2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC) | 979-8-3503-0623-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/MIUCC58832.2023.10278378

Object-Based Video Archive Summarization

Habiba Yasser Adel
Media Engineering and Technology
German University in Cairo
Cairo, Egypt
habiba.badie@student.guc.edu.eg

Ramez M. Elmasry
Media Engineering and Technology
German University in Cairo
Cairo, Egypt
ramez.ibrahim@guc.edu.eg

Mohammed A.-M.Salem
Media Engineering and Technology
German University in Cairo
Cairo, Egypt
mohammed.salem@guc.edu.eg

***Abstract*—Nowadays, footage from surveillance cameras has a huge amount of data and is very long and exhausting to watch. This is where our problem lies, as the need to efficiently explore these videos quickly is rising. So video summarization aids people in exploring videos efficiently by capturing the most important frames in a video. In this paper, we provide a system that summarizes long video footage into short videos displaying the object of interest to the user. The results are very promising, as the system shows the original video where the objects are being tracked, and the customized video shows the summarized movement where the object of interest is the only thing in the video tracked in all the frames of the original video and last is the summarized video which is after extracting the keyframes from the generated customized video, and this video only shows the essential parts of the original video related to this object. There is a percentage of error due to the usage of the Yolo model and the tracking algorithm which are not very accurate as every system has a percentage of error, for static object videos, the duration decreased till it reached 0 seconds, and the compression ratio is 0.005. For nonstatic object videos, the duration on average decreased the half and the size decreased by more than half of the original size. Regarding the summarized video, the duration, size, and compression ratio compared to the customized video did not change much but there was a huge difference if compared to the original video.**

***Index Terms*—Video Summarization, Input Query, Object Detection, Multiple Object Tracking, Computer Vision**

I. Introduction

Video data is now essential in our daily life. Most of the raw videos as camera footage and surveillance videos are too long and contain redundant content. Consequently, the amount of video data people have to watch is overwhelming, which is very exhausting, especially if it is a security matter. This raises new challenges in efficiently exploring both within and across videos. Video summarization helps people explore a video efficiently by capturing the essence of the video. Learning what is essential depends on the information needed by the user.

In this paper, we want to build a system that summarizes long video footage into short and specific videos based on the query that the user enters, to generate a video that summarizes the movement of the object needed by the user to be tracked. The method consists of taking from the user the query which will be compared by the pre-trained model used to the coco object classes and if the query matches one of the classes then this class will be detected and tracked based on its id and an output video will be generated to solely represent the movements of this object throughout the video.

The paper is organized as follows: The related work is described in section II. The methodology is written in section III. The results are analyzed in section IV. Finally, the conclusion and the future work are discussed in Section V.

II. Related work

Regarding the literature review, several approaches were discussed to accomplish various systems similar to the system being implemented in this research. One of the approaches being used by several papers is Convolutional neural networks to track moving objects. Also, another approach was discussed named Deep neural networks to segment videos and calculate deep features. Moreover, there are papers that discussed a Deep summarization network that predicts the probability of selecting each frame as an approach. A widespread approach was using the Query relevance technique. Several authors applied the Convolutional hierarchical attention network that encodes visual information technique [1]–[10], and [12]–[14].

A. Convolutional neural network

CNNs [25] are commonly applied to track objects in video sequences. This process typically involves several steps: Data Preparation, Choosing Network Architecture such as VGGNet, ResNet, or specialized models like GOTURN or DeepSORT.Training by taking a video frame as input and outputs a bounding box around the tracked object. The training process includes a loss function that penalizes differences between predicted and actual bounding boxes. Once the CNN is trained, it can be employed for tracking in new videos.

In each frame, the frame is passed through the CNN to predict the object's bounding box. Many tracking algorithms use CNN's bounding box prediction to update the tracker's internal state, including the object's position, scale, and appearance model. To address difficulties like object occlusion or loss in the frame, advanced techniques like Kalman filtering or particle filtering may be combined with the CNN-based tracker.

CNN-based object tracking methods have demonstrated varying performance based on factors like architecture choice, data quality and quantity, and tracking complexity. They excel

979-8-3503-0623-1/23/$31.00 ©2023 IEEE

in handling changes in object appearance, scale, and pose compared to traditional tracking methods.

However, they also exhibit some limitations such as Computational Intensity, Data Collection: Gathering high-quality labeled training data can be time-consuming and expensive and may not encompass all tracking scenarios, Overfitting: CNNs may overfit to training data, resulting in limited generalization in real-world scenarios.

Techniques like data augmentation and regularization are used to mitigate this issue, Tracking Challenges: CNN-based trackers can still struggle in demanding situations such as rapid object movement, substantial occlusion, or significant changes in object appearance and Robustness

In summary, while CNN-based object tracking in videos has shown promise and achieved impressive results, it also presents computational challenges and requires careful dataset curation and model refinement for robust performance across various tracking scenarios.

B. Deep Neural Network

Deep neural networks, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), are used to segment videos and extract deep features. The process involves Data Preparation: Collect labeled video frames with segmentation masks. Architecture Selection, Processing video frames through the trained network to obtain segmentation masks. For RNNs, consider temporal processing. Feature Calculation: Extracting deep features from segmented regions for further analysis.

Deep networks offer accurate segmentation, capturing complex object boundaries. RNN-based approaches maintain temporal consistency and improve video analysis. However, some drawbacks are considered as High computational requirements for training and inference, Demands for large labeled datasets, Overfitting risk, especially with limited data, Complex model development, and Interpretability challenges with deep features.

In conclusion, deep neural networks excel in video segmentation and feature extraction but pose computational and data challenges.

C. Deep summarization network

DSNs are systems designed to create video summaries by determining the likelihood of selecting each frame or segment in a video. First Data Preparation is done by gathering a dataset of videos with associated human-made summaries or highlights.

Then comes the architectural choice: DSNs typically use Recurrent Neural Networks (RNNs) or Transformer-based models to process video frames or segments. Then during training, DSNs predict the probability of selecting each frame/segment in a video to create a summary. They are guided by a loss function to produce summaries that resemble human-generated ones. In the summarization process, DSNs evaluate each frame/segment and decide which ones to include based on predicted probabilities.

DSNs can generate concise and meaningful video summaries that capture the video's essential content. This method encountered some limitations: Development complexity and a lack of interpretability, Summary quality may vary depending on data and specific tasks, and Overfitting to training data can be an issue.

In essence, DSNs offer flexibility in summary length and automation but face challenges regarding data, complexity, interpretability, subjectivity, and overfitting.

D. Query relevance

Query relevance methods [22] in video summarization aim to produce video summaries that are closely aligned with specific user queries. This method includes: Data Preparation: Gathering videos with associated textual information, like captions or descriptions.Query Processing: Analyzing and understanding user queries using natural language techniques.Relevance Scoring: Evaluating video segments based on their textual content and how well they match the user's query. Include segments with higher relevance scores in the summary. Summary Generation: Form a concise video summary using the most relevant segments.

This method creates summaries that specifically address user queries, ensuring relevance to the user's interests. By focusing on relevant segments, they produce shorter and more to-the-point video summaries, saving time for viewers. Users can input different queries to get summaries on various aspects of the same video content.

The method used in this research is based on the user query that is entered into the system, object detection and tracking are applied using Yolov8 and the centroid tracking algorithm. Also, keyframe extraction was applied by using the histogram similarity measure algorithm and background extraction was implemented by calculating the median frame. Our system does video summarization based on a certain object using Yolov8 (a pre-trained model used for object detection).On the other hand, some of the techniques mentioned above use complex networks to summarize videos by adapting them to detect objects not directly by using Yolo.

III. Methodology

A. Methodology Overview

The block diagram shown in figure 1 shows an input video entered into the system and object detection is done on the video, detecting only what type of object the user enters as a query using the pre-trained model Yolov8, then tracking is applied to assign each object of the same class related to coco classes a unique id, then we separate the background of the video from the foreground to display on it the cropped object of interest. Then we select the keyframes from the generated customized video to output another shorter and more summarized video for the wanted object.

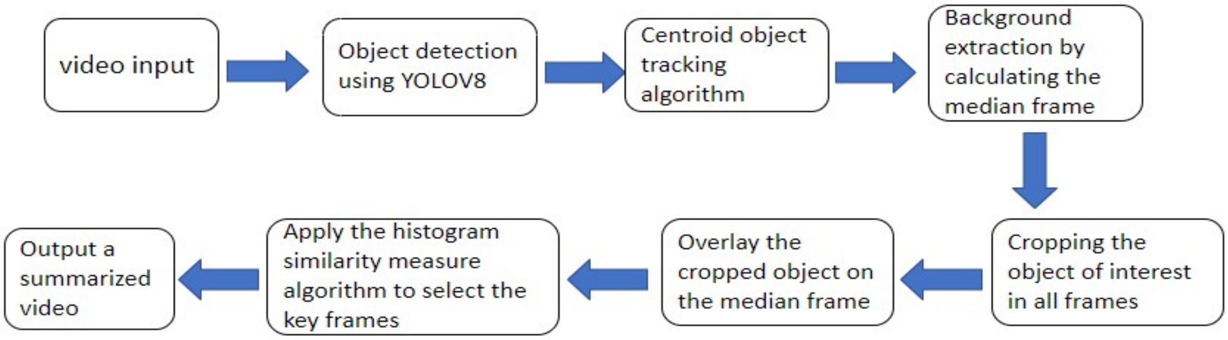

video input
Object detection using YOLOV8
Centroid object tracking algorithm
Background extraction by calculating the median frame
Cropping the object of interest in all frames
Overlay the cropped object on the median frame
Apply the histogram similarity measure algorithm to select the key frames
Output a summarized video

Fig. 1: Video Surveillance System Diagram

B. Object Detection

Object detection is applied by using Yolov8 [15] which is a pre-trained model on the coco dataset that contains 80 object classes that it can detect. Once the object detection is applied a video is out, which is the original video input but with a bounding box on each object detected, displaying the name of the class this object is related to and the confidence score which means how much we are sure that this is an object of a certain class.

C. Object Tracking

An object tracking algorithm [16] is applied to this video to output the same video but with an extra unique id displayed on the bounding box for each object. The tracking algorithm used is the centroid tracker [18], which first; calculates the center of each bounding box using the bounding box coordinates then assigns an id to each bounding box, and lastly computes the Euclidean distance between every pair of centroids (two center points) meaning we assume that the same object will be moved the minimum distance compared to other centroids, which means the two pairs of centroids having minimum distance in subsequent frames are considered to be the same object that will have the same id. The Euclidean distance equation can be defined as shown below:

$$D[(x,y)(w,h)] = \sqrt{(x-w)^2 + (y-h)^2} \tag{1}$$

D. Background Modeling using Median Filter

Background extraction from the video is done by calculating the median frame [19] which is a four-step process: Pixel Comparison: For each pixel position in the frames, the algorithm collects the pixel values at that position across all frames. The algorithm calculates the median value for each pixel position by sorting and selecting the middle value from the set of pixel values across frames. This is done independently for each pixel position.

E. Object Cropping And Overlaying on Median Frame

Object cropping from a video involves the process of extracting specific objects or regions of interest (ROI) from a sequence of video frames. It allows you to isolate and focus on particular objects within the video for further analysis or processing. Cropping is applied by getting the coordinates of the bounding box in each frame for the same object and cropping it to be saved as images. So now we have cropped images saved by the name:(object id, frame number) which describes in which frame this same object occurred. Overlaying of these cropped images on the median frame is done, but the objects must be overlaid in the same positions along the frames to appear as it is in the original video also splitting on the name of the images of the cropped object was done to be able to overlay correctly and in order of the frames.

F. Histogram Similarity Measure Algorithm

The last part of the project is applying the histogram similarity measure [17] to extract the key frames from the generated video to finally have the last output of the system which is a summarized video of the object of interest that contains only the keyframes that appeared in it the object through the whole video. The histogram similarity algorithm first converts each frame image to a grayscale and calculates its histogram. The histogram of a gray image can be treated as a discrete function with a range of integer pixel intensities from 0 to L-1, The histogram of such function can be calculated as the probability of occurrence of a certain gray level as follows:

$$P(L_k) = \frac{n_k}{n} \tag{2}$$

where $P(L_k)$ is the probability of occurrence of a certain gray level, n_k is the number of pixel intensities in that distribution, and n is the total number of pixels. The histogram is then normalized as shown below:

$$F_n = \frac{\text{number of pixels with intensity n}}{\text{total number of pixels}} \tag{3}$$

where F_n is the normalized image, and n is between 0 and $L-1$

The histogram similarity between every two frames is calculated till all the frames are covered by calculating the intersection of both histograms, with the possible value of the intersection lying between 0 (no overlap) and 1 (identical distributions). The overlapping area can be defined as the sum of the minimum value of the two histograms as shown below:

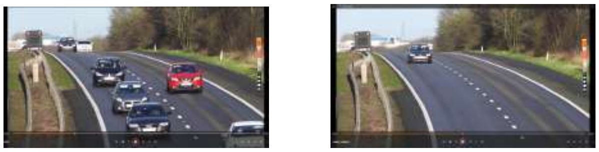
$$\text{Overlapping Area} = \sum_{i=0}^{b} \min(H_1(i),\, H_2(i)) \tag{4}$$

where b is the number of intervals or bins in which the range of pixel intensities is divided, $H_1(i)$, $H_2(i)$ are the histogram values for their respective images.

So now we have the similarity results; we can extract key frames of a video by reading the first and second frames and calculating their histogram and their similarity and if this similarity is greater than a certain threshold number, this means the frames are similar and as a result, only one frame will be considered a keyframe and the output video is generated.

IV. Results

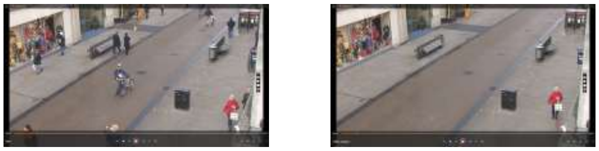
In this section, we talk about our findings. The following Figs. 2–4 shows a visual comparison between three original videos and their corresponding user-customized videos.



(a) Original Video Screenshot

(b) Customised Video Screenshot

Fig. 2: First Video Example:you can see that only the object of interest is the one visible while other objects were removed from the background

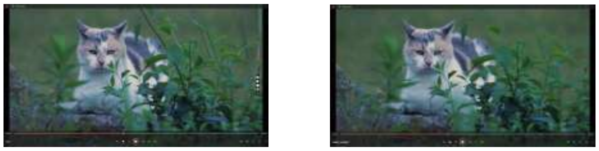


(a) Original Video Screenshot

(b) Customised Video Screenshot

Fig. 3: Second Video Example:the lady in red is the object of interest hence she is the only one visible in the customised video,removing any other object from the background

As shown in table I, we also compared the average duration of the original video with respect to the average duration of the customized video and the summarized video and we found that the average duration has drastically decreased between the original video and the customized video and remained the same comparing the customized video with the summarized video average duration because if the object is moving fast or changes position quickly then each frame will be considered as a key frame so almost the same duration.



(a) Original Video Screenshot

(b) customised video screenshot

Fig. 4: Third Video Example:this is a static cat video,so there is no to slight movement in the video hence the cat will be detected and displayed as it is.

The decrease in duration was more dominant in the static videos and in the videos that have a moving camera (where objects disappear and reappear again). Talking about the average size of the original video, it also decreased to be the new average size of the customized video and decreased a bit more in the summarized video, but it is only a slight decrease in size compared to the customized video's average size.

Moreover, we calculated the average compression ratio between the customized video and the original video and then we computed the compression ratio between the summarized video and the customized video. (compression ratio is calculating how much the size of the video has decreased or increased compared to the other video),It was found that the compression ratio of the customized video with respect to the original video is 0.5 which means that on average almost half the size is down.

To continue even more we computed the average compression ratio of the summarized video with respect to the customized video and it was 0.97 which means that the average size of the summarized video did not differ much from the customized video. There are results that are not 100 percent accurate which is due to the usage of the Yolo model and the centroid tracking algorithm as the tracking may have slight errors, not only that but also these results are for the average number of datasets so the results will vary if calculating the duration and size for each video in the dataset separately.

We have to take into consideration that these results are based on certain objects that the user entered as a query to be detected and also on a certain object ID that the user wants to track.

Figure 5 shows the best video results because it nearly decreased to half of the original size and during execution, it is the most accurate video in detecting and tracking objects, and the background is displayed nicely without blurring as some moving camera videos have a blurry background, the duration here did not change as in this video the object that a user wanted to detect was at the end of the video so the system kept on working without detecting anything till the object of interest appeared and it was detected and so the video result display the video with its duration but with less size.

The datasets used in this project were a set of 17 videos. All videos are used to test the system's functionality. Some videos are short and contain several moving objects as cars

TABLE I: Comparison between Original, Customized, and Summarized Videos

videos	Average Duration(seconds)	Average Size(MB)	Average Compression Ratio
Original Video	0:00:31	7.255	1
Customized Video	0:00:09	2.272	0.515
Summarized Video	0:00:09	2.188	0.970

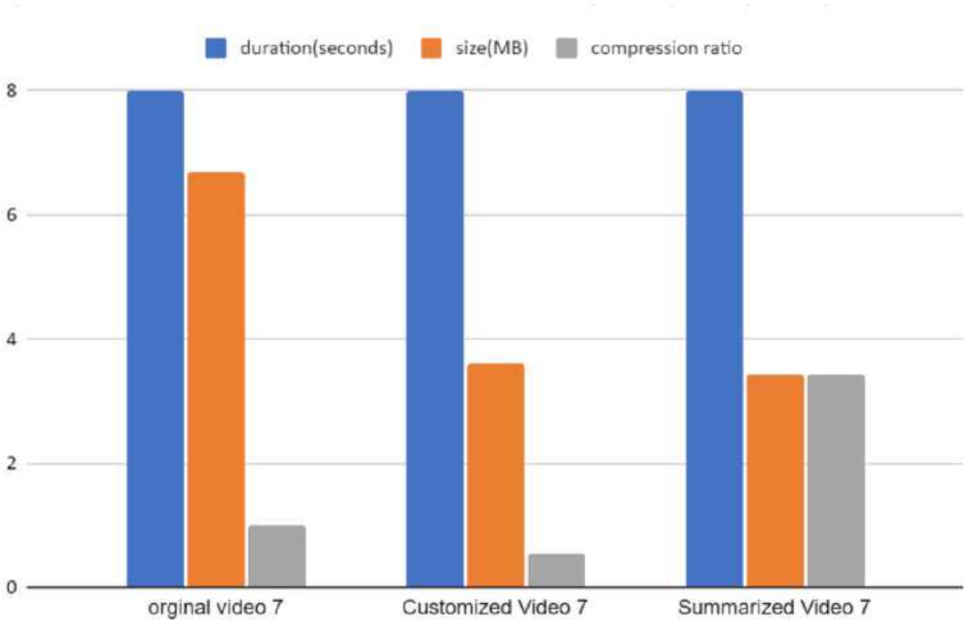


Fig. 5: System Best Video Results

and persons to be able to test object detection and tracking correctly. There are videos that contain static objects and these videos were used to test the system output which is different from other videos as the duration decreased to be 0 seconds and only 1 frame is displayed. Also, some videos were long, to see if the system would keep on working and processing as it should with long videos. The datasets were gathered randomly from several resources including YouTube and GitHub.

No other experiment was found comparing the size, duration, and compression ratio between the summarized videos and the original videos

V. Conclusion & Future Work

In this paper, we discussed a video archive summarization technique based on the user input query object. Furthermore, we discussed how the system was implemented using YOLOv8 and the centroid tracking algorithm. Also how we extracted the background of a video and calculated the median frame was discussed. Then we explained how the summarized video was obtained by the histogram similarity measure. The results of the system were discussed and most of the generated videos(customized/summarized) were accurate. The comparison between the original, customized, and summarized videos on average was pretty good.

Several limitations were encountered during the implementation of this project; the system cannot work on online streams, it can only work on previously recorded videos. For example, the recorded videos of the previous day in a company, not the live camera recording. Due to the Yolo model used, the bounding boxes surrounding an object may not be 100 percent accurate, as when cropping the object, the inside of the bounding box is being cropped which may result in a partial appearance of the object. By taking into consideration the limitations that were found in this research, different models may be used to have more accurate and better results.

References

[1] Huang, J.-H., and Worring, M. (2020). Query-controllable Video Summarization. arXiv. https://doi.org/10.48550/ARXIV.2004.03661

[2] Vasudevan, A. B., Gygli, M., Volokitin, A., and Van Gool, L. (2017). Query-adaptive Video Summarization via Quality-aware Relevance Estimation. arXiv. https://doi.org/10.48550/ARXIV.1705.00581

[3] Haq, H. B. U., Asif, M., Ahmad, M. B., Ashraf, R., and Mahmood, T. (2022). An Effective Video Summarization Framework Based on the Object of Interest Using Deep Learning.

[4] Wu, G., Lin, J., and Silva, C. T. (2021). IntentVizor: Towards Generic Query Guided Interactive Video Summarization. arXiv. https://doi.org/10.48550/ARXIV.2109.14834

[5] Kumar, K. (2021). Text query based summarized event searching interface system using deep learning over cloud. Multimedia Tools and Applications, 80, 1–16. https://doi.org/10.1007/s11042-020-10157-4

[6] Saif K. Jarallah, S. A. M. (2022). Query-Based Video Summarization System Based on Light Weight Deep Learning Model.

[7] Shoitan, R., Moussa, M. M., Gharghory, S. M., Elnemr, H. A., Cho, Y.-I., and Abdallah, M. S. (2023). User Preference-Based Video Synopsis Using Person Appearance and Motion Descriptions. Sensors, 23(3). https://doi.org/10.3390/s23031521

[8] Atif, O., Lee, J., Park, D., and Chung, Y. (2023). Behavior-Based Video Summarization System for Dog Health and Welfare Monitoring. Sensors, 23(6). https://doi.org/10.3390/s23062892

[9] Tahir, M., Qiao, Y., Kanwal, N., Lee, B., and Asghar, M. N. (2023). Privacy Preserved Video Summarization of Road Traffic Events for IoT Smart Cities. Cryptography, 7(1). https://doi.org/10.3390/cryptography7010007

[10] Moussa, M., and Shoitan, R. (2021). Object-based video synopsis approach using particle swarm optimization. Signal Image and Video Processing, 15. https://doi.org/10.1007/s11760-020-01794-1

[11] amit gupta. (2018). Introduction to Deep Learning: Part 1.

[12] Yan, X., Gilani, S. Z., Feng, M., Zhang, L., Qin, H., and Mian, A. (2020). Self-Supervised Learning to Detect Key Frames in Videos. Sensors, 20(23). https://doi.org/10.3390/s20236941

[13] Chamasemani, F. F., Affendey, L. S., Mustapha, N., and Khalid, F. (2015). A Study on Surveillance Video Abstraction Techniques.

[14] Guo, Y., Wang, X., Luo, H., Pu, H., Liu, Z., and Tan, J. (2022). Real-Time Multi-person Multi-camera Tracking Based on Improved Matching Cascade. In J.-F. Zhang, C.-M. Chen, S.-C. Chu, and R. Kountchev (Eds.), Advances in Intelligent Systems and Computing (pp. 199–209). Springer Nature Singapore.

[15] Buhl, N. (2023). YOLO models for Object Detection Explained [YOLOv8 Updated].

[16] Acharya, A. (2022). The Complete Guide to Object Tracking.

[17] Naser, E. F. (2021). Compare Between Histogram Similarity and Histogram Differencing For More Brief Key Frames Extraction from Video Stream. Journal of Physics: Conference Series, 1897(1), 012022. https://doi.org/10.1088/1742-6596/1897/1/012022

[18] Jaiswal, A. (2022). A tutorial on Centroid Tracker and Counter System.

[19] Hung, M.-H., and jeng-shyang pan. (2014). A Fast Algorithm of Temporal Median Filter for Background Subtraction.

[20] Forsyth, D. A., and Ponce, J. (2002). Computer vision: a modern approach. prentice hall professional technical reference.

[21] Zhang, Y., Kampffmeyer, M., Zhao, X., and Tan, M. (2019). Deep Reinforcement Learning for Query-Conditioned Video Summarization. Applied Sciences, 9(4). https://doi.org/10.3390/app9040750

[22] Xiao, S., Zhao, Z., Zhang, Z., Yan, X., and Yang, M. (2020). Convolutional Hierarchical Attention Network for Query-Focused Video Summarization. Proceedings of the AAAI Conference on Artificial Intelligence, 34(7), 12426–12433. https://doi.org/10.1609/aaai.v34i07.6929

[23] Otani, M., Nakashima, Y., Rahtu, E., Heikkilä, J., and Yokoya, N. (2016). Video Summarization using Deep Semantic Features. arXiv. https://doi.org/10.48550/ARXIV.1609.08758

[24] Jadon, S., and Jasim, M. (2020, October). Unsupervised video summarization framework using keyframe extraction and video skimming. 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA). https://doi.org/10.1109/iccca49541.2020.9250764

[25] Mujtaba, G., Malik, A., and Ryu, E.-S. (2022). LTC-SUM: Lightweight Client-Driven Personalized Video Summarization Framework Using 2D CNN . IEEE Access, 10, 103041–103055. https://doi.org/10.1109/access.2022.3209275

[26] Zhou, K., Qiao, Y., and Xiang, T. (2018). Deep Reinforcement Learning for Unsupervised Video Summarization with Diversity-Representativeness Reward. arXiv. https://doi.org/10.48550/ARXIV.1801.00054

[27] Manna, S., Ghildiyal, S., and Bhimani, K. (2020). Face recognition from video using deep learning. 2020 5th International Conference on Communication and Electronics Systems (ICCES), 1101–1106.

[28] Ji, Zhong, Su, Y., Qian, R., and Ma, J. (2010). Surveillance Video Summarization Based on Moving Object Detection and Trajectory Extraction. https://doi.org/10.1109/ICSPS.2010.5555504

[29] Ahmed, A., Kar, S., Dogra, D. P., Patnaik, R., Lee, S., seung Hee-Choi, and Kim, I. (2017). Video synopsis generation using spatio-temporal groups. 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 512–517.

[30] Xie, Y., Wang, M., Liu, X., and Wu, Y. (2017). Surveillance Video Synopsis in GIS. ISPRS International Journal of Geo-Information, 6, 333. https://doi.org/10.3390/ijgi6110333

[31] El-Masry, Mohammed; Fakhr, Mohamed Waleed; Salem, Mohammed A.-M.: 'Action recognition by discriminative EdgeBoxes', IET Computer Vision, 2018, 12, (4), p. 443-452, DOI: 10.1049/iet-cvi.2017.0335 IET Digital Library, https://digital-library.theiet.org/content/journals/10.1049/iet-cvi.2017.0335

[32] Ezzat, M. A., Abd El Ghany, M. A., Almotairi, S., and Salem, M. A.-M. (2021). Horizontal Review on Video Surveillance for Smart Cities: Edge Devices, Applications, Datasets, and Future Trends. Sensors, 21(9). https://doi.org/10.3390/s21093222

[33] Maryam Nabil Al-Berry, Mohammed A.-M. Salem, Hala Mousher Ebeid, Ashraf S. Husseino, Mhammed F. Tolba, "Fusing Directional Wavelet Local Binary Pattern and Moments for Human Action Recognition", IET Computer Vision, Volume 10, issue 2, March 2016. pp. 153-162. doi: 10.1049/iet-cvi.2015.0087.