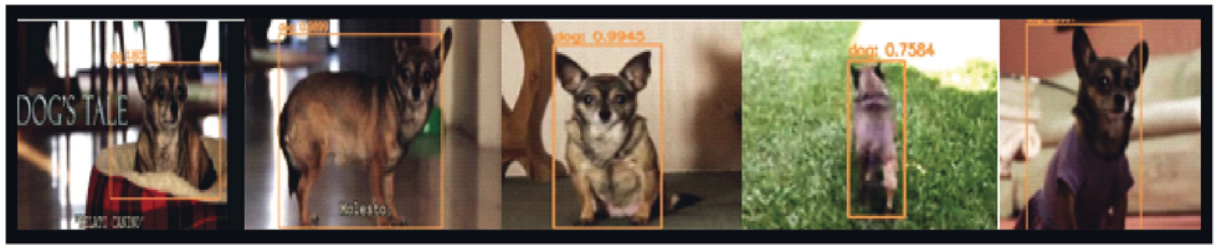
Figure 19: Detection of person: (a) manual method and (b) proposed method.

Table 14: Confusion matrix based on the truck.

		Predicted label	
		OoI	Outlier
Actual label	OoI	60	0
	Outlier	0	6540



(a)



(b)

Figure 20: Detection of dog: (a) manual method and (b) proposed method.

Table 15: Confusion matrix based on dog.

		Predicted label	
		OoI	Outlier
Actual label	OoI	1740	0
	Outlier	0	6060

Table 16: Experimental results of the proposed framework on the TVSum dataset.

Sr. no.	Object type	Scenarios	Video duration	Summarized duration	Saved time (%)	*P* (%)	*R* (%)	*F*1-score (%)	Accuracy (%)
1	Person	Documentary 1	1.38 min.	27 sec.	72.2%	98.7	100	100	99.6
2	Truck	Truck accident	5.22 min.	49 sec.	84.7%	100	100	100	100
3	Truck	Festival	1.50 min.	1 sec.	99%	100	100	100	100
4	Person	News	2.18 min.	43 sec.	68.8%	99.9	100	100	99.9
5	Dog	Documentary 2	2.10 min.	29 sec.	77.6%	100	100	100	100
			Total time 13.18 min.	Total summarized time = 2.49 min.	Total saved time 78.82%				Overall accuracy 99.9%

(a)

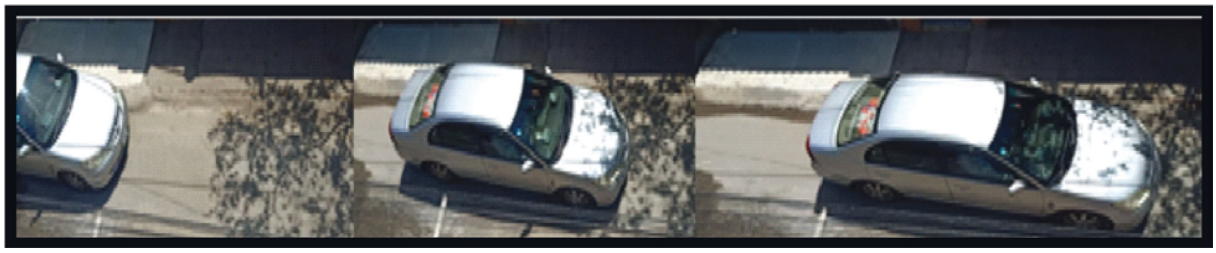
Figure 21: Continued.

(b)

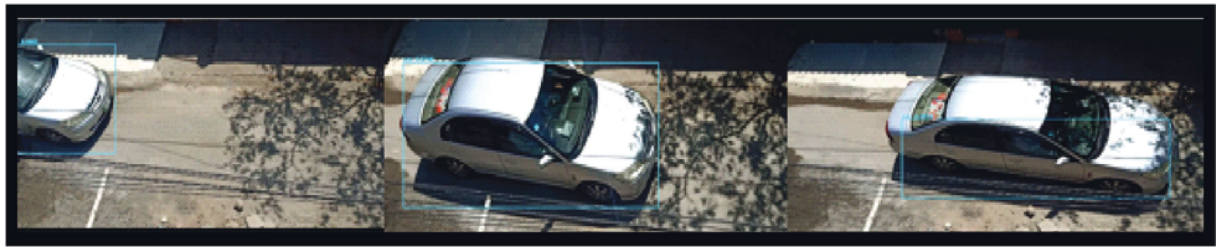
Figure 21: Detection of airplane: (a) manual method and (b) proposed method.

Table 17: Confusion matrix based on the airplane.

		Predicted label	
		OoI	Outlier
Actual label	OoI	360	0
	Outlier	0	1080



(a)



(b)

Figure 22: Detection of car: (a) manual method and (b) proposed method.

Table 18: Confusion matrix based on the car.

		Predicted label	
		OoI	Outlier
Actual label	OoI	120	0
	Outlier	0	11040

comprises several objects such as person, car, and tree. In this video, the person is taken as an OoI. The video has a duration of 5.16 minutes, and its resolution is 854×480.

Figure 23 shows the frame-level comparisons of the proposed method with the manual method. It reflects that the first four frames are captured in both methods, while the fifth frame is wrongly predicted by the proposed method.

The confusion matrix for the person as an OoI is shown in Table 19. It shows that, out of 779 frames containing the object person, 718 frames are detected by the proposed method. It shows 61 wrongly detected frames (frames without the person).

4.3.4. Scenario 4. In this video, mobile is taken as an OoI. The video sequence contains the precautions scenes related to mobile snatching in this scenario. The video has a duration of 2.49 minutes, and its resolution is 640×360. It comprises several objects such as person, mobile, and bike.

(a)

(b)

Figure 23: Detection of person: (a) manual method and (b) proposed method.

Table 19: Confusion matrix based on person.

		Predicted label	
		OoI	Outlier
Actual label	OoI	718	61
	Outlier	0	18181

Figure 24 shows frame-level comparisons of the proposed method with the manual method. It reflects that the first six frames are captured in both methods, while the seventh and eighth frames mentioned in Figure 21(a) are missed by the proposed method. The reason is that the mobile size in that frame is tiny that can only be seen through the naked eyes (manual method). Similarly, the proposed method wrongly predicts the seventh frame in Figure 23(b).

The confusion matrix for mobile as an OoI is shown in Table 20. It shows that, out of 992 frames containing the mobile, 660 frames are detected by the proposed method. There are 262 wrongly detected frames, and 3 are falsely detected.

4.3.5. Scenario 5. In this video, the bike is considered as an OoI. It comprises several objects such as a person, bike, and trees. The video sequence contains bike snatching scenes from the roadside in this scenario. The video's length is 30 seconds, and its resolution is 640×360.

Figure 25 presents a frame-level comparison of the proposed method with the manual method. It shows that all the frames captured by both methods are properly detected, so there is no incorrectly detected or missing frame.

Table 21 shows the confusion matrix for the bike as an OoI. It shows that, out of 300 frames containing the bike, all frames are detected by the proposed method. None of the frames is incorrectly detected or missed by the proposed method.

4.3.6. Scenario 6. In this scenario, the video sequence contains gun testing scenes. The video has a duration of 3.03 minutes, and its resolution is 640×360. It comprises several objects such as a person, umbrella, and pistol. In this video, the pistol is considered an OoI.

Figure 26 presents a frame-level comparison of the proposed method with the manual method. It shows that all the frames captured by both methods are properly detected; therefore, there is no incorrect detected or missing frame.

Table 22 shows the confusion matrix for pistol as an OoI. It shows that, out of 9000 frames containing the pistol, all of the frames are detected using the proposed method. None of the frames is incorrectly detected or missed by the proposed method.

4.3.7. Results Summary. Table 23 presents the experimental results of the proposed framework. In experimental analysis, several scenarios have been taken from different scenes or locations such as airport, parking, and street containing various objects such as a person, bike, and airplane.

In best cases such as street video, airport, and bike snatching, the recall and precision of the proposed method is 100%, which shows that the proposed framework identifies the object precisely and generates a summary of the full video. Similarly, the recall is less in the worst cases, such as mobile snatching. The reason is that the object's size in the video frame is tiny, which can be seen only through the naked eyes. Consequently, the proposed method is unable to detect such objects. The overall accuracy of the

(a)

(b)

Figure 24: Detection of the mobile: (a) manual method and (b) proposed method.

Table 20: Confusion matrix based on mobile.

		Predicted label	
		OoI	Outlier
Actual label	OoI	660	3
	Outlier	262	9215

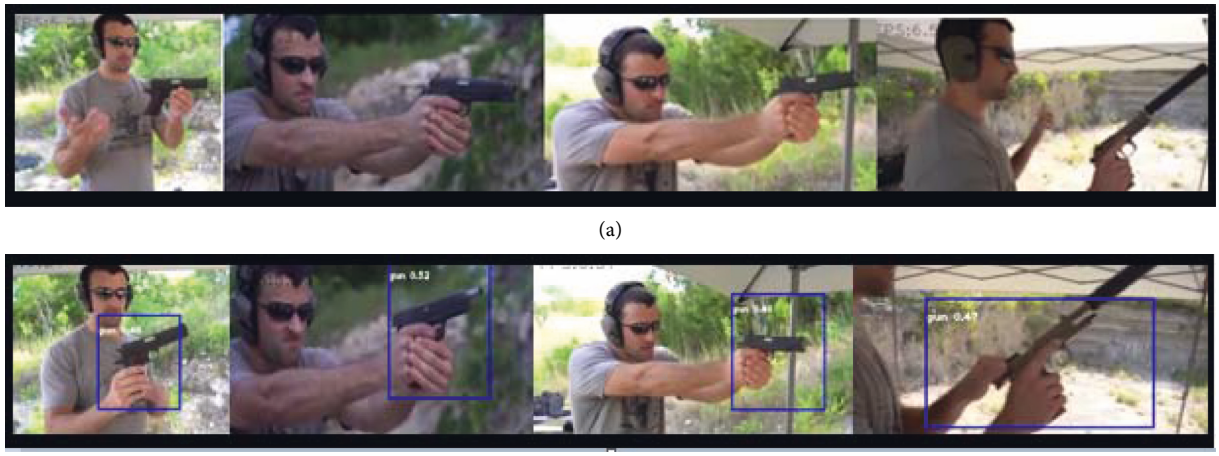
(a)

(b)

Figure 25: Detection of the bike: (a) manual method and (b) proposed method.

Table 21: Confusion matrix based on the bike.

		Predicted label	
		OoI	Outlier
Actual label	OoI	300	0
	Outlier	0	1500



(a)

(b)

Figure 26: Detection of the pistol: (a) manual method and (b) proposed method.

Table 22: Confusion matrix based on the pistol.

		Predicted label	
		OoI	Outlier
Actual label	OoI	9000	0
	Outlier	0	1980

Table 23: Experimental results of the proposed effective framework.

Sr. no.	Object type	Scenarios	Video duration	Summarized duration	Saved time (%)	*P* (%)	*R* (%)	*F*1-score (%)	Accuracy (%)
1	Aeroplane	Airport	24 sec.	6 sec.	75%	100	100	100	100
2	Car	Street video	3.06 min.	2 sec.	99.4%	100	100	100	100
3	Person	Car parking	5.16 min.	16 sec.	94.9%	92	100	95.8	99
4	Mobile	Mobile snatching	2.49 min.	11 sec.	93%	99	71	82.6	97
5	Bike	Bike snatching	30 sec.	5 sec.	83%	100	100	100	100
6	Pistol	Gun testing	3.03 min	2.30 min	81.9%	100	100	100	100
			Total time 15.09 min.	Total summarized time = 3.10 min.	Total saved time 87.86%				Overall accuracy 99.33%

proposed framework is 99.33%, and the total saved time is 87.86%.

4.4. Comparative Analysis. This section presents a comparative analysis of the proposed framework with the existing VS techniques. The comparative analysis is based on the following fundamental features:

(i) F1: customized object type (OoI)

(ii) F2: frame extraction based on object

(iii) F3: object detection accuracy

(iv) F4: summarization rate

Table 24 shows that most existing techniques generally perform object detection rather than focusing on the specific object (i.e., does not consider an object as an Object of Interest). Similarly, many techniques performed frame extraction by redundant frame elimination and scene elimination instead of focusing on the objects. The analysis shows that the proposed framework is unique and contains the most relevant features for VS. The uniqueness of our proposed framework is that it performs video summarizing based on objects of interest given to the system at the time of providing input. Added advantages of the proposed VS framework are simplicity and ease of understanding, with accuracy of 99.6, 99.9, and 99.3% and summarization rate of 82.8, 78.8, and 91.7% of three different datasets such as VSUMM, TVSum, and own dataset, which increases its efficiency as compared to other methods.

To further evaluate the performance of our proposed framework for VS, a comparative analysis between the proposed framework and other state-of-the-art VS techniques is performed as given in Table 25.

Table 24: Comparative analysis of the proposed VS effective framework with existing techniques.

Name	F1	F2	F3	F4
Srinivas et al. [21]	✗	✗	✗	1.8% improvement in (IFBFM)
Ma at el. [39]	✗	✗	✗	35%–48.28%
Wang and Ngo [20]	✗	✓	94%	50%
Lai at el. [45]	✗	✓	97%	—
Almeida et al. [30]	✗	✗	✗	75%
Varghese and Nair [38]	✗	✗	✗	55%
Chong-Wah Ngo et al. [34]	✗	✗	✗	10%–25%
Davila and Zanibbi [33]	✗	✓	96.28% recall	50%
Proposed model	✓	✓	99.9%	87.86%

Table 25: Comparative analysis of the proposed VS effective framework with state-of-the-art techniques.

Name	Precision	Recall	$F1$-score
FSM [20]	40.73	54.43	46.59
SVM [20]	49.7	71.2	58.53
HMM [20]	51.2	53.36	52.25
F-HHMM [20]	72.1	66.13	68.98
A-HHMM [20]	77.2	74.83	75.99
DT [39]	42.57	32.04	35.30
STIMO [39]	41.12	47.81	42.50
VSUMM [39]	50.43	45.34	46.51
MSRm [39]	38.48	50.19	41.85
MSRa [39]	40.03	52.05	43.56
SOMP [39]	41.83	55.02	45.33
AGDS [39]	41.35	58.40	46.27
CRmax [39]	44.21	55.17	47.32
CRavg [39]	44.94	56.44	48.28
DSNET on SumMe [65]	50.8	51.9	51.2
DSNET on TVSum [65]	61.9	61.9	61.9
HOMER [54]	—	—	46.9
DR-DSN [55]	—	—	41.4
ResNet-152 + GRU [55]	—	—	43.7
Proposed framework on VSUMM	99.38	88.0	93.34
Proposed framework on TVSum	99.72	100	99.85
Proposed framework on own dataset	98.5	95.16	96.40

5. Conclusion

This paper presents an effective VS framework that summarizes the video based on the OoI. The proposed framework is very effective, optimal, and performed much faster than other state-of-the-art methods for summarizing the video. The OoI-based solution makes it more reliable and flexible to generate the relevant video summary. YOLOv3 empowers the proposed framework to detect various objects efficiently and precisely. For validation of the proposed framework, extensive experiments are performed on three different datasets: the VSUMM dataset, the TVSum dataset, and the own dataset. The proposed VS framework has achieved an accuracy of 99.6% with high processing speed and overall saved time of 82.8% if full video is played to detect the OoI on the VSUMM dataset. Similarly, the accuracy of 99.9% with a summarization rate of 78.8% on the TVSum dataset is achieved. The accuracy of the own dataset is 99.3%, and the overall saved time is 87.86%. A desktop application is also developed that provides ease of use and customized object selection. In future, this work can be extended by enriching the dictionary and training the model for more OoI. It can be deployed in real-time environment to record summarized video for multiple nature of crime scenes.

Data Availability

The data used to support the study's findings are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] W. Ullah, A. Ullah, T. Hussain et al., "Artificial Intelligence of Things-assisted two-stream neural network for anomaly detection in surveillance Big Video Data," *Future Generation Computer Systems*, vol. 129, pp. 286–297, 2022.

[2] E. Cosgrove, "One billion surveillance cameras will be watching around the world in 2021, a new study says," 2019, https://www.cnbc.com/2019/12/06/one-billion-surveillance-cameras-will-be-watching-globally-in-2021.html.

[3] SecurityInfoWatch, "Data generated by new surveillance cameras to increase exponentially in the coming years," 2016, https://www.securityinfowatch.com/video-surveillance/news/12160483/data-generated-by-new-surveillance-cameras-to-increase-exponentially-in-the-coming-years.

[4] X. Yan, S. Z. Gilani, M. Feng, L. Zhang, H. Qin, and A. Mian, "Self-supervised learning to detect key frames in videos," *Sensors*, vol. 20, no. 23, p. 6941, 2020.

[5] B. Korbar, D. Tran, and L. Torresani, "Scsampler: sampling salient clips from video for efficient action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6232–6242, IEEE, Seoul, Korea (South), November 2020.

[6] J. Huo and T. L. van Zyl, "Unique faces recognition in videos," in *Proceedings of the 2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, pp. 1–7, IEEE, Rustenburg, South Africa, July 2020.

[7] S. Manna, S. Ghildiyal, and K. Bhimani, "Face recognition from video using deep learning," in *Proceedings of the 2020 5th International Conference on Communication and Electronics Systems (ICCES)*, pp. 1101–1106, IEEE, Coimbatore, India, June 2020.

[8] J. Zhang, S. Chen, S. Tian, W. Gong, G. Cai, and Y. Wang, "A crowd counting framework combining with crowd location," *Journal of Advanced Transportation*, vol. 2021, Article ID 6664281, 12 pages, 2021.

[9] S. A. Velastin, R. Fernández, J. E. Espinosa, and A. Bay, "Detecting, tracking and counting people getting on/off a metropolitan train using a standard video camera," *Sensors*, vol. 20, no. 21, Article ID 6251, 2020.

[10] K. Khan, R. U. Khan, W. Albattah et al., "Crowd counting using end-to-end semantic image segmentation," *Electronics*, vol. 10, no. 11, Article ID 1293, 2021.

[11] N. Mufti and S. A. A. Shah, "Automatic number plate Recognition: a detailed survey of relevant algorithms," *Sensors*, vol. 21, Article ID 3028, 2021.

[12] J. Shashirangana, H. Padmasiri, D. Meedeniya, and C. Perera, "Automated license plate recognition: a survey on methods and techniques," *IEEE Access*, vol. 9, pp. 11203–11225, 2020.

[13] M. Asif, M. Bin Ahmad, S. Mushtaq, K. Masood, T. Mahmood, and A. Ali Nagra, "Long multi-digit number recognition from images empowered by deep convolutional neural networks," *The Computer Journal*, vol. 117, 2021.

[14] S.-H. Zhong, J. Lin, J. Lu, A. Fares, and T. Ren, "Deep semantic and attentive network for unsupervised video summarization," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 18, no. 2, pp. 1–21, 2022.

[15] M. Furini, F. Geraci, M. Montangero, and M. Pellegrini, "STIMO: STIll and MOving video storyboard for the web scenario," *Multimedia Tools and Applications*, vol. 46, no. 1, pp. 47–69, 2010.

[16] Z. Elkhattabi, Y. Tabii, and A. Benkaddour, "Video summarization: techniques and applications," *International Journal of Computer and Information Engineering*, vol. 9, pp. 928–933, 2015.

[17] A. Bora and S. Sharma, "A review on video summarization approcahes: recent advances and directions," in *Proceedings of the 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pp. 601–606, IEEE, Greater Noida, India, October 2018.

[18] M. Tahir, I. A. Taj, P. A. Assuncao, and M. Asif, "Low complexity high efficiency coding of light fields using ensemble classifiers," *Journal of Visual Communication and Image Representation*, vol. 66, Article ID 102742, 2020.

[19] Q. G. K. Safi, T. Nawaz, S. M. A. Shah, and T. Mahmood, "Intelligent device independent ui adaption for heterogeneous ubiquitous environments," *IJCSNS*, vol. 11, p. 75, 2011.

[20] F. Wang and C.-W. Ngo, "Summarizing rushes videos by motion, object, and event understanding," *IEEE Transactions on Multimedia*, vol. 14, pp. 76–87, 2011.

[21] M. Srinivas, M. M. M. Pai, and R. M. Pai, "An improved algorithm for video summarization - a rank based approach," *Procedia Computer Science*, vol. 89, pp. 812–819, 2016.

[22] K. Kumar, D. D. Shrimankar, and N. Singh, "Event bagging: a novel event summarization approach in multiview surveillance videos," in *Proceedings of the 2017 International Conference on Innovations in Electronics, Signal Processing and Communication (IESC)*, pp. 106–111, IEEE, Shillong, India, April 2017.

[23] S. S. Thomas, S. Gupta, and V. K. Subramanian, "Perceptual video summarization—a new framework for video summarization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, pp. 1790–1802, 2016.

[24] F. Cricri, S. Mate, I. D. Curcio, and M. Gabbouj, "Salient event detection in basketball mobile videos," in *Proceedings of the 2014 IEEE International Symposium on Multimedia*, pp. 63–70, IEEE, Taichung, Taiwan, December 2014.

[25] M. Cote, F. Jean, A. B. Albu, and D. Capson, "Video summarization for remote invigilation of online exams," in *Proceedings of the 2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–9, IEEE, Lake Placid, NY, USA, March 2016.

[26] R. Agyeman, R. Muhammad, and G. S. Choi, "Soccer video summarization using deep learning," in *Proceedings of the 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 270–273, IEEE, San Jose, CA, USA, March 2019.

[27] E. Bulut and T. Capin, "Key frame extraction from motion capture data by curve saliency," in *Proceedings of the 20th International Conference on Computer Animation and Social Agents*, pp. 1822–185, CGS, Hasselt, Belgium, 2007.

[28] C. Li, Y.-T. Wu, S.-S. Yu, and T. Chen, "Motion-focusing key frame extraction and video summarization for lane surveillance system," in *Proceedings of the 2009 16th IEEE International Conference on Image Processing (ICIP)*, pp. 4329–4332, IEEE, Cairo, February 2010.

[29] M. Ajmal, M. Naseer, F. Ahmad, and A. Saleem, "Human motion trajectory analysis based video summarization," in *Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 550–555, IEEE, Cancun, Mexico, December 2017.

[30] J. Almeida, R. d. S. Torres, and N. J. Leite, "Rapid video summarization on compressed video," in *Proceedings of the 2010 IEEE International Symposium on Multimedia*, pp. 113–120, IEEE, Taichung, Taiwan, December 2010.

[31] Y. J. Lee, J. Ghosh, and K. Grauman, "Discovering important people and objects for egocentric video summarization," in *Proceedings of the 2012 IEEE conference on computer vision and pattern recognition*, pp. 1346–1353, IEEE, Providence, RI, USA, June 2012.

[32] A. Rav-Acha, Y. Pritch, and S. Peleg, "Making a long video short: dynamic video synopsis," in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, pp. 435–441, IEEE, New York, NY, USA, June 2006.

[33] K. Davila and R. Zanibbi, "Whiteboard video summarization via spatio-temporal conflict minimization," in *Proceedings of the 2017 14th IAPR International conference on document analysis and recognition (ICDAR)*, pp. 355–362, IEEE, Kyoto, Japan, November 2017.

[34] C.-W. Chong-Wah Ngo, Y.-F. Yu-Fei Ma, and H.-J. Hong-Jiang Zhang, "Video summarization and scene detection by graph modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 296–305, 2005.

[35] U. Damnjanovic, V. Fernandez, E. Izquierdo, and J. M. Martinez, "Event detection and clustering for surveillance video summarization," in *Proceedings of the 2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services*, pp. 63–66, IEEE, Klagenfurt, Austria, May 2008.

[36] S. Jai-Andaloussi, A. Mohamed, N. Madrane, and A. Sekkaki, "Soccer video summarization using video content analysis and social media streams," in *Proceedings of the 2014 IEEE/ACM International Symposium on Big Data Computing*, pp. 1–7, IEEE, London, UK, December 2014.

[37] T. Hussain, K. Muhammad, W. Ding, J. Lloret, S. W. Baik, and V. H. C. de Albuquerque, "A comprehensive survey of multi-view video summarization," *Pattern Recognition*, vol. 109, Article ID 107567, 2021.

[38] J. Varghese and K. R. Nair, "An algorithmic approach for general video summarization," in *Proceedings of the 015 Fifth International Conference on Advances in Computing and Communications (ICACC)*, pp. 7–11, IEEE, Kochi, India, September 2015.

[39] M. Ma, S. Mei, S. Wan, Z. Wang, and D. D. Feng, "Robust video summarization using collaborative representation of adjacent frames," *Multimedia Tools and Applications*, vol. 78, no. 20, pp. 28985–29005, 2019.

[40] M. Miniakhmetova and M. Zymbler, "An approach to personalized video summarization based on user preferences analysis," in *Proceedings of the 2015 9th International Conference on Application of Information and Communication Technologies (AICT)*, pp. 153–155, IEEE, Rostov on Don, Russia, October 2015.

[41] H. B. U. Haq, M. Asif, and M. B. Ahmad, "Video summarization techniques: a review," *International Journal of Scientific & Technology Research*, vol. 9, pp. 146–153, 2020.

[42] S. Uchihachi, J. T. Foote, and L. Wilcox, *Automatic Video Summarization Using a Measure of Shot Importance and a Frame-Packing Method*, Google Patents, 2003.

[43] Z. Lu and K. Grauman, "Story-driven summarization for egocentric video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2714–2721, IEEE, Portland, OR, USA, June 2013.

[44] Y. Jiang, K. Cui, B. Peng, and C. Xu, "Comprehensive Video Understanding: video summarization with content-based video recommender design," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, October 2019.

[45] P. K. Lai, M. Décombas, K. Moutet, and R. Laganiere, "Video summarization of surveillance cameras," in *Proceedings of the 2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 286–294, IEEE, Colorado Springs, CO, USA, August 2016.

[46] K. Peker and F. Bashir, *Content-based Video Summarization Using Spectral Clustering*, ResearchGate, Berlin, Germany, 2007.

[47] V. T. Chasanis, A. C. Likas, and N. P. Galatsanos, "Scene detection in videos using shot clustering and sequence alignment," *IEEE Transactions on Multimedia*, vol. 11, pp. 89–100, 2008.

[48] W. Sabbar, A. Chergui, and A. Bekkhoucha, "Video summarization using shot segmentation and local motion estimation," in *Proceedings of the Second International Conference on the Innovative Computing Technology (INTECH 2012)*, pp. 190–193, IEEE, Casablanca, Morocco, September 2012.

[49] G. Evangelopoulos, K. Rapantzikos, A. Potamianos, P. Maragos, A. Zlatintsi, and Y. Avrithis, "Movie summarization based on audio visual saliency detection," in *Proceedigs of the 2008 15th IEEE International Conference on Image Processing*, pp. 2528–2531, IEEE, San Diego, CA, USA, December 2008.

[50] S. E. F. De Avila, A. P. B. Lopes, A. da Luz Jr, and A. de Albuquerque Araújo, "VSUMM: a mechanism designed to produce static video summaries and a novel evaluation method," *Pattern Recognition Letters*, vol. 32, no. 1, pp. 56–68, 2011.

[51] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "Tvsum: summarizing web videos using titles," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5179–5187, IEEE, Boston, MA, USA, June 2015.

[52] Y.-F. Ma and H.-J. Zhang, "A model of motion attention for video skimming," in *Proceedings of the International Conference on Image Processing*, IEEE, Rochester, NY, USA, September 2002.

[53] M. Sridevi and M. Kharde, "Video summarization using highlight detection and pairwise deep ranking model," *Procedia Computer Science*, vol. 167, pp. 1839–1848, 2020.

[54] H. Meyer, P. Wei, and X. Jiang, "Intelligent video highlights generation with front-camera emotion sensing," *Sensors*, vol. 21, no. 4, Article ID 1035, 2021.

[55] M. S. Afzal and M. A. Tahir, "Reinforcement learning based video summarization with combination of ResNet and gated recurrent unit," in *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2021)*, pp. 261–268, SciTePress, Karachi, Pakistan, February 2021.

[56] P. Gunawardena, H. Sudarshana, O. Amila, R. Nawaratne, D. Alahakoon, and A. S. Perera, "Interest-oriented video summarization with keyframe extraction," in *Proceedings of the 2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer)*, pp. 1–8, IEEE, Colombo, Sri Lanka, September 2019.

[57] J. Meng, H. Wang, J. Yuan, and Y.-P. Tan, "From keyframes to key objects: video summarization by representative object proposal selection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1039–1048, IEEE, Las Vegas, NV, USA, December 2016.

[58] Z. Fataliyev, D. Han, Y. Imamverdiyev, and H. Ko, "Video summarization based on extracted key position of spotted objects," in *Proceedings of the 2015 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 331-332, IEEE, Las Vegas, NV, USA, March 2015.

[59] T.-Y. Lin, M. Maire, S. Belongie et al., "Microsoft coco: common objects in context," in *Proceedings of the Computer Vision - ECCV 2014*, pp. 740–755, Springer link, New Yark, NY, USA, April 2014.

[60] Roboflow, "Pistols dataset," 2020, https://public.roboflow.com/object-detection/pistols.

[61] A. Kathuria, "What's new in YOLO v3? Towards Data Science," 2018, https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b.

[62] N. Chauhan, "Yolo object detection made easy," 2020, https://medium.com/analytics-vidhya/yolo-object-detection-made-easy-7b17cc3e782f.
[63] J. Hui, "Object detection: speed and accuracy comparison (Faster R-CNN, R-FCN, SSD, FPN, RetinaNet and YOLOv3)," 2018, https://jonathan-hui.medium.com/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359.
[64] J. Redmon and A. Farhadi, "Yolov3: an incremental improvement," 2018, https://arxiv.org/abs/1804.02767.
[65] W. Zhu, J. Lu, J. Li, and J. Zhou, "DSNet: a flexible detect-to-summarize network for video summarization," *IEEE Transactions on Image Processing*, vol. 30, pp. 948–962, 2020.