International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

State-of-the-Art Techniques for Real-Time Video Segmentation: A Review

A. M. Bhugul-Rajurkar[1], Dr. V. S. Gulhane[2]

[1]Research Scholar, Sipna College of Engineering & Technology, Amravati, Maharashtra, India
Email id: *ashwinibhugul[at]gmail.com*

[2]Professor, Sipna College of Engineering & Technology, Amravati, Maharashtra, India
Email id: *vijaygulhane27[at]gmail.com*

Abstract: ***A key stage in video processing is video segmentation, which allows videos to be divided into frames for use in applications like video analysis, coding, and editing. Video segmentation can be done using multiple state-of-the-art traditional methods. However, the suitability of various approaches varies when it comes to real-time video processing. Because video data is dynamic, real-time video analysis faces numerous difficulties. The real-time identification and extraction of objects or regions of interest from a video stream has distinct problems since it requires great efficiency and low latency. This paper aims to address these problems by examining certain real-time video segmentation techniques. This study provides insight into the applicability of these strategies to real-time video settings by assessing the effectiveness and efficiency of these methods. We have done study of a few state-of-art video segmentation techniques presented in this paper. The review research that comes from this study will be very helpful to academics and practitioners in their search for the finest real-time video object detection and tracking systems.***

Keywords: Segmentation, Video Processing, Real Time

1. Introduction

Partitioning a video sequence into informative and coherent pieces is known as video segmentation [2], and it is an essential procedure in computer vision and video analysis. Usually, each segment corresponds to a particular area, item, or motion pattern within the video frames. Numerous applications, such as object tracking, scene comprehension, activity detection, video compression, and video editing, heavily rely on video segmentation. Video segmentation [2] works with the temporal dimension, taking into account the interactions and changes between successive frames, as opposed to image segmentation, which divides a single picture into various sections. The segmentation process becomes more difficult due to this temporal feature since objects may move, change appearance, or interact over time.

A necessary step in many complex video analysis jobs is video segmentation. For instance, precise segmentation makes it possible to continuously identify and track objects as they move across the video frames in object tracking. Segmenting films into informative pieces helps with activity recognition by assisting in distinguishing various activities or events occurring within the scene.

Deep learning and Convolutional neural networks (CNNs) have advanced significantly, notably improving accuracy and efficiency in the field of video segmentation [2]. The temporal adaptation of methods like semantic segmentation [11] and instance segmentation has made it feasible to segment objects and regions over several frames. Real-time video segmentation is of great interest to various applications, such as interactive systems, self-driving automobiles, and surveillance, since it focuses on obtaining segmentation conclusions rapidly.

Video segmentation [2] remains a difficult task despite great progress because of several elements such motion blur, occlusions, varying lighting conditions, and intricate object interactions. Researchers are always investigating novel algorithms and models. To overcome these difficulties and develop precise, reliable, and computationally efficient video segmentation methods.

This review study attempts to examine different real-time video segmentation [2] methods in this context. By looking at these approaches' advantages, disadvantages, and applicability in various situations. Our objective is to deepen our understanding of modern video segmentation [2] approaches and their practical applications in computer vision**.** Some of such techniques have been studied by us in this paper. Some techniques work very well on live or online videos. The video segmentation techniques such as semantic segmentation, block based method, and temporal segmentation are widely used techniques which give accurate output. There are many different techniques for segmenting videos; we have examined a few of the more well-known or frequently applied techniques. The methods are reviewed in the following chapter.

2. Comparative Study of Realtime Video Segmentation Techniques

Video segmentation is generally used to separate foreground from background. So that we can see objects clearly. There are many unique methods implemented by some researchers for video segmentation. But, for real time there are very few. In this study paper we are going to study a few of these techniques in detail and will compare those ones. Some techniques have been implemented using Deep Learning, Machine Learning, etc.

a) Semantic Image Segmentation With Deeplab [11]
As deep Convolutional neural networks (DCNNs) process input data and progress through layers, a notable challenge

arises in segmenting objects within images, particularly those of smaller scales [11]. This challenge arises due to the phenomenon where the input feature map gradually reduces in size as it traverses deeper into the network. Consequently, intricate details and nuances of smaller objects might be disregarded, leading to a potential loss of crucial information for accurate segmentation. DeepLabV3 stands as a pinnacle in the realm of deep learning architectures, specifically tailored to excel in the complex realm of semantic segmentation tasks. An evolution of its predecessors, DeepLabV1 and DeepLabV2, this state-of-the-art architecture emerged from the efforts of the Google Research team. Its inception dates back to 2017, and since then, It has found several uses in fields including autonomous driving, satellite image interpretation, and medical images analysis. One of the standout advantages that distinguishes DeepLabV3 from its counterparts in semantic segmentation and classification models is its exceptional accuracy in the realm of multi-scale segmentation. Multi-scale segmentation is an advanced method of examining an image at many scales in order to fully capture items of different sizes and complex shapes [11]. Atrous convolutions are a key component of DeepLabV3's approach to solving the challenging problem of multi-scale segmentation. This technological achievement, together with its range of applications, positions DeepLabV3 as a leading semantic segmentation model, changing the field of computer vision by facilitating extremely precise and thorough image analysis [11].

DeepLabV3's architecture is a very complex and well-designed framework that is especially optimized to perform exceptionally well in the field of semantic segmentation. It is a potent tool in this context since its main goal is to achieve accuracy in scenarios involving several scales. Deep Convolutional neural networks are the foundation of its architecture (CNNs) [4], and its distinctive feature is the application of atrous (dilated) convolutions, which provide the robust acquisition of contextual data at various scales without causing the model's complexity to rise [11].

A brief overview of the architecture of DeepLabV3 is given below:

Input Layer: The network accepts images, and depending on the specific implementation or intended use case, the size of the input image may be flexible.

Feature Extraction Backbone: DeepLabV3 employs a powerful feature extraction backbone network, often based on architectures like ResNet, MobileNet, or Xception. The primary function of this network is to capture hierarchical features from the input image, establishing a foundational basis for subsequent processing steps.

ASPP (Atrous Spatial Pyramid Pooling): This is the heart of DeepLabV3's architecture. ASPP employs atrous convolutions at multiple rates to capture context at different scales. This step addresses the challenge of multi-scale segmentation by allowing the network to understand objects of varying sizes. ASPP produces multi-scale feature representations [11].

Decoder: Following ASPP, a decoder network refines the multi-scale features obtained. It often employs techniques like bilinear upsampling or transposed convolutions to increase the spatial resolution of features.

Skip Connections: The crucial task of combining information from the encoder and decoder stages is accomplished via skip connections. Fine-grained features are crucially preserved by these linkages when upsampling takes place. As a result, the network performs better on tasks like object identification and picture segmentation by maintaining a more thorough grasp of the input image.

Final Prediction: The architecture concludes with a final prediction layer, where a pixel-wise softmax or sigmoid activation generates the segmented output, assigning class labels to each pixel.

Output: The final result of the network is a semantic segmentation map, in which the class or object that each pixel in the input picture represents is labeled. This map offers a pixel-by-pixel classification of the picture, enabling a thorough comprehension of the many items and their limits in the scene. The utilization of atrous convolutions in ASPP is a pivotal differentiator for DeepLabV3. Atrous convolutions modify the receptive field of filters by introducing gaps between filter values, allowing context to be captured at varying scales. This technique, combined with the feature extraction backbone, skip connections, and decoding mechanisms, enables DeepLabV3 to produce highly accurate and contextually rich segmentation results across a range of object scales.
DeepLabV3's versatility and accuracy make it a valuable tool across various domains and applications where precise semantic segmentation is crucial. It is useful in Medical Image Analysis, Autonomous Driving, Satellite Image Analysis, Agriculture, etc.

On a number of semantic image segmentation benchmarks, such as PASCAL VOC, PASCAL-Context, PASCAL-Person-Part, and Cityscapes, DeepLab has shown outstanding performance [11]. It has continuously produced cutting-edge outcomes, creating new standards for the precision of semantic picture segmentation.

b) Deep Learning Method-Temporal Video Scene Segmentation [1]

Video (temporal) segmentation is the division of a video stream into homogeneous, discontinuous groupings of successive frames based on predetermined criteria. In this technique, CNNs and RNNs are combined to create the network architecture for this work. RNNs use Long-Short Term Memory (LSTM) to do the multimodal fusion and learn the temporal correlations between video shots [30] From the features that are retrieved from each video input modality, CNNs may identify patterns [1]. Each ConvFeat, CSIFT, and MFCC extracted descriptor was fed into a CNN network in order to identify meaningful patterns among the input modalities. Every CNN produces a separate set of multimedia tools and applications. Due to its higher semantic complexity than In contrast to the 128-dimensional vectors that CNN's outputs for CSIFT and MFCC depict, ConvFeat's output is expressed as a 256-dimensional vector

[1]. Using CNNs for both intra-shot and temporal analysis, this deep learning [1] method extracts features using a combination of RNNs. The method extracts several attributes from an input video by utilizing a deep network architecture. Small activation filter sizes, inspired by VGGNet [1], are a feature of the basic Convolutional, temporal (1D) max-pooling [1], and ReLU layers [1] design that make up each CNN neural network. The temporal complexity of the low-level input features is gradually decreased by these stages.

The algorithm known as visual keyframe selection has been implemented by the author [1]. Only a tiny percentage of every video, which contains visual information, is used by this algorithm. The method receives HSV 8: 4: 4 [1] normalized histograms from the shot frames (H → [f0, f1,. . ., fn−1]) as input, and K must be the number of required keyframes that is chosen. [1]. The final keyframe selection is determined by comparing the previously chosen keyframes and determining the maximum value of similarity frames. The keyframe is added in the keyframe list if no similarity is found.

Algorithm ends with the list which contains selected keyframes which is the output for this algorithm. They have used per-trained 16-layered VGGNet network3 for each keyframe. To learn the patterns from extracted features of video they have used CNNs and as RNNs is based on LSTM this architecture uses a combination of both. The performance of this method is more accurate for some datasets.

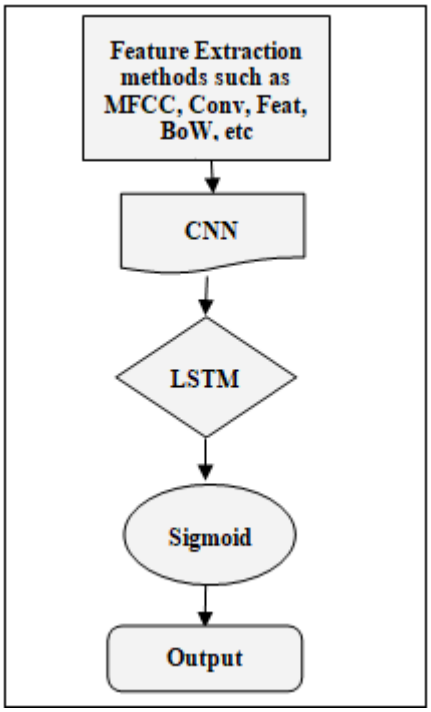

Figure 1: Video scene segmentation using deep-learning Network Architecture [1]

Figure 1 shows network architecture for the said method. As RNNs [1] use LSTM [1] the output is exact and accurate. This research method proposes a new multi modal approach that goes beyond only using RNNs to temporally segment [1] a movie into scenes. In contrast to similar methods using a publicly available video dataset, this approach carefully combines CNN and RNN capabilities to build a new architecture that yields better efficacy results on the task. We can use this technique during object detection to increase the efficiency of output as a preprocessing step in detection of objects in the images or video. The suggested method may better understand the temporal connections between features, which is helpful in identifying how the subjects are changing and producing a better scene segmentation.

c) Real Time Object Segmentation Using Swiftnet Technique [2]

The most advanced real-time video object segmentation (VOS) model available is called SwiftNet [2]. On the DAVIS [2] 2017 validation dataset, it obtains a J&F score of 77.8% and 70 FPS [2], outperforming all current solutions in terms of overall accuracy and speed performance [2]. Being a semi-supervised model, it can separate objects in a video sequence using just one tagged frame. SwiftNet is based on a matching-based VOS framework, but it introduces a number of optimizations to make it more efficient and accurate. This technique gives speed and accuracy as this uses a strong segmentation algorithm using VOS. It works on spatiotemporal redundancy and also helps to resolve problems with real time using Pixel adaptive memory. This method also uses Light-Aggregation Encoder (LAE) which gives good performance [2].

The Pixel-Adaptive Memory (PAM) module is a pivotal innovation within SwiftNet [2]. PAM introduces adaptability into the process of matching objects in both spatial and temporal dimensions, effectively reducing redundancy. Temporally, PAM focuses on updating memory only when objects exhibit significant changes between frames. Spatially, PAM is discerning, targeting memory updates and matching operations exclusively on dynamic pixels while disregarding static ones. This selective approach substantially diminishes unnecessary computations, resulting in remarkable enhancements in both the speed and precision of the segmentation process.

By looking for pixels that match the best, a spatiotemporal matching problem can be resolved. VOS is separated into several techniques, such as query matching and reference modeling [2]. The first frame, x1, of the video sequence V = [x1, x2, · · ·, xT], has mask y1 [2] annotated on it. One-shot video object separation (VOS) seeks to separate objects from the background by generating a mask for each frame t [2]. From VOS, mask computation is performed through object modeling and matching. First, SwiftNet is pre-trained using simulated data generated from the MS-COCO [2] dataset. The SwiftNet [2] is semi-supervised and it is for real time videos. SwiftNet is a video object segmentation model designed to process video sequences and produce segmentation masks for each frame, indicating which pixels belong to different objects. Here's an overview of how SwiftNet works:

Input and Encoding: Using a video sequence as input, SwiftNet creates feature maps by encoding the current frame as well as reference frames—previously seen frames kept in the Pixel-Adaptive Memory, or PAM module.

Matching Operation: SwiftNet compares the encrypted data using feature maps from the reference and current frames. This approach calculates a similarity score for each pixel in the current frame with respect to each pixel in the reference frames.

Segmentation Mask Generation: A segmentation mask for the current frame is then created using the calculated similarity scores. The item with the highest similarity score between that pixel and the reference frames is allocated to each pixel in the current frame.

Applications of SwiftNet include:

Video Surveillance: SwiftNet may be used in video surveillance to separate out objects of interest from the background of the footage, such as people and cars. This makes tracking and identifying objects easier. s

Robotics: SwiftNet's object segmentation capabilities are valuable for robots. It enables them to perceive and interact with their environment by segmenting objects, aiding in navigation and interaction tasks.

Augmented Reality: When virtual items or information need to be precisely positioned and interact with the actual environment, SwiftNet can be utilized in augmented reality applications. Object segmentation aids in finding items and surfaces that are appropriate for augmentation. With applications in computer vision, robotics, and augmented reality, among other domains, SwiftNet is a flexible tool due to its ability to conduct real-time object segmentation in films. This method achieves performance especially for real time videos. It works pixel wise. SwiftNet is effective and compatible with other techniques [2].

SwiftNet incorporates a light-aggregation encoder as another key feature, aimed at optimizing reference encoding for efficiency. This encoder leverages a reversed sub-pixel convolution technique to create a low-resolution reference frame. This low-resolution reference frame is specifically designed to be more efficient for matching operations, making the overall process faster and more resource-effective.

d) Semantic Segmentation, Object Detection And Human Pose Estimation [10]

Three core problems in computer vision, each addressing a distinct facet of visual perception, are object identification, semantic segmentation, and human position estimation. The process of identifying important body joints or landmarks in a picture or video, usually for humans, is known as "human pose estimation. " These landmarks include the locations of the head, hands, and feet as well as joints like the elbows, knees, and shoulders. Applications of this techniques may include:

- **Human-computer interaction:** Gesture recognition, sign language interpretation, and virtual reality.
- **Healthcare:** Monitoring patient movements for rehabilitation or medical diagnosis.
- **Sports Analytics:** Analyzing athletes' movements for training and performance assessment.
- **Semantic Segmentation:** By assigning a class name to every pixel in an image (such as "car, ""tree, " or "person"), semantic segmentation [10] divides the picture into areas that correspond to distinct item categories.
- **Autonomous vehicles**: Identifying objects and their locations on the road.
- **Remote sensing:** Urban planning, monitoring the environment, and classifying land cover.
- **Medical image analysis:** Tumour detection and organ segmentation in radiology.
- **Object Detection:** The process of locating and identifying things of interest inside an image or video frame is known as object detection. Bounding boxes are frequently used to indicate the positions of objects.
- **Surveillance**: Detecting intruders or suspicious human activities in security camera footage.
- **Retail:** Tracking and managing inventory by recognizing products on store shelves.
- **Autonomous robots:** Navigating and interacting with the environment by detecting obstacles and objects.

Each of these tasks typically requires the use of different models, techniques, and datasets, although there has been a trend in recent years to combine them into more comprehensive frameworks for holistic scene understanding HRNet, short for High-Resolution Network, has indeed emerged as a state-of-the-art algorithm with remarkable achievements in various computer vision tasks, especially in face landmark identification, semantic segmentation, and posture estimation of humans. Its outstanding performance has been shown on several benchmark datasets, highlighting how well it handles the difficulties associated with high-resolution picture processing. [10].

In semantic segmentation tasks, HRNet has showcased its prowess on prominent datasets such as PASCAL Context, LIP, and Cityscapes. Its ability to maintain high-Because of its ability to capture fine-grained information using resolution feature maps, it is a good choice for applications requiring accurate pixel-level labeling [10].

Additionally, HRNet has significantly advanced the domains of human position estimation and face landmark identification. Its architecture's ability to process high-resolution inputs is crucial for precisely locating face landmarks and determining the postures of people in photos. Its higher performance in these domains has been demonstrated on datasets such as AFLW, COFW, and 300W [10].

HRNet is pushing the boundaries of computer vision technology, its impact reverberates across a spectrum of applications, from autonomous driving to medical image analysis, further cementing its reputation as a pivotal algorithm in the world of visual recognition and understanding [10].

The process of semantic segmentation with HRNet involves several key steps. HRNet's unique architecture, which maintains high-resolution feature maps throughout the network, contributes to its effectiveness in capturing fine-grained details. Here's a simplified overview of the segmentation process using HRNet:

- **Input Image:** Begin with the input image, which is typically of high resolution. HRNet is designed to handle high-resolution inputs effectively.
- **Feature Extraction:** The first several layers of the HRNet architecture process the input picture and extract feature maps from it. HRNet maintains numerous concurrent streams of feature maps at various resolutions, in contrast to standard CNNs that downsample the feature maps early on to save computational strain.
- **Multi-Resolution Fusion:** One of the key innovations in HRNet is the multi-resolution fusion mechanism. To build a feature representation that preserves context and fine-grained information, the parallel feature maps from various resolutions are combined in this stage. By combining high-resolution and low-resolution feature maps, this fusion is accomplished, enabling HRNet to efficiently gather both local and global data.
- **Semantic Segmentation Head:** The aggregated feature maps are run via a semantic segmentation head subsequent to multi-resolution fusion. This portion of the network classifies pixels in an image by giving each pixel a class label [11].
- **Output:** The final output of the HRNet-based segmentation model is a dense prediction map that labels each pixel in the input image with its corresponding class.
- **Post-processing:** Post-processing techniques like boundary refinement, conditional random fields (CRF), or other smoothing approaches may be used to increase the quality of the segmentation findings, depending on the particular application and needs.

HRNet's primary innovation is its capacity to preserve high-resolution feature maps across the network, which enables it to record minute details while also taking context into account. Because of this design decision, HRNet is especially well-suited for jobs requiring accurate object delineation, such as semantic segmentation [10].

It's important to note that the architecture and details of HRNet can vary based on specific versions and implementations, and researchers often fine-tune these models for optimal performance on different datasets and tasks. Additionally, the process may involve training the model on labelled data before it can accurately perform semantic segmentation on new, unseen images [10].

e) YOLACT-You Only Look at Coefficients [15]

It is a cutting-edge deep learning model created especially for segmenting instances in real time. Instance segmentation is the task of not only detecting and localizing objects in an image but also distinguishing individual object instances by assigning a unique label or identity to each instance. YOLACT's effectiveness and strong performance in this difficult computer vision job have drawn notice. On the MS COCO [15] dataset, this straightforward fully Convolutional model provides state-of-the-art results. Robotics, driverless cars, and object tracking are just a few of the real-world uses for YOLACT's speed and accuracy. To begin with, YOLACT is used to encode the input image into a set of feature maps. . Following that, the feature maps are sent to two branches operating in parallel: one for segmentation and the other for classification.

The classification branch predicts a set of bounding boxes and corresponding class labels for each object in the picture. The segmentation branch anticipates a set of mask coefficients for each item in the images [15].

Next, a segmentation mask for every item is created using the mask coefficients. The segmentation mask indicates which pixels in the image belong to which object. Because YOLACT is a fully Convolutional model, real-time performance is achievable. This implies that it may be used with a GPU, enabling quick parallel processing. Key features and characteristics of YOLACT include:

Efficiency: YOLACT is engineered for real-time or near-real-time applications, making it suitable for use cases where low-latency segmentation is critical.

Mask Prediction: YOLACT not only provides bounding boxes for object detection but also generates pixel-level masks that precisely outline the shape of each detected object instance.

Single Shot: YOLACT is a one-shot model that, unlike two-stage detectors, can complete object detection and instance segmentation [11] in a single forward run over the network. It accelerates as a result.

Linear Combinations: YOLACT employs linear combinations of features at different resolutions to capture both high-level and low-level visual information, improving segmentation accuracy.

Multi-Task Head: YOLACT uses a multi-task head that simultaneously predicts class scores, object bounding box coordinates, and object masks, allowing for efficient joint optimization of these tasks.

Panoptic Segmentation: YOLACT may also be expanded to panoptic segmentation, which is an instance segmentation job combined with semantic segmentation (giving each pixel a class label). Here are some specific examples of YOLACT's performance on different tasks:

On the COCO test-dev set, YOLACT achieves an mAP of 29.8% for instance segmentation, 34.3% for object detection, and 42.5% for bounding box detection [15].

On the Cityscapes test set, YOLACT achieves an mAP of 38.7% for instance segmentation and 82.1% for semantic segmentation [15].

YOLACT scores a mAP of 84.1% for instance segmentation on the PASCAL VOC test set [15]. Although YOLACT is currently in development, future improvements to its performance are anticipated.

f) Fast Video Object Segmentation Using Guided Co-Segmentation Network [5]

Large deformations, a lot of object changes, and severe occlusions are common problems with traditional VOS

approaches. Hence author [5] suggested the Guided Co-Segmentation Network (GCSeg) for quick and precise VOS in order to overcome these difficulties. The online video object segmentation serves as the foundation for this video segmentation method. Objects in live videos are segmented using a guided co-segmentation network. The technique of segmenting similar objects or regions shared by several photographs or video frames is commonly referred to as co-segmentation. "Guided" co-segmentation suggests using extra data or direction to increase the segmentation results' accuracy. The network is made up of many modules, including Reference and Co-segmentation modules.

In order to produce reliable and accurate VOS, GCSeg [15] is a lightweight and effective network that takes into account temporal inter-frame linkages that are short-, middle-, and long-term.

The probability of incorrect or erroneous results is decreased by the adaptive search approach. In the GCSeg network there are two parts: Reference Module which focuses on encoding the foreground regions and Co segmentation Module which extracts regions from the current frame. A frame sequence I = {I0, I1, · · ·, It · · · IT } represents a video [5]. Accurate findings are produced by the Co-segmentation Module, which creates more informative frames. Co-segmentation Module encodes connection between current and previous frames.

In the architecture of co-segmentation technique, the task at hand is to extract target items that are in close proximity to the ground [5]. The strategy used by the method is simple and obvious; it takes into account inter-frame interactions across the short, middle, and long terms [5].

The most advanced real-time VOS model available today, GCSeg, combines great speed and accuracy. It has several potential uses in robotics, augmented reality, and video surveillance. This method has shown excellent results on internet video datasets like Davis 2016 and 2017 [5].

3. Comparative Study

We have done the study of these techniques and here are some of the findings we have observed. These are based on our study only**.** Real-time observations show that all of the strategies function effectively, but they also have some drawbacks or restrictions. Certain techniques only yield positive outcomes when applied to particular datasets, while others are unable to identify finely defined image borders.

Table 1: Comparative Analysis in view of Performance and Disadvantages

Sr. No.	Methods	Performance	Findings/Disadvantages
1	Semantic Segmentation with Deeplab[11]	Good for datasets like PASCAL, VOC, etc.	1. Unable to precisely capture an object's delicate edges 2. When trained on limited data, over fitting may occur. 3. Dependency on Resources
2	Deep Learning Method-Temporal Video Scene Segmentation	Achieves 82% accuracy for single shotgun dataset.	1. It could be computationally costly to train. 2. It may require large annotated datasets for training. 3. The model is sensitive to noise.
3	Object Segmentation Using SwiftNet Technique[2]	Accuracy 81% on DAVIS dataset.	The prototype may be sensitive to picture disturbance and challenging to distinguish between objects that resemble background objects.
4	Semantic Segmentation , Object Detection And Human Pose Estimation[5]	Superior results for some datasets PASCAL, COFW,etc	challenges with compatibility, inference, integration, or training.
5	YOLACT-You Only Look At CoefficienTs[15]	Good Speed and accuracy as it is trained on GPU	It may respond to prototype masks insufficiently.
6	Fast Video Object Segmentation Using Guided Co-Segmentation Network [5]	Accuracy is based on dataset	Adding both modules may give more accurate results.

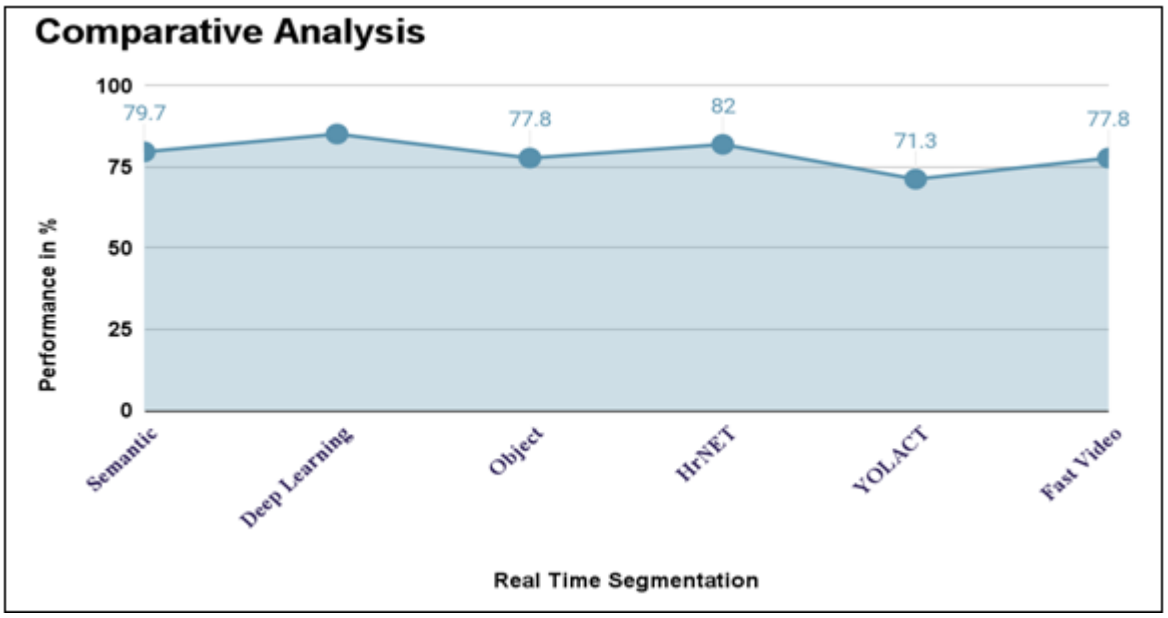

Figure 2: Graph for Comparative Analysis of real time video Segmentation Methods based on their used datasets

We have also investigated the benefits of the aforementioned methods. Semantic Segmentation with Deeplab gives high accuracy and it has a good ability to extract features at different scales due to its multi resolution approach. The use of dilated convolutions allows for large receptive fields without a significant increase in computation [20]. For precise scene boundary identification, deep learning models for temporal video scene segmentation must be able to capture temporal relationships and comprehend the context throughout time. This approach performs well when learning intricate patterns and representations from big datasets. Object Segmentation Using SwiftNet Technique is state of art technique that supports high processing speed with adaptability to various images. In order to extract features, YOLACT uses a lightweight backbone network, which improves its accuracy and real-time speed. One can also improve the performance of real time video object detection by building an hybrid model by using the mentioned techniques.

4. Conclusion

The foundational element of video processing is video segmentation, which is an essential per-processing stage. When considering real-time CCTV data analysis, where the use of advanced algorithms makes it possible to identify potentially suspicious things like knives and weapons, its importance becomes even more apparent. Furthermore, establishing temporal boundaries between frames improves the overall effectiveness of video segmentation in surveillance applications. The study paper has given a review of a few most widely used real-time video segmentation techniques, each of which stands out for its own methodology and impressive precision. Deep learning approaches, namely Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have become the most popular and regularly produce the best outcomes among these tactics. The use of deep learning techniques emphasizes how flexible and resilient these methods must be to address the intricacies present in video segmentation assignments. This review study provides researchers with the necessary information to assess and choose the most appropriate strategy based on a critical analysis of a wide range of innovative techniques. This review article provides an extensive toolkit for real-time video segmentation through the exploration and comprehensive use of a wide range of approaches. These studies provide researchers with an extensive tool set that they may use to evaluate a variety of innovative techniques, enabling them to choose and implement the most successful strategy. These results add to the continuous progress in video processing and provide a path map for researchers attempting to traverse the ever-changing segmentation methodology environment, so they may make well-informed judgments based on the most recent developments in the field.

References

[1] T. Trojahn, R. Goularte, 'Temporal video scene segmentation using deep-learning', Multimedia Tools and Applications, Volume 80, Issue 12, Springer, pp 17487–17513, May 2021.
[2] H. Wang, X. Jiang, H. . Ren, Y. Hu, et al, 'SwiftNet: Real-time Video Object Segmentation', Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp.1296-1305, IEEE Xplore, 2021.
[3] A. Dixit, 'Google Enables Mobile Real-time Video Segmentation in YouTube Stories', Youtube Stories, 2018.
[4] M. Kalezic, P. Sekulic, S. Kovacevic, 'Video Object Segmentation using Optical Flow and Recurrent Neural Networks', pp.1–4, IEEE 9th Mediterranean Conference on Embedded Computing (MECO), 2020.
[5] W. Liu, G. Lin, T. Zhang etal., 'Guided Co-Segmentation Network for Fast Video Object Segmentation', IEEE Transactions on Circuits and Systems for Video Technology, Volume: 31, Issue: 4, pp.1607-1617, July 2020.
[6] R. Jain, P. Jain, T. . Kumar et al., 'Real time video summarizing using image semantic segmentation for

CBVR', Journal of Real-Time Image Processing 18, pp.1827–1836, Springer 2021.
[7] R. Yao, G. Lin, S. Xia et al., 'Video object segmentation and tracking: A survey'. ACM Transactions on Intelligent Systems and Technology vol.11, Issue 4 pp.1-47, May 2020.
[8] A. Khoreva, F. Perazzi, R. Benenson et al., 'Learning video object segmentation from static images', IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), December 2016.
[9] F. Perazzi. 'Video Object Segmentation', PhD thesis, ETH Zurich, 2017.
[10] H. Marius, 'HRNet explained: Human Pose Estimation, Semantic Segmentation and Object Detection', Revealing what's behind the state-of-the art algorithm HRNet, Towards Data Science, Oct 2021.
[11] L. Chen, G. Papandreou, I. Kokkinos, et al., 'DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs', IEEE Transactions on Pattern Analysis and Machine Intelligence PP (99), June 2016.
[12] G. Li, Y. Xie, T. Wei et. al., 'Flow guided recurrent neural encoder for video salient object detection', IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 2018.
[13] S. Yurtkulu, Y. sahin, G. Unal, 'Semantic Segmentation with Extended DeepLabv3 Architecture', 27th Signal Processing and Communications Applications Conference (SIU), ISSN: 2165-0608, 2019.
[14] B. Daniel, Z. Chong, X. Fanyi, Y. Lee, 'YOLACT: Real-Time Instance Segmentation', 27th Signal Processing and Communications Applications Conference (SIU), 2019.
[15] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixe, D. Cremers, L. Van Gool et al., 'One-shot video object segmentation', Proc. Computer Vision and Pattern Recognition (CVPR), IEEE, pp.221-230, Jul.2017.
[16] H. Hu, S. . Lan, Y. Jiang, et al., 'FastMask: Segment multiscale object candidates in one shot'. In CVPR, IEEE, pp.991–999, 2017.
[17] H. Noh, S. Hong and B. Han, 'Learning deconvolution network for semantic segmentation', IEEE International Conference on Computer Vision (ICCV) Vis., pp.1520-1528, Dec.2015.
[18] L. Bao, B. Wu, W. Liu., 'Video object segmentation via inference in a cnn-based higher order spatio-temporal', In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5977–5986, 2018.
[19] S. Tsang, 'Review: DeepLabv3 — Atrous Convolution (Semantic Segmentation) ', Towards Data Science, Jan 2019.
[20] N. Remzan, K. Tahiry, A. Farchi, 'Automatic Classification Of Preprocessed Mri Brain Tumors Images Using Deep Convolutional Neural Network', International Journal on Technical and Physical Problems of Engineering" (IJTPE), Issue 54, Volume 1, pp.68-73, March 2023.
[21] A. Dhandayuthapani J. Lawrence, 'Plant Disease Recognition Using Optimized Image Segmentation Technique', International Journal on Technical and Physical Problems of Engineering" (IJTPE), Issue 50 Volume 14, pp.211-218, March 2022.