Video Object Segmentation using Adversarial Techniques

A THESIS

submitted by

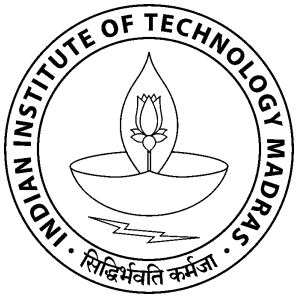
SAPTAKATHA ADAK

for the award of the degree

of

MASTER OF SCIENCE

(by Research)



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

NOVEMBER 2020

THESIS CERTIFICATE

This is to certify that the thesis titled **Video Object Segmentation using Adversarial Techniques**, submitted by **Saptakatha Adak**, to the Indian Institute of Technology, Madras, for the award of the degree of **Master of Science**, is a bona-fide record of the research work done by him under my supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Prof. Sukhendu Das
Research Guide
Professor
Dept. of Computer Science & Engg.
IIT Madras

Place: Chennai

Nov. 30, 2020.

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank my advisor Prof. Sukhendu Das, for his support throughout my stay at IIT Madras. His continuous encouragement and guidance has helped me to develop my skills as a researcher in the field of computer vision and to achieve several accolades and success in several fields in these fruitful years of my life.

I would also like to thank all the faculty members of the CS&E Department for providing a wonderful research environment. I am especially grateful towards my course instructors who have enhanced my knowledge in various aspects of the subjects. I would like to express my gratitude to all my General Test Committee members whose various comments about my research work have prompted me to revisit and improve my ideas. I am also thankful to all the CS&E department staffs for their continuous support.

The people who made my stay at IITM most memorable are my friends, hostel-inmates and colleagues, without whose love and support this journey would indeed have been a boring one. I would like to thank all the VP lab members, both past and present, for making the lab a wonderful place to work in. I would especially like to thank my fellow research scholars of the department for making my stay wonderful and exciting.

Finally and most importantly, I would like to thank my family their constant unconditional love, motivation and support. Without their dedication towards my education and well-being, this research work would not have been possible.

Above all, I thank the Almighty for His blessings throughout my life.

ABSTRACT

KEYWORDS: Video Object Segmentation; Generative Adversarial Networks (GAN); Patch-wise Symmetric Difference Loss (PSDL); Inter-frame Temporal Symmetric Difference Loss (ITSDL); Intra Frame Temporal Loss (IFTL); Graph Convolutional Neural Networks (GCNN); Motion based Aggregation; Occlusion aware Aggregation.

Video Object Segmentation (VOS) has recently emerged as a popular semi-supervised learning problem in the field of Computer Vision. It aims to segment objects in a video sequence under challenging situations such as change of object appearance, occlusions, camera view change, background clutter and motion blur. The popularity of this domain lies in the fact that it has profound impact in the fields of bio-medical imaging, self-driving cars, video editing, robotics, surveillance, video prediction, etc.

Although Convolutional Neural Networks (CNNs) have been used in the past for the purpose of foreground segmentation in videos, adversarial training based methods have not been explored thoroughly, in spite of its extensive use for solving many other problems in Computer Vision. We deal with the complex task of VOS under unconstrained environments, with an assumption that the object of interest is delineated in the first frame by the user.

The thesis first describes a GAN based framework along with the use of an Intersection-over-Union (IoU) score based Patch-wise Symmetric Difference Loss (PSDL) function to train the model, for solving the problem of foreground object segmentation in videos. This network in spite of independently processing the sequence of video frames is still able to maintain the temporal coherency between them, without the use of any explicit trajectory-based information. The proposed method, when evaluated on popular real-world video segmentation datasets viz. DAVIS-2016, SegTrack-v2 and YouTube-Objects, exhibits substantial performance gain over the recent state-of-the-art methods.

The second approach improves over the previously mentioned PSDL loss by using optical vectors which aid in capturing motion features between consecutive frames in videos and thus in turn enhances the segmentation quality. On the other hand, the Inter-frame Temporal Loss (IFTL) function along with its long-range variant captures the temporal information from the sequence of video frames. The incorporation of these temporal information-based objective functions thus stabilizes the training process to provide much improved segmentation over other state-of-the-art methods proposed earlier in literature.

Finally, we design a dual-channel Graph Convolutional Neural Network (GCNN) using the raw image and optical flow features to model the inter-pixel relationships in the video frames across space-time. Moreover, a direction oriented motion based aggregator mechanism is introduced which not only captures the variations in motion patterns involved with multiple objects moving in different directions with dissimilar speeds but also efficiently handles the dynamic (temporal) changes in appearance of the same in terms of pose and scale. A novel occlusion aware aggregation scheme is also incorporated to assist the model in segmenting objects under occlusion or re-identifying targets appearing in the frame after prolonged disappearance. Performance analysis on DAVIS-2016 and DAVIS-2017 datasets show the effectiveness of our proposed method in foreground segmentation of objects in videos over the existing state-of-the-art techniques. Control experiments done on CamVid road segmentation dataset shows the generalising capability of the model for scene segmentation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

Adam	Adaptive Moment Estimation
adv	Adversarial
BN	Batch Normalization
bce	Binary cross-entropy
cce	Categorical cross-entropy
CNN	Convolutional Neural Network
CRF	Conditional Random Field
CV	Computer Vision
FC	Fully connected layer
FF	Feature as input, feature as output
fps	Frames per second
FS	Feature as input, segmentation as output
GAN	Generative Adversarial Network
GCN	Graph Convolutional Network
GCNN	Graph Convolutional Neural Network
gdl	Gradient divergence loss
GT	Ground-Truth
IoU	Intersection-over-Union
IFTL	Intra Frame Temporal Loss
ITSD	Inter-frame Temporal Symmetric Difference
ITSDL	Inter-frame Temporal Symmetric Difference Loss
L-IFTL	Long-range Intra Frame Temporal Loss
mIoU	Mean Intersection-over-Union
MO	Moving Object categories
MRF	Markov Random Field
MSK	Mask-Track
OnAVOS	Online Adaptation of convolutional neural networks for VOS
OSVOS	One-Shot VOS
Pixel-GCN	Pixel Graph Convolutional Network
PSD	Patch-wise Symmetric Difference
PSDL	Patch-wise Symmetric Difference Loss
PReMVOS	Proposal-generation, Refinement and Merging for VOS
R-CNN	Region based CNN
ReLU	Rectified Linear Unit
RGB	Red Green Blue channel
SEG	Segmentation
TempSeg-GAN	Temporally aided Segmentation GAN
VidSeg-GAN	Video Segmentation GAN
VOS	Video Object Segmentation
VSS	Video Semantic Segmentation

w/	with
w/o	without
YTO	YouTube-Objects

LIST OF SYMBOLS

x	Input data
z	Noise variable
p_data	Data distribution
$p_{\mathbb{G}}$	Data distribution of the generator
p_z	Noise data distribution
$\mathbb{E}[\cdot]$	Expectation
$\mathbb{G}$	Generator network of GAN
$\mathbb{D}$	Discriminator network of GAN
$\mathbb{E}$	Encoder module
$\theta_{\mathbb{G}}$	Generator parameter set
$\theta_{\mathbb{D}}$	Discriminator parameter set
$\mathcal{O}_{\mathbb{G}}$	Generator output
$\mathcal{O}_{\mathbb{D}}$	Discriminator output
$\mathcal{O}_{\mathbb{E}}$	Encoder output
$\mathcal{I}$	Input RGB image
Y	Ground-Truth segmentation mask
$\hat{Y}$	Predicted segmentation mask
λ	Regularization hyper-parameter
W	Width of an image / segmentation mask
H	Height of an image / segmentation mask
c	Class instance
$\mathbb{C}$	Classes / Categories set
S	Width / Height of transformed image /mask used for computation
s	Width / Height of transformed image /mask patch used for computation
$\boldsymbol{y}$	Vector of ground-truth labels in Y
$\hat{\boldsymbol{y}}$	Vector of predicted labels in $\hat{Y}$
P	Ground-Truth segmentation mask patch
$\hat{P}$	Predicted segmentation mask patch
$\boldsymbol{p}$	Vector of ground-truth labels in patch P
$\hat{\boldsymbol{p}}$	Vector of predicted labels in patch $\hat{P}$
$\boldsymbol{M}_p$	Set of mislabelled pixels in a patch
$\lvert \cdot \rvert$	Cardinality of a set
W^*_{t-1}	Optical flow vector map patch at time instance $(t-1)$
P^*_t	Optical vector warped ground-truth mask patch at time instance t
$\boldsymbol{p^*}$	Vector of labels in patch P^*
δ	Penalizing constant
$d(\cdot,\cdot)$	Euclidean distance measure
$\overrightarrow{q}$	Vector formed of discriminator outputs
$\mathbb{H}$	Graph convolutional layer
f	Propagation function
V	Set of graph nodes

E	Set of graph edges
$\mathbb{N}$	Number of nodes in a graph
D	Dimension of input features of a graph node
F	Dimension of output features of a graph node
M	Number of consecutive frames that forms the graph
v_i'	Updated feature vector of i-th node of a graph
$\mathbb{S}$	Adjacency matrix in a graph
$\mathcal{N}_{(\cdot)}^{k,t}$	Set of neighbouring nodes of a graph node belonging to k-th spatial section at time instance t
$(\cdot)^T$	Transpose of a matrix / vector
$\phi(\cdot)$	Function to extract the graph node features
σ	Sigmoid function
$\mathcal{G}_{rgb}$	RGB image feature based pixel graph
$\mathcal{G}_{opt}$	Optical flow feature based pixel graph
$\mathcal{G}_{pix}$	Pixel graph (generalized notation)
$\mathbb{S}_{ij}$	Adjacency function between i-th node and j-th node
$\mathbb{W}$	Learnable transformation weight matrix
t	Time instance
$\mathbb{T}$	Current time-step
l	Layer instance
L	number of graph aggregation layers
$\mathbb{F}(\cdot,\cdot)$	Similarity estimation function
Γ^i	Set of adaptive weights for i-th graph node
γ_k^i	Adaptive weight for k-th spatial section of i-th graph node
$\|\|\cdot\|\|_t$	concatenation across time-steps
$\|\|\cdot\|\|_k$	concatenation across spatial sections
$\|\|\cdot\|\|_{k,t}$	concatenation across spatial sections and time-steps
V_{rgb}^L	RGB image features related to nodes in the L-th graph layer
V_{opt}^L	Optical flow features related to nodes in the L-th graph layer
$V_{combined}^L$	Combined features related to nodes in the L-th graph layer
d	Degree of a graph node
$\mathcal{A}_{motion}$	Motion based aggregation
$\mathcal{A}_{occ}$	Occlusion aware aggregation
$\mathcal{U}_{motion}^l$	Motion based update of l-th aggregation layer
$\mathcal{U}_{occ}^l$	Occlusion aware update of l-th aggregation layer
$\mathcal{L}_{L_1}$	Mean Absolute Error Loss / L1 Loss
$\mathcal{L}_{adv}^{\mathbb{G}}$	Generative adversarial loss
$\mathcal{L}_{adv}^{\mathbb{D}}$	Discriminative adversarial loss
$\mathcal{L}_{adv}$	Adversarial Loss
$\mathcal{L}_{bce}$	Binary Cross-entropy Loss
$\mathcal{L}_{cce}$	Categorical Cross-entropy Loss
$\mathcal{L}_{gdl}$	Gradient Divergence Loss
$\mathcal{L}_{IFTL}$	Intra Frame Temporal Loss
$\mathcal{L}_{ITSDL}$	Inter-frame Symmetric Difference Loss
$\mathcal{L}_{L-IFTL}$	Long-range Intra Frame Temporal Loss
$\mathcal{L}_{PSDL}$	Patch-wise Symmetric Difference Loss
$\mathcal{J}$	Jaccard Index / Region similarity
$\mathcal{J}_{mean}$	Mean of Jaccard Index / Region similarity
$\Delta\mathcal{J}_{mean}$	Difference of Jaccard Mean

$\mathcal{J}_{recall}$	Recall of Jaccard Index / Region similarity
$\mathcal{J}_{decay}$	Decay of Jaccard Index / Region similarity
$\mathcal{F}$	Contour accuracy
$\mathcal{F}_{mean}$	Mean of Contour accuracy
$\mathcal{F}_{recall}$	Recall of Contour accuracy
$\mathcal{F}_{decay}$	Decay of Contour accuracy
$\mathcal{T}$	Temporal (in-)stabilty
$\mathcal{T}_{mean}$	Mean of Temporal (in-)stabilty
$\mathcal{G}_{mean}$	Global mean

CHAPTER 1

Introduction

Segmentation of objects in videos has been an interesting yet challenging task that has gained importance in the field of Computer Vision and Machine Learning in the recent decade. Learning to segment objects in real-time in intelligent systems is essential for tracking and also to take decisions while planning its future course of actions. Apart from these, segmentation is also used as an additional component in many other tasks to provide the networks with contextual information. The popularity of Video Object Segmentation mainly lies in its profound impact in the domains of bio-medical research, autonomous systems, robotics, video editing, etc. Majority of the existing works for Video Object Segmentation (VOS) are based on object matching (Caelles *et al.*, 2017; Voigtlaender and Leibe, 2017), mask propagation (Perazzi *et al.*, 2017; Yang *et al.*, 2018) and object proposals (Li *et al.*, 2017; Luiten *et al.*, 2018), utilizing the power of Convolutional Neural Networks (CNNs) for implementation. But, none of them has used adversarial training techniques in spite of their effectiveness as a holistic approach in generative modelling.

In our work, the proposed architectures consist of a Generative Adversarial framework for foreground object segmentation in videos coupled with novel Intersection-over-union and temporal information based loss functions for training the network. For the last part of our research work, we have proposed a spatio-temporal graph convolutional network along with novel aggregation functions to segment multiple objects in videos having different motion patterns. Experimentations on the real-world benchmark datasets show the significant gain in performance of our models over the existing state-of-the-art methods both quantitatively as well as qualitatively.

1.1 Problem definition, Objective and Scope

In this thesis, the problem of Video Object Segmentation has been studied under a semi-supervised setup, where segmentation of foreground (moving) objects of interest

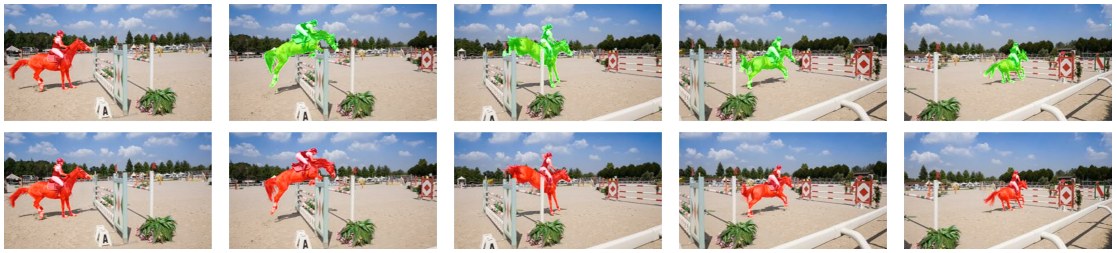


Figure 1.1: Illustration of Video Object Segmentation. First row: The ground-truth segmentation of the initial frame (red) assists the model to learn the particular object to track and segment in the remaining frames (green) of the video independently. Second row: The corresponding ground-truth masks (red) of the predicted segmented masks (green) (in first row). The frames at an equal interval of video sequence length have been shown (best viewed in color).

(either connected or individual) are performed in the consecutive video frames, given the ground-truth annotated mask for only the first (initial) frame of the video. The problem statement is pictorially explained in figure 1.1 where, the initial frame in the first row shows the ground-truth annotated mask (in red) used as input (along with remaining RGB video frames), followed by the results of network generated masks for the subsequent frames of the video (in green). The video frames in figure 1.1 are chosen at an identical interval of sequence length, only for the sake of illustration. The bottom row exhibits the corresponding ground-truth masks of the object(s) in the video frames, as given for the sample video in DAVIS-2016 dataset (Perazzi *et al.*, 2016).

With the recent advancement of deep learning techniques, there have been many works based on Convolutional Neural Networks (CNNs) which not only have improved the performance for problems like image classification (Krizhevsky *et al.*, 2012; Simonyan and Zisserman, 2014; He *et al.*, 2016), object detection (Ren *et al.*, 2015; Redmon *et al.*, 2016; Liu *et al.*, 2016), etc., but also in the field of image segmentation (Maninis *et al.*, 2016; Xie and Tu, 2015; Caelles *et al.*, 2017; Voigtlaender and Leibe, 2017), using pre-trained weights of image recognition models on ImageNet (Deng *et al.*, 2009). Video Object Segmentation is no exception from this. Most of the works done in this domain in the recent past, are based on object matching, mask propagation and object proposals, utilizing the power of deep CNNs. However, the major disadvantage of using deep architectures is their hunger for a large amount of training data. Recently, frameworks based on Generative Adversarial Networks (GAN) (Goodfellow *et al.*, 2014) has also become very popular for their holistic approach in generative modelling in tasks like image super resolution (Ledig *et al.*, 2017), face generation (Karras *et al.*,

2018), image translation (Isola *et al.*, 2017), etc. In spite of their effectiveness in image segmentation (Souly *et al.*, 2017; Luc *et al.*, 2016), none have used GAN for segmenting objects in video sequences.

In our work, to deal with the direct but a non-trivial problem of object segmentation in videos, we have used deep networks along with temporally aided objective functions for training the model under an adversarial setting. In the latter part of the work, a variant of Graph Convolutional network coupled with novel aggregator schemes has been proposed to solve the same problem in a unique manner.

1.2 Research Issues

Video Object Segmentation, being a recently emerged field of research in Computer Vision, has several challenges attached to it. The significant issues that are required to be resolved to generate feasible results have been addressed here:

- The common problem related to videos is motion blur, which occurs due to the rapid motion of objects and shaky camera motion. Apart from these, the other challenges are low image resolution, background clutter, occlusion, lack of association among objects, change of object appearance due to deformation, scale and pose variations.

- Temporal consistency between the long-term frames in the videos is a vital issue which often affects the model to produce sub-optimal outcomes.

- Temporal discontinuity is also another difficult problem in these field of research where objects undergo partial to full occlusion, object moving out of the frame followed by re-appearing in the video after prolonged disappearance.

- The real-world scenes often contain multiple objects moving in different directions with dissimilar speeds. Most of the existing methods fail in these types of difficult scenarios and produce normal motion patterns. Thus, one of the important challenges is to come up with a network that is robust to all the different motion patterns of various objects.

- To formulate suitable objective functions that may be implemented to train a GAN for the complex task of Video Object Segmentation.

1.3 Summary of Research Work

The thesis describes three novel methods adopted for solving the task of Video Object Segmentation (VOS). The three methods are presented in a chronological order, as developed by us to solve the same problem, where one latter method improves on the shortcomings of the former. The commonality is the use of GANs in all methods for training and testing under the assumption of semi-supervised settings.

The first work introduces a GAN based framework (VidSeg-GAN) to generate segmentation of objects of interest in videos. Along with this, an Intersection-over-Union score based Patch-wise Symmetric Difference Loss (PSDL) function is used for better training of the model. The network processes the sequence of video frames one at a time and maintains the temporal stability between them by modelling the motion features implicitly. Substantial performance gain over the state-of-the-art methods is observed while evaluating our proposed model on popular real-world VOS datasets viz. DAVIS-2016, SegTrack-v2 and YouTube-Objects.

Though the previous method generates improved results compared to the existing methods, it fails to produce a satisfactory performance in maintaining temporal consistency among the long-distant segmented masks due to the absence of explicit trajectory flow-based information. To overcome the drawback of the first method, a new technique has been introduced. The second approach extends the previous work by incorporating optical flow based temporal information in formulating two novel objective functions: (i) *Inter-frame Temporal Symmetric Difference Loss (ITSDL)*, and (ii) *Intra Frame Temporal Loss (IFTL)*. The former aids in capturing motion features between consecutive frames of videos and thus, in turn, enhances the segmentation quality of the predicted masks. Whereas, the latter along with its long-range variant maintains the temporal consistency among the sequence of video frames. The incorporation of these temporal information-based objective functions, thus stabilizes the training process to provide much improved segmentation over other state-of-the-art methods published earlier in the literature. However, the proposed model exhibits suboptimal performance in complex situations involving multiple objects moving simultaneously with varying velocities.

Finally, we design a dual-channel Graph Convolutional Network, which uses the raw RGB image and optical flow vector based features to model the inter-pixel relationships

in the video frames across space-time. Moreover, a direction oriented motion based aggregator mechanism is introduced which not only captures the variety in motion patterns involved with multiple objects moving in different directions with dissimilar speeds, but also efficiently models the change of appearance of the same in terms of pose and scale with time. A novel occlusion aware aggregation scheme is also incorporated to assist the network in segmenting objects undergoing occlusion or re-identifying targets disappearing from the frames and re-surfacing after some time. Performance analysis on DAVIS-2016 and DAVIS-2017 datasets show the effectiveness of our proposed method in foreground segmentation of objects in videos over the existing state-of-the-art techniques. Controlled experiments done on CamVid road segmentation dataset also shows the generalising capability of the model for semantic segmentation in the videos.

1.4 Organization of the Thesis

The chapters of the thesis are organized illustrating the progress of Video Object Segmentation (VOS) from using adversarial networks with specific tailored objective functions to the use of Graph Convolutional Networks (GCN) coupled with novel aggregation mechanisms.

Chapter 2 briefly discusses the various approaches proposed in the recent past related to the segmentation of the objects in videos. The chapter describes several methods based on object matching, mask propagation and object proposals, harnessing the power of Convolutional Neural Networks (CNNs) for the task of Video Object Segmentation. Existing works on Video Semantic Segmentation (VSS) under complex scenarios have also been discussed in this chapter, as an extension of VOS problem.

Chapter 3 introduces Generative Adversarial Network (GAN) based framework using a novel Patch-wise Symmetric Difference Loss (PSDL) along with the standard adversarial losses to segment the objects much smaller in scale precisely. Though the network does not involve any explicit trajectory based information, still it is able to maintain the temporal coherency among the sequences of video frames by processing them independently. The results on real-world benchmark VOS datasets viz. DAVIS-2016, SegTrack-v2 and YouTube-Objects have been used to prove the performance of our approach in comparison with the existing state-of-the-art methods.

Chapter 4 describes a temporally aided extension of the model discussed in the previous chapter. Two novel cost functions: Inter-frame Temporal Symmetric Difference Loss (ITSDL) and Intra Frame Temporal Loss (IFTL) have been implemented which uses optical flow not only to capture motion features among video frames in enhancing segmentation quality but also maintains the stability among them to improve the training process. These leads to much more improved results on the VOS benchmark datasets in comparison to our earlier method.

Chapter 5 illustrates a dual-channel Graph Convolutional Network (GCN) based on spatio-temporal inter-pixel relationships used for the task of VOS. Additionally, motion and occlusion aware mechanisms are incorporated to learn the motion pattern involved with multiple objects, appearance change of targets with time and also complex scenarios of re-identifying objects under occlusions. Experiments on VOS benchmark datasets (DAVIS-2016 and DAVIS-2017) and CamVid road scene segmentation dataset show significant gain in performance of the proposed model over other state-of-the-art methods.

Chapter 6 concludes the thesis with a summary of all our research works done in the field of Video Object Segmentation along with the challenges resolved to obtain a perceptually good quality result. It also provides direction to a few possible extensions of our work to be explored in future.

CHAPTER 2

Review of Related Works

2.1 Technical Background

In this chapter, before diving deep into the advancements made in the field of Video Object Segmentation in the recent past, a brief overview of the two fundamental concepts in generative modelling using deep neural networks *viz.* Generative Adversarial Networks (GAN) and Graph Convolutional Networks (GCN) have ben discussed. These frameworks have been used to design the proposed techniques mentioned in the thesis.

2.1.1 Generative Adversarial Networks

In the past, the deep generative models had less influence in the field of video object segmentation due to the complexities involved in the approximation of the likelihood estimations and other probabilistic computations while training the network. However, the difficulty of approximating the underlying distribution from a large amount of training data was eradicated to a large extent by Goodfellow *et al.* (2014) with the introduction of the concept of adversarial training of two models: generator and discriminator.

In this min-max game theory based setting, the generator model produces data distribution from noise by mimicking the original ground-truth data against the adversary of the discriminator. The discriminative model is modelled to differentiate whether a sample is obtained from the data distribution generated by the generator or from the original distribution. The data distribution of the generator ($p_{\mathbb{G}}$) over the input data x is modelled by defining a prior over the input noise variable $p_z(z)$, which represents a mapping $\mathbb{G}(z;\theta_{\mathbb{G}})$. $\mathbb{G}$ being a differentiable function (here, a deep neural network) whereas $\theta_{\mathbb{G}}$ are its parameters. Another function, $\mathbb{D}(x,\theta_{\mathbb{D}})$ is defined with parameters $\theta_{\mathbb{D}}$ to represent the probability of x being sampled from the original distribution, instead of $p_{\mathbb{G}}$, using a scalar value as output. Thus the generator network is trained to minimize the difference between generated data and the original one, while the discriminator is

trained to maximize the same by assigning correct labels to the sampled data. The overall equation of the two networks involved in a two-player zero-sum min-max game can be expressed as:

$$\min_{\mathbb{G}} \max_{\mathbb{D}} u(\mathbb{G}, \mathbb{D}) = \mathbb{E}_{x \sim p_{data}}[\log(\mathbb{D}(x))] + \mathbb{E}_{z \sim p_z}[\log(1 - \mathbb{D}(\mathbb{G}(z)))] \tag{2.1}$$

Like the conventional expectation-maximization algorithm, the method is also implemented in an iterative manner.

In the field of Video Object Segmentation, since the problem statement involves video clips as the input data, the generator network has been modelled to produce photo-realistic segmentation masks of the input sequence of video frames. At the same time, the discriminator is trained to differentiate between the ground-truth mask and the generated one. With the model approaching equilibrium, the generator produces masks close to the target mask, making it gradually tough for the discriminator to distinguish between the original and synthetic data.

2.1.2 Graph Convolutional Neural Networks

Researches have been performed to learn arbitrary graphical structures implementing deep neural architectures like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), etc. Recently, a spectral graph convolutional technique has been introduced by Kipf and Welling (2016) to learn the features associated with a graphical structure. The inputs of the network are: (i) a feature representation v_i corresponding to every node i, expressed as matrix X of dimension $\mathbb{N} \times D$, where $\mathbb{N}$ and D being the number of graph nodes and input features respectively, and (ii) illustration of the graph in the form of a matrix (like an adjacency matrix). A feature matrix of dimension $\mathbb{N} \times F$ is obtained as a nod-level output Z from the network with F being the number of output features for each node.

The non-linear function used to describe a graph convolutional layer of a neural network, using the setting as mentioned earlier, is expressed as:

$$\mathbb{H}^{(l+1)} = f(\mathbb{H}^{(l)}, \mathbb{S}) \tag{2.2}$$

where, $\mathbb{H}^{(0)} = X$ and $\mathbb{H}^{(L)} = Z$; L is the number of layers in the network; f is a propagation function. A most commonly used form of this propagation function is $f(\mathbb{H}^{(l)}, \mathbb{S}) = D^{-1}\mathbb{S}\mathbb{H}^{(l)}$, where, $\mathbb{S}$ is the adjacency matrix of the graph and D^{-1}, the inverse degree matrix.

2.2 Recent Advancements

Video Object Segmentation (VOS) has been a challenging and important problem in the field of Computer Vision and Machine Learning in recent past due to its profound applications in domains like medical imaging, robotics, video editing, video prediction, action recognition etc. In this thesis, we deal with the problem of segmenting multiple objects of interests over a sequence of RGB video frames in a semi-supervised setting by providing the ground-truth annotation for the initial frame. Although there has been enormous progress in semantic segmentation using Convolutional Neural Networks (CNNs) (Long *et al.*, 2015; Chen *et al.*, 2017; Peng *et al.*, 2017), VOS has always remained challenging for objects moving in and out of frame, lack of any concrete association among objects due to shape and appearance deformation, camera shake, occlusions, motion blur and background clutter.

A number of methods have been proposed in the recent past based on object matching, mask propagation and object detection to deal with these challenges.

Matching-based methods. In matching-based approaches, the appearance information of the objects of interest is extracted from the given first frame ground-truth segmentation mask, to propagate the segmentation in the subsequent frames. A deep siamese network was proposed by Shin Yoon *et al.* (2017) for inter-frame object matching in feature space. In Caelles *et al.* (2017); Maninis *et al.* (2018) CNNs are used as the parent network for training on still images followed by fine-tuning of the same with annotated object mask of the initial video frame leading to one-shot learning. Khoreva *et al.* (2017); Voigtlaender and Leibe (2017) enriched the appearance information by synthesizing more training data depending on the first frame. Chen *et al.* (2018*b*); Hu *et al.* (2018*c*) used first frame supervision to learn the pixel-wise embeddings, which aid in pixel classification in subsequent video frames. Tracking of various object parts along with region-of-interest segmentation and similarity-based aggregation was proposed in Cheng

et al. (2018) to tackle the challenges like occlusions and deformations.

Mask propagation based methods. Apart from these matching-based methods, temporal information propagated from the past frames were used to refine the segmentation masks in mask propagation based approaches. Perazzi *et al.* (2017) is an example of this genre trained with augmented segmentation masks of static images. This work was extended in Hu *et al.* (2018*b*) to find motion cues by implementing active contour on optical flow. Temporal information from nearby key frames were combined in Wang *et al.* (2018); Xiao *et al.* (2018) to deal with the challenges of missing objects in situations like occlusions and rapid motion. The temporal coherency is further improved in Bao *et al.* (2018) by embedding the mask propagation in the space-time based MRF model inference. Wug Oh *et al.* (2018) utilized a siamese framework to incorporate instance detection within mask propagation to segment the target without relying on online fine-tuning for a given video. Jampani *et al.* (2017) combined CNN training with the idea of bilateral filtering in their video propagation network, which propagates information across video sequences. In addition, Conditional Batch Normalization (CBN) proposed by Yang *et al.* (2018) does not require online learning to extract spatio-temporal features. In ObjFlow (Tsai *et al.*, 2016), segmentation of videos was done by building a graph over super-pixels, using appearance terms based on convnet and optical flow estimation. Instead of using super-pixels, Märki *et al.* (2016) tried to solve it in an efficient way by projecting the problem into a bilateral space. Recently, Oh *et al.* (2019*b*) has followed the mask propagation approach by leveraging memory networks, where the past frames with object masks form external memory and provide spatio-temporal guidance information to segment the objects in the current frame. In FEELVOS (Voigtlaender *et al.*, 2019) information is transferred from the previous frames of the videos to the current frame using a semantic pixel-wise embedding coupled with local and global matching mechanisms.

Object detection based methods. Another type of method used for VOS is based on object proposals, which are cropped by the detection algorithms before passing it into the segmentation module. VS-ReID algorithm proposed by Li *et al.* (2017) segments missing objects in videos. Whereas, Faster R-CNN (Ren *et al.*, 2017) was used in Sharir *et al.* (2017) to generate object proposals for precise bounding box information. In Luiten *et al.* (2018) the supervised targets among video frames detected by Mask R-CNN (He *et al.*, 2017) were cropped to provide as an input to the DeeplabV3+ (Chen *et al.*,

2018*a*) module. Majority of the detection-based methods choose a single proposal at a time in a greedy manner.

Hybrid approaches. Recently, a new genre has been introduced, known as the *hybrid methods*, which involves a fusion of two or more types of approaches mentioned above, to solve the problem of segmentation of objects in videos. RANet (Wang *et al.*, 2019) integrates the insights of both matching and propagation based techniques by using a ranking based attention module to select the perceptually more effective segmentation map for the purpose of matching and then transmitting the visual information through the generated frames for fine-grained VOS performance. Xu *et al.* (2019) follow the object detection based Multiple Hypothesis Tracking mechanism to propagate the bounding box proposals from the previous frames to the predicted frames.

Unsupervised methods. Besides semi-supervised techniques, lots of researches have been done in this field based on unsupervised approaches. In this setting, there are no human interactions or ground-truth annotation masks involved, and visually salient objects are segmented based on motion and appearance. The drawback of unsupervised methods is that they fail to segment multiple objects as they show suboptimal performance in identifying a particular instance. A combination of appearance-based model and optical flow network has been used in Jain *et al.* (2017) to segment generic foreground objects in videos. Similarly, Tokmakov *et al.* (2017) segment moving objects in video clips by implementing a Recurrent Neural Network for motion estimation. The drawback of the unsupervised setting lies in the fact that the users do not have the freedom to choose the object(s) of interest.

Interactive methods. Another popular way of solving the Video Object Segmentation problem is by leveraging the interactive mode. In this approach, the user actively takes part in the process by providing various types of input such as hand-drawn scribbles, bounding boxes or masks to select the object of interest initially and also to refine the segmentation output in the course of the process to achieve a satisfactory solution with minimum interactions. Wang *et al.* (2005); Price *et al.* (2009); Shankar Nagaraja *et al.* (2015) utilized hand-crafted energy terms to solve spatio-temporal graphs. Whereas, local classifiers are used by Bai *et al.* (2009); Zhong *et al.* (2012) to match the corresponding patches between reference and target frames. Tracking has been incorporated in Agarwala *et al.* (2004); Li *et al.* (2016) to segment objects in an interactive scenario.

Deep interactive image segmentation method (Xu *et al.*, 2016) is used in Benard and Gygli (2017); Oh *et al.* (2019*a*) to select objects of interest using given initial clicks or strokes by the user, followed by propagation of the object masks through the video using semi-supervised video object segmentation method (Caelles *et al.*, 2017). Oh *et al.* (2019*a*) introduced CNN based framework for simultaneous training of connected interaction and propagation modules to solve the problem.

Short-comings. Although the existing approaches produce good quality segmentation for long video clips, they fail in cases where multiple objects are involved in motion with different speeds in various directions, producing sub-optimal segmentations. The main reason behind this is the inability of the networks to model the inter-pixel relationships explicitly. The reasoning of inter-pixel or inter-object relationships has been previously studied in the domains like object recognition (Gkioxari *et al.*, 2018) and detection (Hu *et al.*, 2018*a*), modelling interactions (Watters *et al.*, 2017) and visual question answering (Santoro *et al.*, 2017). Thus, efficient learning of spatio-temporal relationship over multiple time-steps is required to implement the interaction framework in the task of VOS. Another significant drawback of the explicit trajectory flow intensive methods is that these methods produce unsatisfactory results in cases of objects undergoing occlusion, scene change and re-identification of objects moving out of the frame and re-appearing after some time, due to the sudden disruption of the temporal information. Independent processing of frames or estimation of both forward and reverse motion features of objects can be accounted for overcoming this challenge.

2.3 Motivation

Recently, Generative Adversarial Networks (GANs) have taken over the field of image segmentation (Isola *et al.*, 2017; Luc *et al.*, 2016; Souly *et al.*, 2017) because of their success in generative modelling. Isola *et al.* (2017) mention the capability of GAN to learn the mapping function to translate images of one domain to another, whereas Luc *et al.* (2016); Souly *et al.* (2017) used conventional GAN architecture for semi- and weakly supervised semantic segmentation.

The main motivation behind using GAN based framework in our work is to harness its power to learn the mapping from the input (RGB space) to the output space (segmentation

mask space). In this thesis, GAN based models have been proposed to subdue the limitations of the previous methods by segmenting object(s) of interest using spatio-temporal information present in the videos under a semi-supervised setting. Two different ways of utilizing this information in training the network have been adopted: (i) by incorporating novel objective functions to train the model to learn the input-output mapping, and (ii) deploying graph-like structures over GAN to model the underlying motion by learning the relationships between pixels.

Extensive comparative and ablation studies on various real-world benchmark datasets: DAVIS-2016 (Perazzi *et al.*, 2016), DAVIS-2017 (Pont-Tuset *et al.*, 2017*a*), SegTrack-v2 (Li *et al.*, 2013), YouTube Objects (Prest *et al.*, 2012) and CamVid (Brostow *et al.*, 2009) datasets exhibit substantial gain in performance of our approaches, both qualitatively and quantitatively. The proposed models display optimal solution even in challenging scenarios like objects undergoing occlusion, objects re-appearing in the frame after prolonged disappearance and multiple objects moving in different directions with various motion patterns and speeds.

CHAPTER 3

VidSeg-GAN: Generative Adversarial Network for Video Object Segmentation (VOS) Tasks

This chapter describes our first approach towards solving the problem of video object segmentation. Here, a Generative Adversarial Network (GAN) has been introduced along with an Intersection-over-Union (IoU) guided objective function to produce segmentation masks of object(s) involved in the videos. Although Convolutional Neural Networks (CNNs) (Caelles *et al.*, 2017; Perazzi *et al.*, 2017; Voigtlaender and Leibe, 2017) have been used in the recent past for the purpose of foreground segmentation in videos, adversarial training methods have not been used effectively to solve this problem, in spite of its extensive use for solving many other problems in Computer Vision. Recently, Generative Adversarial Networks (GANs) are taking over the field of image segmentation (Isola *et al.*, 2017; Luc *et al.*, 2016; Souly *et al.*, 2017) because of their success in generative modelling. Isola *et al.* (Isola *et al.*, 2017) mention the capability of GAN to learn the mapping function to translate images of one domain to another, whereas (Luc *et al.*, 2016; Souly *et al.*, 2017) used conventional GAN architecture for semi- and weakly supervised semantic segmentation. The main motivation behind using GAN based framework is to harness its power to learn the mapping from the input (RGB space) to the output space (segmentation mask space). In our work, we build a deep generator-discriminator framework and carry out adversarial training over it. We also introduce a Patch-wise Symmetric Difference Loss (PSDL) along with the standard GAN losses to help the model to improve minute details while performing foreground object segmentation over frames of video sequences, when provided with the initial annotated frame.

In some earlier works (Perazzi *et al.*, 2017; Jampani *et al.*, 2017; Tsai *et al.*, 2016), flow features and motion trajectories have been extensively used to capture the temporal consistency between subsequent frames to segment moving objects in videos. The major shortcoming of these methods is that they fail to segment objects disappearing from the frames and resurfacing after some time or segmenting the object of interest

Figure 3.1: Segmentation output of MSK (Perazzi *et al.*, 2017) on YouTube-Objects dataset. The red mask of the first frame denotes the ground-truth, whereas the green masks demonstrate the output of the MSK model. The network fails to generate the segmentation mask of the object of interest (car) in the latter part of the video sequence (as shown in the last 2 frames). The frames at an equal interval of video sequence length have been shown (best viewed in color).

after a scene change, due to the disruption of the flow trajectories (refer figure 3.1). This issue is overcome in our proposed method through the processing of the video frames independently using a deep generative adversarial framework along with the novel objective function (PSDL), which is able to maintain the temporal coherency across frames without the use of any explicit trajectory-based information, to provide superior results. The PSDL not only aids the network in training the mapping from RGB image space to the segmentation space but also helps in improving the minute details involved. The proposed end-to-end trainable model exhibits substantial performance gain over the state-of-the-art methods when evaluated on popular real-world video object segmentation datasets *viz.* DAVIS-2016 (Perazzi *et al.*, 2016), SegTrack-v2 (Li *et al.*, 2013) and YouTube-Objects (Prest *et al.*, 2012) (refer figure 3.4).

3.1 Video Segmentation Generative Adversarial Network (VidSeg-GAN)

In this section, we propose a generative adversarial network (GAN) based framework (VidSeg-GAN) to generate segmentation masks of objects of interest in videos. The proposed architecture transforms the direct but non-trivial problem of segmenting the videos into a generative task which can be better handled under an adversarial setting. The use of an encoder-decoder model with skip connections as generator sub-network captures contextual information which helps the framework to remember and segment the object of interest throughout the video sequence. In addition to the adversarial losses to train the GAN, a novel Patch-wise Symmetric Difference Loss (PSDL) based on

Intersection-over-Union (IoU) index, has been incorporated which improves quantitative result as well as stabilizes the training. The proposed PSDL considers small patches of estimated segmented mask and compares it with the corresponding ground truth mask patches. PSDL decreases the number of pixels erroneously identified in the predicted masks, thus helping in improvement of minute details in the segmentation results. Effectiveness of the proposed loss function is evident from the performance of our GAN based model over three popular real-world video datasets, in estimating the segmentation masks of video sequences.

Generative Adversarial Networks (GAN) (Goodfellow *et al.*, 2014) are composed of two models: (i) the Generator ($\mathbb{G}$) and (ii) the Discriminator ($\mathbb{D}$). The aim of the generator $\mathbb{G}$ is to extract features from the true data distribution p_{data} to generate photo-realistic images which are close to the original one, thus making it difficult for the discriminator to distinguish between real and generated images. On the other hand, optimization of the discriminator $\mathbb{D}$ is done to predict whether the generated output is real or synthetic. Thus, the alternate learning process of the Generator and Discriminator models in this network resembles the two-player min-max games (Goodfellow *et al.*, 2014). Simultaneous minimization of the loss at $\mathbb{G}$ and maximization of the same at $\mathbb{D}$ is done through the overall objective function which is defined as follows:

$$\min_{\mathbb{G}} \max_{\mathbb{D}} u(\mathbb{G}, \mathbb{D}) = \mathbb{E}_{x \sim p_{data}}[\log(\mathbb{D}(x))] + \mathbb{E}_{z \sim p_z}[\log(1 - \mathbb{D}(\mathbb{G}(z)))] \tag{3.1}$$

where, x is an original image from the true distribution p_{data} and z is a vector sampled from the distribution p_z, usually to be uniform or Gaussian. A variation of the equation 3.1 is implemented as the adversarial loss in this chapter, since the input to the framework is video frame sequence, instead of a vector z.

The proposed architecture is illustrated in figure 3.2. An encoder-decoder setting has been chosen for the generator where the convolutional layers are accompanied by pooling and unpooling layers (Zeiler and Fergus, 2014). ReLU is used as the non-linear activation function for layers involved in the generator. Batch-normalization (Ioffe and Szegedy, 2015) layers and dropout have also been implemented in this network. The context-based information is captured by the contracting path of the encoder, whereas the symmetric expanding path of the decoder is used to localize the information precisely. Inspired by "U-net" (Ronneberger *et al.*, 2015), skip connections have been employed

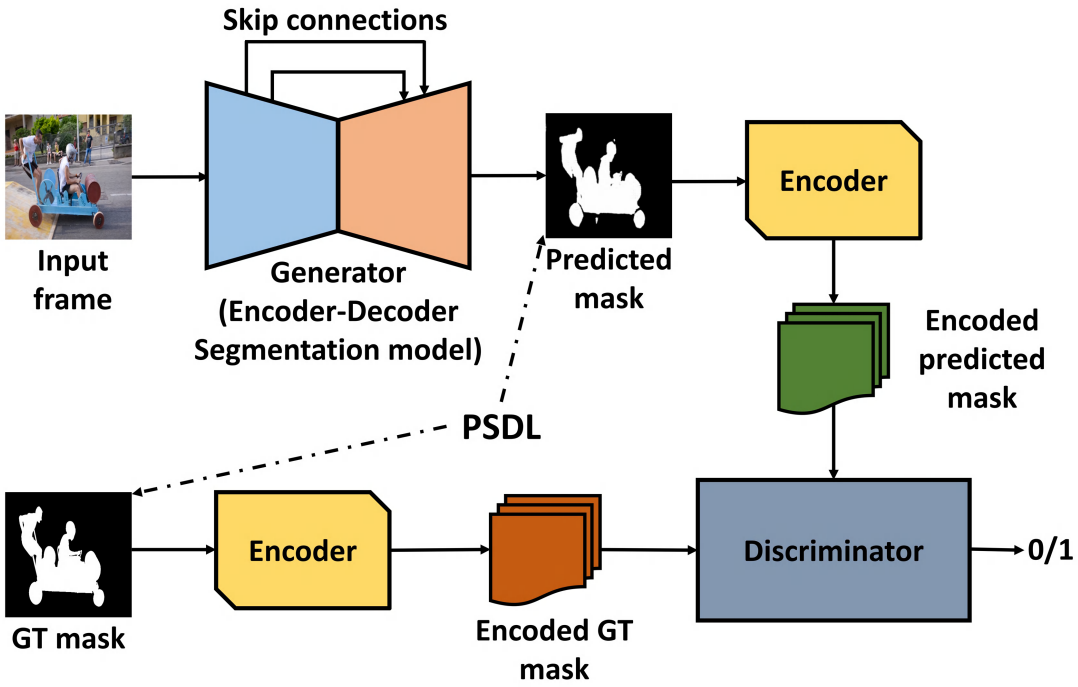


Figure 3.2: Proposed VidSeg-GAN architecture (GT refers to Ground-Truth). Dotted lines refer to the Patch-wise Symmetric Difference Loss ($\mathcal{L}_{PSDL}$) estimated between the network generated output mask and the ground-truth mask.

between each layer i and $(n-i)$, where n is the total number of layers. Low-level information like prominent edge and corner details are shared between the initial and final layers of the generator by concatenating all feature maps of layer i with those at the $(n-i)$ through skip connections. RGB video frames of dimension $(W \times H \times 3)$ passed sequentially form the input to the generator, while the output is a segmentation map of dimension $(W \times H \times 1)$, corresponding to each input frame. Two separate encoders of the same configuration are implemented to encode the predicted and ground-truth mask (see figure 3.2). Pooling layers combine with the convolutional layers to down-sample the input in each of these encoders. The encoded predicted and ground-truth mask obtained as output from the encoders, act as the discriminator input. The discriminator, formed of convolutional modules with fully-connected layers at the end, predicts 0 or 1 as output, denoting synthetic or real data.

The encoder modules encode the predicted output of the generator and the ground-truth before providing it into the discriminator, instead of using the generator output directly, which makes the proposed GAN architecture different from the traditional one (Goodfellow *et al.*, 2014). The motivation behind the encoding mask is that it facilitates the discriminator to differentiate between the real and fake outputs more efficiently by

increasing the separability in the projected high-dimensional feature space than in the RGB image space.

3.1.1 VidSeg-GAN training

The proposed GAN framework is trained in the same manner as that of the conventional adversarial networks with a minor variation. The generator $\mathbb{G}$ of VidSeg-GAN acts a segmentation model that generates output mask considering the joint data distribution of the input video frame ($\mathcal{I}$) and its corresponding ground-truth mask (Y). Whereas, the discriminator $\mathbb{D}$ differentiates between generated and the original mask, thus facilitating the training process by mitigating the disparity between the prediction and the ground-truth. The objective function of the adversarial training is defined as:

$$\min_{\theta_{\mathbb{G}}} \max_{\theta_{\mathbb{D}}} \sum_{\mathcal{I}} \mathcal{L}_{bce}(Y, \mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}})) - \lambda \big[\mathcal{L}_{bce}(1, \mathcal{O}_{\mathbb{D}}(\mathcal{O}_{\mathbb{E}}(Y); \theta_{\mathbb{D}})) + \mathcal{L}_{bce}(0, \mathcal{O}_{\mathbb{D}}(\mathcal{O}_{\mathbb{E}}(\mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}})); \theta_{\mathbb{D}})) \big] \tag{3.2}$$

where, $\theta_{\mathbb{G}}$ and $\theta_{\mathbb{D}}$ are parameters of generator and discriminator respectively. $\mathcal{L}_{bce}$ denotes the binary cross-entropy loss, which is as follows:

$$\mathcal{L}_{bce}(K, K') = -\sum_{i=1}^{|K|} K'_i \log(K_i) + (1 - K'_i) \log(1 - K_i), \quad K_i \in \{0, 1\}, K'_i \in [0, 1] \tag{3.3}$$

where, K' and K are the discriminator output and target respectively.
$\mathcal{O}_{\mathbb{G}}$ and $\mathcal{O}_{\mathbb{D}}$ are the outputs of the generator and discriminator respectively. $\mathcal{O}_{\mathbb{E}}(X)$ represents the encoded output form of the segmented mask, X. The discriminator labels 1 and 0 refers to whether the input is from the ground-truth Y or the generator $\mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}})$ respectively. λ is a regularization hyper-parameter.

Generator training

Keeping the discriminator weight parameters unchanged, the generator is optimized by passing a video sequence frame ($\mathcal{I}$) and its corresponding ground-truth mask Y into $\mathbb{G}$, and the network is trained to minimize the adversarial loss function in equation 3.2 w.r.t.

$\theta_{\mathbb{G}}$, as:

$$\mathcal{L}_{adv}^{\mathbb{G}}(\mathcal{I}) = \min_{\theta_{\mathbb{G}}} \sum_{\mathcal{I}} \mathcal{L}_{bce}(Y, \mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}})) + \lambda \mathcal{L}_{bce}(1, \mathcal{O}_{\mathbb{D}}(\mathcal{O}_{\mathbb{E}}(\mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}})); \theta_{\mathbb{D}})) \quad (3.4)$$

In equation 3.4, the consistency of the predicted segmentation with the target mask at each position is maintained by the first term, whereas the second term penalizes the unfitting structure between the generated output and ground-truth.

Discriminator training

Encoded predicted and target maps $\mathcal{O}_{\mathbb{E}}(\mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}}))$ and $\mathcal{O}_{\mathbb{E}}(Y)$ are fed into the discriminator ($\mathbb{D}$) instead of sending them directly, since it aids in differentiating the synthetic data from the true distribution in the encoded space. Here, training of $\mathbb{D}$ is done in such a way that it classifies $(\mathcal{O}_{\mathbb{E}}(\mathcal{I}), \mathcal{O}_{\mathbb{E}}(Y))$ into class 1 and assigns class 0 to $(\mathcal{O}_{\mathbb{E}}(\mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}})), \mathcal{O}_{\mathbb{E}}(Y))$ respectively. D is trained by the objective function obtained by minimizing equation 3.2 w.r.t. $\theta_{\mathbb{D}}$, as follows:

$$\mathcal{L}_{adv}^{\mathbb{D}} = \min_{\theta_{\mathbb{D}}} \sum_{\mathcal{I}} \left[\mathcal{L}_{bce}(1, \mathcal{O}_{\mathbb{D}}(\mathcal{O}_{\mathbb{E}}(Y); \theta_{\mathbb{D}})) + \mathcal{L}_{bce}(0, \mathcal{O}_{\mathbb{D}}(\mathcal{O}_{\mathbb{E}}(\mathcal{O}_{\mathbb{G}}(\mathcal{I}; \theta_{\mathbb{G}})); \theta_{\mathbb{D}})) \right] \quad (3.5)$$

Though the theoretical foundation of this alternate optimization process of generator and discriminator is logically firm and well-established, in reality it is susceptible to mode collapse (Radford *et al.*, 2015; Salimans *et al.*, 2016) leading to instabilities in training. A patch-wise Intersection-over-Union (IoU) based objective function is formulated and used along with the existing conventional adversarial losses described in equations 3.4 - 3.5 to overcome the implicit instability and to produce better predicted segmented masks.

Pre-trained weights on ImageNet (Deng *et al.*, 2009) has been used to train the generator network in our proposed VidSeg-GAN architecture. For training on DAVIS-2016 (Perazzi *et al.*, 2016) dataset, 30 sequences of annotated frames are used. While validation, the model is fine-tuned with the initial frame of each of the video sequence, belonging to the validation set, along with its ground-truth mask to capture the appearance of the specific object of interest, followed by feeding the remaining video frames sequentially into the VidSeg-GAN to generate the corresponding predicted segmented masks.

3.2 Patch-wise Symmetric Difference Loss (PSDL)

Intersection-over-Union (IoU) is used as an evaluation metric for Segmentation tasks, in the form of Jaccard Index ($\mathcal{J}$) for measuring the region of similarity between the predicted mask ($\hat{Y}$) and the target mask (Y), i.e. the number of overlapping pixels. In other words, it gives an idea of the number of mispredicted pixels present in the estimated mask compared to the ground-truth. Thus, the Jaccard index based on IoU score of class c is defined as:

$$\mathcal{J}_c(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{|\{\boldsymbol{y}_v = c\} \cap \{\hat{\boldsymbol{y}}_v = c\}|}{|\{\boldsymbol{y}_v = c\} \cup \{\hat{\boldsymbol{y}}_v = c\}|}, \quad \forall v = 1, \cdots, S^2 \tag{3.6}$$

where, each pixel i of an image $\mathcal{I}$ is classified into class $c \in \mathbb{C}$. In our work , $\mathbb{C} = \{0, 1\}$, $\hat{\boldsymbol{y}}$ and $\boldsymbol{y}$ denote the vector of predicted labels and ground-truth labels in the estimated mask ($\hat{Y}$) and target mask (Y) respectively; $\boldsymbol{y}, \hat{\boldsymbol{y}} \in \mathbb{C}^{S^2}$; S is the dimension of the masks. $\mathcal{J}_c$ gives a ratio in $[0, 1]$ of intersection between estimated and ground-truth mask over their union, with the standard that $0/0 = 1$. Following this, the corresponding Intersection-over-Union loss (Rahman and Wang, 2016; Matthew and Blaschko, 2018) obtained for minimization can be written as:

$$\Delta_{\mathcal{J}_c}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = 1 - \mathcal{J}_c(\boldsymbol{y}, \hat{\boldsymbol{y}}) \tag{3.7}$$

We relied upon using the loss function in a patch-wise manner than using it globally over the entire mask. This is due to the fact that occlusion of the object of interest by other objects often gives rise to small disjoint masks, which do not contribute to the objective function when estimated globally. Whereas, evaluating patch-wise in a smaller scale enhances the attention over the small disjoint sections, which in turn contributes to the better training of the model. Also, this patch-wise objective function is quite effective as it adds a greater weight on the mask pixels, which makes it effective for the object of interest with very small scale, which often gets missed when evaluated globally.

For formulating the patch-wise loss function, we calculate the number of mispredicted pixels ($|M_p|$) in a patch of the segmented mask. The set of mispredicted pixels (M_p) for class $c \in \{0, 1\}$, obtained by using Symmetric difference of the two vectors $\boldsymbol{p}$ and $\hat{\boldsymbol{p}}$, is:

$$\boldsymbol{M}_p(\boldsymbol{p}, \hat{\boldsymbol{p}}) = \{\boldsymbol{p}_u = c, \hat{\boldsymbol{p}}_u \neq c\} \cup \{\boldsymbol{p}_u \neq c, \hat{\boldsymbol{p}}_u = c\}, \quad \forall u = 1, \cdots, s^2 \tag{3.8}$$

Algorithm 3.1: Patch-wise Symmetric difference score for estimating similarity between predicted mask(s) and ground-truth mask(s).

Input: Ground-truth masks (Y), Predicted masks ($\hat{Y}$)
Output: Symmetric difference score ($Score_{PSD}$)
// $s =$ height and width of an patch on the mask
// $S =$ height and width of the predicted masks
// $N =$ Number of masks predicted
Initialize: $Score_{PSD} = 0$;
for $t = 1$ *to* N **do**
  for $i = 0$ *to* S, $i \leftarrow i + 1$ **do**
    for $j = 0$ *to* S, $j \leftarrow j + 1$ **do**
      $\hat{P} \leftarrow extract\_patch(\hat{Y}, i, j, s)$;
      /* Extracts a patch from the predicted mask $\hat{Y}$, of dimension $s \times s$ starting from the top-left pixel index (i, j). */
      $P \leftarrow extract\_patch(Y, i, j, s)$;
      /* Extracts corresponding patch from the ground-truth mask Y, of dimension $s \times s$. */
      $\hat{\boldsymbol{p}} \leftarrow$ vector of predicted labels of pixels obtained from the patch $\hat{P}$;
      $\boldsymbol{p} \leftarrow$ vector of ground-truth labels of pixels obtained from the patch P;
      $\boldsymbol{M}_p \leftarrow \{\boldsymbol{p}_u = 1, \hat{\boldsymbol{p}}_u = 0\} \cup \{\boldsymbol{p}_u = 0, \hat{\boldsymbol{p}}_u = 1\}, \forall u = 1, \cdots, s^2$(refer equation 3.8);
      /* $\boldsymbol{M}_p$ denotes the set of mislabelled pixels in the patch. */
      $Score_{PSD} \leftarrow Score_{PSD} + \frac{|\boldsymbol{M}_p|}{s^2}$ (refer equation 3.9);
    end
  end
  $Score_{PSD} \leftarrow {}^{Score_{PSD}}/_{(S-s)^2}$; // Average over all the patches
end
$Score_{PSD} \leftarrow {}^{Score_{PSD}}/_{N}$; // Average over all the masks

where, $\hat{\boldsymbol{p}}$ denotes the vector of predicted labels in the $s \times s$ dimension patch of the estimated mask ($\hat{Y}$) and $\boldsymbol{p}$ is the vector of ground-truth labels in the corresponding patch of the target mask (Y); $\boldsymbol{p}, \hat{\boldsymbol{p}} \in \{0, 1\}^{s^2}$.

The Patch-wise Symmetric Difference score (PSD_{patch}) for each patch of a predicted mask can be written as:

$$PSD_{patch} = \frac{|\boldsymbol{M}_p|}{s^2} \tag{3.9}$$

which gives a ratio of mispredicted pixels to the total number of pixels for each patch.

The process of finding the PSDL by matching the corresponding local patches of the

estimated and ground-truth masks is described in algorithm 3.1. An objective function is modelled by calculating the PSD score for the generator ($\mathbb{G}$) network, where it minimizes the score over an input batch. The loss function, $\mathcal{L}_{PSDL}$ is defined as:

$$\mathcal{L}_{PSDL}(Y, \hat{Y}) = Score_{PSD}(Y, \hat{Y}) \tag{3.10}$$

where, $\hat{Y}$ and Y represent the predicted and ground-truth mask, and $Score_{PSD}$ denotes the average symmetric difference score over all the masks, obtained using algorithm 3.1. The generator minimizes $\mathcal{L}_{PSDL}$ along with the adversarial losses mentioned in section 3.1.

Overall loss. Finally, the PSDL is combined with the generative adversarial loss (refer equation 3.4) and the traditional $\mathcal{L}_{L_1}$ loss with different weights, as follows:

$$\mathcal{L}_{combined} = \alpha_{adv}\mathcal{L}^{\mathbb{G}}_{adv}(\mathcal{I}) + \alpha_{L_1}\mathcal{L}_{L_1}(Y, \hat{Y}) + \alpha_{PSDL}\mathcal{L}_{PSDL}(Y, \hat{Y}) \tag{3.11}$$

where, the weights *viz.* α_{L_1} and α_{PSDL} are set to 0.5 while α_{adv} is made equal to 0.1. The overall loss is minimized during the Generative Adversarial Network training using Adam optimizer (Kingma and Ba, 2014).

3.3 Experimental Results and Discussions

In this section, the performance analysis of our proposed model for video object segmentation has been discussed on three benchmark real-world datasets. The training set is used to train the model, while validation has been carried out on independent frames by feeding a single input image, in a sequential manner, into the deep network which generates segmentation mask as output. Before validation, the model is fine-tuned with the first frame of each video sequence to specify the object of interest. Three metrics: Region similarity ($\mathcal{J}$), Contour accuracy ($\mathcal{F}$) and Temporal (in-)stability ($\mathcal{T}$) (Perazzi *et al.*, 2016) have been used to compare the results of our network with the existing state-of-the-art techniques.

3.3.1 Datasets

The proposed GAN based method is evaluated on three benchmark real-world video object segmentation datasets *viz.* DAVIS-2016 (Perazzi *et al.*, 2016), SegTrack-v2 (Li *et al.*, 2013) and YouTube-Objects (Prest *et al.*, 2012) dataset. The challenging characteristics that these datasets include are change of appearance of objects, background clutter, motion blur, occlusion, shape deformation, etc.

The recently-released **DAVIS-2016 dataset** (Perazzi *et al.*, 2016) includes 50 full-resolution High-Definition video sequences totalling $3,455$ frames. Each of the frames is densely segmented with pixel-level accuracy, which separates single or multiple connected objects from the background. All the videos in this dataset are present as a temporal sequence of frames which are used for training and validation. Among these, 30 pre-defined (Perazzi *et al.*, 2016) video sequences are used for training and remaining for validation purpose in our experiments.

SegTrack-v2 (Li *et al.*, 2013) consists of 14 videos with 24 objects. Each of the 947 frames is provided with pixel-level segmentation mask. Instance-level segmentation is adopted to annotate sequences having multiple objects, and each annotation is treated as an individual object.

YouTube-Objects Dataset (Prest *et al.*, 2012) contains video sequences with 10 object classes. A subset consisting of 126 videos with about $20,000$ frames is considered, where the pixel-level ground-truth annotation masks are obtained from (Jain and Grauman, 2014).

3.3.2 Network Architecture Details

Elaborate description of the generator ($\mathbb{G}$), discriminator ($\mathbb{D}$) and encoder ($\mathbb{E}$) networks for experimental analysis are exhibited in table 3.1. The generator network consists of convolution layers with ReLU non-linearity, batch-normalization layers and dropout at a rate of 50%. The encoder-decoder network of $\mathbb{G}$ utilizes skip-connections to connect layer i with layer $(n - i)$ by concatenating the feature maps of former with that of latter. The images are also upsampled by a factor of 2 into higher resolution in terms of both height and width using Unpooling layers (Zeiler and Fergus, 2014). Pre-trained ImageNet (Deng *et al.*, 2009) weights have been used to initialize the generator, whereas

Table 3.1: Architectural details of VidSeg-GAN; $\mathbb{G}$, $\mathbb{D}$ and $\mathbb{E}$ denotes generator, discriminator and encoder networks respectively.

Network	**Generator ($\mathbb{G}$)**	**Discriminator ($\mathbb{D}$)**	**Encoder ($\mathbb{E}$)**
Number of feature maps	64, 128, 256, 512, 512, 512, 512, 256, 128, 64	256, 512, 512	64, 128, 256
Kernel sizes	5, 3, 3, 3, 3, 3, 3, 3, 3, 5	3, 5, 5	5, 3, 3
Fully connected	N/A	1024, 512	N/A

the learning rate is set to 0.002 for training, which decreases gradually over time upto 0.0004. For the discriminator ($\mathbb{D}$), the learning rate is kept fixed to 0.01 and the network also uses ReLU non-linearities. To train the overall network, mini-batches of 50 frames of video sequences are used.

3.3.3 Evaluation metric for Segmentation

Quantitative analysis of the performance of our proposed system is done using the estimated segmented masks, in comparison with the ground-truth masks, over three methods (Perazzi *et al.*, 2016) : (i) Region similarity ($\mathcal{J}$), (ii) Contour accuracy ($\mathcal{F}$) and (iii) Temporal (in-)stability ($\mathcal{T}$). These methods are discussed below.

Region similarity ($\mathcal{J}$) measures the number of mislabelled pixels, accounting to how good the match is between the estimated ($\hat{Y}$) and the target mask (Y). Jaccard Index ($\mathcal{J}$) is used to measure this region-based segmentation similarity which is defined as *intersection-over-union* of the predicted and ground-truth mask, i.e. $\mathcal{J} = \frac{|\hat{Y} \cap Y|}{|\hat{Y} \cup Y|}$.

Contour accuracy ($\mathcal{F}$) measures the precision of the segmentation boundaries. The spatial extent of the estimated mask $\hat{Y}$ can be interpreted as a set of closed contours $u(\hat{Y})$. P_u and R_u denote the contour-based precision and recall between $u(\hat{Y})$ and $u(Y)$ contour points. F-measure, defined as $\mathcal{F} = \frac{2P_u R_u}{P_u + R_u}$, gives the value of the contour accuracy.

Temporal (in-)stability ($\mathcal{T}$) measures the consistency of the video sequence frames over time. Unstable boundaries and jitters produce unwanted artifacts in video object segmentation. Hence, $\mathcal{T}$ penalizes the undesired instabilities and thus, giving a measure of how smooth and precise are the transformations in a video sequence so that it can be termed as stable. $\mathcal{T}$ is measured using the formulation discussed in (Perazzi *et al.*, 2016).

3.3.4 Performance Analysis of Video Object Segmentation

Pre-training of the proposed VidSeg-GAN model is first done on the ImageNet (Deng *et al.*, 2009) dataset, followed by training on DAVIS-2016 (Perazzi *et al.*, 2016) dataset, where the network uses annotated frames from 30 pre-allocated video sequences. To increase the number of training samples, the input frames are augmented by random rotation, flipping and zooming. While validation, the initial frame of each of the 20 video sequences, belonging to the validation set, along with its ground-truth mask is fed into the GAN to fine-tune the generator before feeding the remaining video frames sequentially to generate the corresponding predicted segmented mask. Fine-tuning aids in capturing the appearance of the specific object of interest required for foreground segmentation. For a fair comparison with the recent state-of-the-art methods, Online adaptation technique, with same parameter settings as used in (Voigtlaender and Leibe, 2017), has been adopted along with post-processing of the generated masks with a well-tuned Conditional Random Field (CRF) (Krähenbühl and Koltun, 2011) module. In online adaptation, the pixels of the foreground segmentation above a certain threshold with very confident predictions (positive samples) is considered for training the model. These pixels retain a memory of the positive class, which counterweights the negative samples added during training. Along with online adaptation, test time augmentation of the initial evaluation frames by random zooming, flipping and rotating is done to boost the performance of the network.

Ablation studies on DAVIS-2016

The performance of the proposed method is studied using variations of the proposed model and using the metric $\mathcal{J}_{mean}$ obtained in each of the cases, and shown in table 3.2. The removal of the encoder modules from the VidSeg-GAN framework exhibit a dip in performance of the network (see the first row of table 3.2) compared to the baseline model. This is because the encoder modules project the generated output and the ground-truth to a high-dimensional feature space, which increases the separability between them for better discrimination. We also omit the fine-tuning based on the first frame of the video sequences and evaluate the output on the validation set in an unsupervised setup. A substantial decrease in $\mathcal{J}_{mean}$ is noticed (row 2 of table 3.2) relying only on pre-trained ImageNet (Krizhevsky *et al.*, 2012) weights and DAVIS-2016 training data, thus making fine-tuning indispensable for expanding the tracking capabilities in the video sequences.

Table 3.2: Ablation studies of our proposed method on DAVIS-2016 dataset. One variation is made at a time, keeping the rest of the system intact, to observe the contribution of the respective module. The last row exhibits the result after adding online adaptation and CRF on the top of our baseline method. The result of the best configuration is shown in **bold**. The right-most column gives the $\mathcal{J}_{mean}$ difference ($\Delta\mathcal{J}_{mean}$) of performance of different settings in comparison with the baseline method (in row 6).

Aspect	System variant	$\mathcal{J}_{mean}$	$\Delta\mathcal{J}_{mean}$
Network	w/o Encoder	81.3	-1.2
Training	No fine-tuning	76.3	-6.2
	No DAVIS-2016 training	64.8	-17.7
	No pre-trained ImageNet weights	79.0	-3.5
Loss	Only $\mathcal{L}_1$ (not using $PSDL$)	80.1	-2.4
	Combined ($\mathcal{L}_1 + PSDL$) (baseline VidSeg-GAN)	82.5	-
Add-ons	VidSeg-GAN + Online Adaptation + CRF	**86.2**	+ 3.7

Again, relying only on pre-trained ImageNet weights and fine-tuning, while skipping the training on DAVIS-2016 (Perazzi *et al.*, 2016) dataset, shows a drastic drop in the value of $\mathcal{J}_{mean}$ as 64.8 (row 3 of table 3.2) in the performance of the model. Removing the pre-trained ImageNet weights results in a minor drop in $\mathcal{J}_{mean}$, owing to the loss of scale information. We argue that tracking a specific object in a video requires a reasonable amount of pre-knowledge which comes from pre-training the network on ImageNet (Deng *et al.*, 2009) dataset which consists of $\sim$10 M objects belonging to 1000 categories. Thus, these pre-trained weights assist the model in learning the general objectness prior, while the training set of DAVIS-2016 provides an advantage in evaluation by aiding the model to adapt the characteristics of the dataset. The initial frames of the validation set videos guide the network to track the specific object of interest throughout the sequences.

The proposed objective functions (equations 3.10 - 3.11) play an important role in the generation of segmentation masks in the sequence of video frames. Using only $\mathcal{L}_1$ loss produces holes in the segmented mask and some of them contain small blobs generated outside the region of interest, causing inaccurate segmentation. On the other hand, the proposed Patch-wise Symmetric Difference Loss (PSDL), combined with $\mathcal{L}_1$ produces impressive results (see figure 3.3), where the PSDL helps in removing the blob-like artifacts, thereby improving the contours of the output masks. We also add an online adaptation, with identical parameter settings as of (Voigtlaender and Leibe, 2017) along with augmentation of the initial test frame and a well-tuned post-processing CRF on

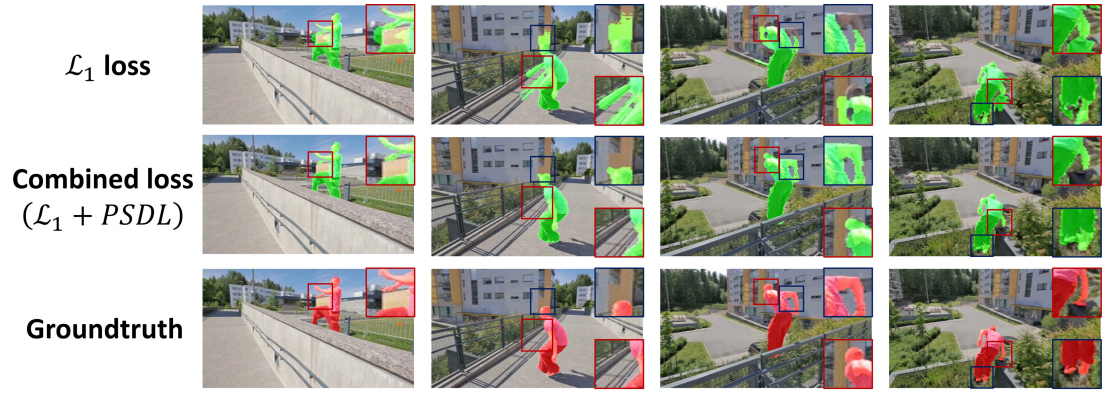


Figure 3.3: Qualitative comparison of segmentation generated by VidSeg-GAN on DAVIS-2016 dataset, using only $\mathcal{L}_1$ loss and Combined ($\mathcal{L}_1 + PSDL$) loss function. Cropped images in insets exhibit zoomed-in patches for better visibility of the estimated segmented masks in areas with occlusion, background clutter and significant motion blur (best viewed in color).

top of our proposed method to boost the $\mathcal{J}_{mean}$ value further. It is evident from table 3.2 that, each of the above factors is important and removing any one of them causes deterioration in performance. $\Delta\mathcal{J}_{mean}$ is used to measure the difference of performance of the various models in comparison with the baseline model (row 6 of table 3.2) in terms of the $\mathcal{J}_{mean}$ metric.

Quantitative analysis with existing methods

The major part of our experiments is performed on the DAVIS-2016 (Perazzi *et al.*, 2016) dataset, which consists of high-resolution video sequences with all of their frames annotated with pixel-level segmentation. For DAVIS-2016, three metrics: (i) region similarity in terms of Jaccard index ($\mathcal{J}$), (ii) contour accuracy ($\mathcal{F}$) and (iii) temporal (in-)stability of the segmented masks ($\mathcal{T}$), have been relied upon for evaluation. The validation set of DAVIS-2016 has been used to compute and compare the results among existing methods.

We compare our work quantitatively with a number of recent and state-of-the-art methods including semi-supervised techniques like OnAVOS (Voigtlaender and Leibe, 2017), MSK (Perazzi *et al.*, 2017), OSVOS (Caelles *et al.*, 2017), LT (Khoreva *et al.*, 2017), VPN (Jampani *et al.*, 2017), OFL (Tsai *et al.*, 2016) and BVS (Märki *et al.*, 2016). Unsupervised techniques like FSEG (Jain *et al.*, 2017), LMP (Tokmakov *et al.*, 2017) have also been taken into consideration during the comparison. We also include two informative bounds which depict the quality that an oracle would reach by selecting the best segmented object proposal out of two state-of-the-art-techniques (COB (Maninis

et al., 2016) and MCG (Pont-Tuset *et al.*, 2017*a*)). The quantitative analysis of different techniques on the validation set of DAVIS-2016 is shown in table 3.3. In terms of region similarity $\mathcal{J}$, our baseline method (*VidSeg-GAN w/o adapt*) beats all other existing techniques except OnAVOS (Voigtlaender and Leibe, 2017) which uses online adaptation, test time augmentation and CRF. On using those add-ons on the top of our base model (*VidSeg-GAN adapt*), the result obtained surpasses OnAVOS (Voigtlaender and Leibe, 2017). In terms of contour accuracy $\mathcal{F}$, though *VidSeg-GAN adapt* performs better than OnAVOS (Voigtlaender and Leibe, 2017) in $\mathcal{F}_{recall}$ measure, still it remains slightly less than that of OSVOS (Caelles *et al.*, 2017). In case of temporal (in-)stability $\mathcal{T}$, in spite of not using temporal information, VidSeg-GAN shows impressive performance (refer row 7, in table 3.3). In case of both $\mathcal{J}_{decay}$ and $\mathcal{F}_{decay}$, though our proposed method beats the semi-supervised techniques, it fails in comparison to the unsupervised ones (Jain *et al.*, 2017; Tokmakov *et al.*, 2017). The overall performance of VidSeg-GAN is better than the existing semi-supervised and unsupervised techniques in terms of $\mathcal{J}_{mean}$, $\mathcal{J}_{recall}$ and $\mathcal{F}_{mean}$. Our method also shows impressive results compared to those obtained by an oracle selecting the best object proposal from state-of-the-art object proposals: COB (Maninis *et al.*, 2016) and MCG (Pont-Tuset *et al.*, 2017*a*) (refer rows 1-2, 4-5 and 7, in table 3.3). The *Patch-wise Symmetric Difference Loss (PSDL)* used in VidSeg-GAN can be accounted for the success of our base model which has outperformed majority of the existing state-of-the-art techniques by minimizing the number of mispredicted pixels in segmentation. Thus, it not only increases the $\mathcal{J}_{mean}$ value, but also improves the segmented contour by working on small patches. From table 3.3, it is evident that our base VidSeg-GAN model performs better than all other existing techniques, including OnAVOS (Voigtlaender and Leibe, 2017) with no online adaptation, which clarifies the effectiveness of our proposed *PSDL* function.

Table 3.3: Quantitative analysis of VidSeg-GAN with other existing state-of-the-art techniques on DAVIS-2016 validation set. The comparison results of other methods are quoted from the respective previous works and (Perazzi *et al.*, 2016, 2017). Best results are represented in **bold**. Here, "w/o adapt" denotes 'without' and "adapt" represents 'with' online adaptation. ↑: 'higher the value better'; ↓: 'lower the value better'.

Measure	Semi-supervised										Unsupervised		Bounds	
	VidSeg-GAN (ours)		OnAVOS		MSK	OSVOS	LT	VPN	OFL	BVS	FSEG	LMP	COB	MCG
	w/o adapt	adapt	w/o adapt	adapt										
$\mathcal{J}_{mean}\uparrow$	82.5	**86.2**	81.7	85.7	80.3	79.8	80.5	70.2	68.0	60.0	70.7	70.0	79.3	70.7
$\mathcal{J}_{recall}\uparrow$	93.4	**96.3**	92.2	95.4	93.5	93.6	-	82.3	75.6	66.9	83.5	85.0	94.4	91.7
$\mathcal{J}_{decay}\downarrow$	8.1	6.5	11.9	7.1	8.9	14.9	-	12.4	26.4	28.9	1.5	**1.3**	3.2	**1.3**
$\mathcal{F}_{mean}\uparrow$	81.4	**84.9**	81.1	84.2	75.8	80.6	-	65.5	63.4	58.8	65.3	65.9	75.7	62.9
$\mathcal{F}_{recall}\uparrow$	91.5	92.4	88.2	88.7	88.2	**92.6**	-	69.0	70.4	67.9	73.8	79.2	88.5	76.7
$\mathcal{F}_{decay}\downarrow$	9.3	6.8	11.2	7.8	9.5	15.0	-	14.4	27.2	21.3	**1.8**	2.5	3.9	1.9
$\mathcal{T}_{mean}\downarrow$ (GT 8.8)	25.2	20.7	27.3	**18.5**	18.6	37.8	-	32.4	22.2	34.7	32.8	57.2	44.1	69.8

OnAVOS: (Voigtlaender and Leibe, 2017)

MSK: (Perazzi *et al.*, 2017)

OSVOS: (Caelles *et al.*, 2017)

LT: (Khoreva *et al.*, 2017)

VPN: (Jampani *et al.*, 2017)

OFL: (Tsai *et al.*, 2016)

BVS: (Märki *et al.*, 2016)

FSEG: (Jain *et al.*, 2017)

LMP: (Tokmakov *et al.*, 2017)

COB: (Maninis *et al.*, 2016)

MCG: (Pont-Tuset *et al.*, 2017*a*)

Table 3.4: Video object segmentation results of VidSeg-GAN in comparison with other existing methods on YouTube-Objects and SegTrack-v2 datasets. Results are compared with (Caelles *et al.*, 2017; Perazzi *et al.*, 2017; Khoreva *et al.*, 2017; Voigtlaender and Leibe, 2017) methods. Best results are in **bold**.

Method	Dataset, $\mathcal{J}_{mean}$	
	YouTube-Objects	SegTrack-v2
Grabcut oracle	67.6	74.2
BVS (Märki *et al.*, 2016)	59.7	58.4
OFL (Tsai *et al.*, 2016)	70.1	67.5
TRS (Xiao and Jae Lee, 2016)	-	69.1
OSVOS (Caelles *et al.*, 2017)	72.5	65.4
Masktrack (Perazzi *et al.*, 2017)	72.6	70.3
LucidTracker (Khoreva *et al.*, 2017)	76.2	77.6
OnAVOS, no adaptation (Voigtlaender and Leibe, 2017)	76.6	-
OnAVOS, online adaptation (Voigtlaender and Leibe, 2017)	77.4	-
VidSeg-GAN (ours)	76.9	76.5
VidSeg-GAN, online adaptation (ours)	**77.8**	**77.7**

For a complete evaluation of our method, we have also experimented on SegTrack-v2 (Li *et al.*, 2013) and YouTube-Objects (Prest *et al.*, 2012) datasets and compared our results (refer table 4.2) with OFL (Tsai *et al.*, 2016), BVS (Märki *et al.*, 2016), TRS (Xiao and Jae Lee, 2016), OSVOS (Caelles *et al.*, 2017), MSK (Perazzi *et al.*, 2017), LucidTracker (Khoreva *et al.*, 2017) and OnAVOS (Voigtlaender and Leibe, 2017) methods. The pre-computed evaluation results are obtained from the respective previous works. Due to the lack of proper training set in YouTube-Objects dataset, the same parameters as of DAVIS-2016 are used, and the pre-training step on DAVIS-2016 training set is removed to evaluate the generalization capability of our method. While evaluating on this dataset, we have been consistent with (Khoreva *et al.*, 2017) i.e. the frames in which the object of interest are absent has also been included. Both VidSeg-GAN base network (76.9 $\mathcal{J}_{mean}$) and its online adapted version (77.8 $\mathcal{J}_{mean}$) along with test time augmentation, post-processing CRF performed better than OnAVOS in the respective cases (refer table 3.4).

Assessment on the SegTrack-v2 dataset is performed following the similar choice of settings as done in recent existing methods (Perazzi *et al.*, 2017; Khoreva *et al.*, 2017). Same protocols as of DAVIS-2016 evaluation are carried out by fine-tuning on the initial frame of the test video sequence. Table 3.4 shows that our base network performs better than MaskTrack (Perazzi *et al.*, 2017), despite the fact that the sequences

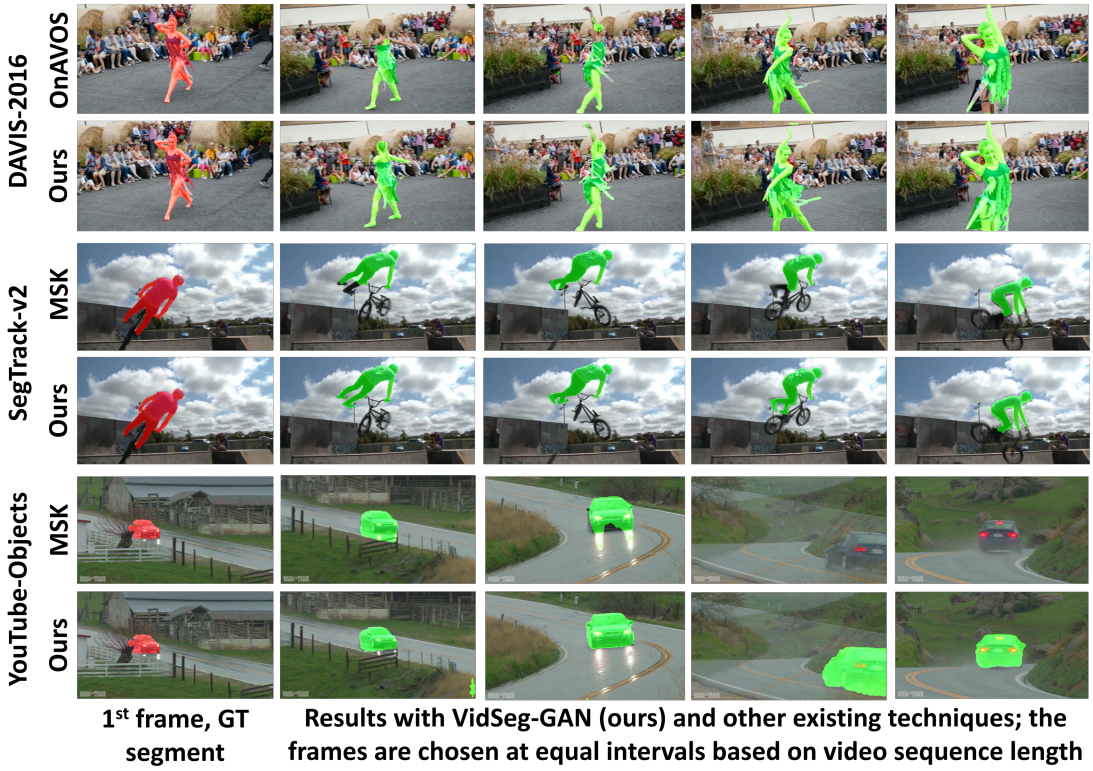


Figure 3.4: Qualitative results of the online adaptive version of our proposed method on three real-world datasets exhibit impressive results in challenging situations like change of appearance, occlusions, camera view change, background clutter and motion blur, when compared to OnAVOS (Voigtlaender and Leibe, 2017) and MSK (Perazzi *et al.*, 2017) methods (best viewed in color).

in this dataset have significantly less motion and occlusion than in DAVIS-2016, which favour techniques that use temporal information. LucidTracker (Khoreva *et al.*, 2017) outperforms our VidSeg-GAN base model by a narrow margin. However, online adaptive version of our model, enabled with test time augmentation and CRF module, shows dominating result over all the techniques, including LucidTracker (Khoreva *et al.*, 2017). Figure 3.4 shows the comparative study of the qualitative results of our *online adapted* VidSeg-GAN with OnAVOS (Voigtlaender and Leibe, 2017) and MaskTrack (Perazzi *et al.*, 2017) on three popular real-world datasets, where our method performs well in segmenting the specific object of interest under difficult conditions like background clutter, viewpoint change, motion blur, occlusions and shape deformation of object.

In figure 3.4, it is evident from the comparative results on DAVIS-2016 dataset (first 2 rows) that OnAVOS (Voigtlaender and Leibe, 2017) suffers from background clutter, which leads to degradation in performance of segmentation of the upper portion of the body of the dancer in frames 2 and 3. It also fails to segment the lower portion of the body after change of pose of the object of interest in the last frame. Whereas,

Figure 3.5: Segmentation results of our proposed VidSeg-GAN model on three benchmark real-world datasets *viz.* DAVIS-2016 (Perazzi *et al.*, 2016), SegTrack-v2 (Li *et al.*, 2013) and YouTube-Objects (Prest *et al.*, 2012).

our VidSeg-GAN exhibits superior performance in the following frames owing to its encoder-decoder arrangement in the generator module and the proposed Patch-wise Symmetric Difference Loss (PSDL), which improves the result by segmenting intricate details in the objects. In SegTrack-v2 dataset (refer figure 3.4, row 3 and 4), MSK (Perazzi *et al.*, 2017) shows suboptimal performance in segmenting the lower portion of the cyclist due to the random motion involved in the video which affects the trajectory flow information. VidSeg-GAN performs comparatively better than MSK because it processes each frame independently leading to superior per-frame segmentation of the object of interest. Similarly, the trajectory flow information of MSK (Perazzi *et al.*, 2017) gets disrupted with the sudden change of scene and viewpoint in frames 4 and 5 (see figure 3.4, second last row) leading to failure in segmenting the car in the latter part of the video. On the other hand, VidSeg-GAN produces optimal result (figure 3.4, last row) by segmenting each frame individually. All these cases are examples which improve the performance of our method over the existing ones in terms of region similarity ($\mathcal{J}$) and contour accuracy ($\mathcal{F}$), depending on the quality of segmentation (refer tables 3.3 and 3.4). More qualitative results of VidSeg-GAN on the three real-world datasets are shown

in figure 3.5. Qualitative results in the form of video-clips for the proposed VidSeg-GAN model are available in (VidSeg-GAN_Res).

3.4 Summary

This chapter discusses a modified Generative Adversarial Network (GAN), which is done by the implementation of a skip-connection enabled encoder-decoder type architecture in the generator module, for the task of Video Object Segmentation. An additional encoder module is introduced to bring variation in the training of the discriminator. Inclusion of a novel Patch-wise Symmetric Difference Loss (PSDL) also improves the segmentation results significantly over the existing and state-of-the-art methods. To the best of our knowledge, this is the first implementation of a GAN based framework for the task of video object segmentation. However, there are scope of improvements as the model relies on specific tailored objective functions and fails to produce a satisfactory performance in maintaining temporal consistency among the long-range segmented masks due to the absence of explicit trajectory flow-based information.

CHAPTER 4

TempSeg-GAN: Adversarial Segmention of Objects in Videos using Temporal Information

This chapter provides a temporal information-based approach using Generative Adversarial Networks (GAN) to solve the problem of segmenting object(s) from videos. The GAN based framework, described in chapter 3, introduced a hand-crafted heuristic-based objective function, Patch-wise Symmetric Difference Loss (PSDL), which although produced better results than the state-of-the-art, had its drawback in the form of its inability to maintain the temporal coherency among long-distant segmented masks. This can be accounted for due to the absence of an explicit trajectory flow-based information in the VidSeg-GAN model. The method described in this chapter overcomes this limitation partly by incorporating optical flow as a temporal information in the formulation of two novel objective functions: (a) *Inter-frame Temporal Symmetric Difference Loss (ITSDL)*, and (b) *Intra Frame Temporal Loss (IFTL)*. ITSDL not only improves the segmentation quality but also maintains the temporal stability of the motion features generated by the network, with the help of optical flow vectors. The Intra Frame Temporal Loss (IFTL) complements the ITSDL by maintaining the temporal consistency among the generated masks on a global scale. The overall proposed system of segmenting objects in videos mimics an encoder-decoder like arrangement via inclusion of the novel loss functions for training the GAN (Goodfellow *et al.*, 2014) architecture. As our work deals with videos instead of arbitrary data distributions, the input of the proposed model is a sequence of video frames, while the output is segmentation mask of the object(s) corresponding to the RGB input frames. The end-to-end trainable model exhibits improved performance compared to the state-of-the-art on three real-world benchmark video object segmentation datasets.

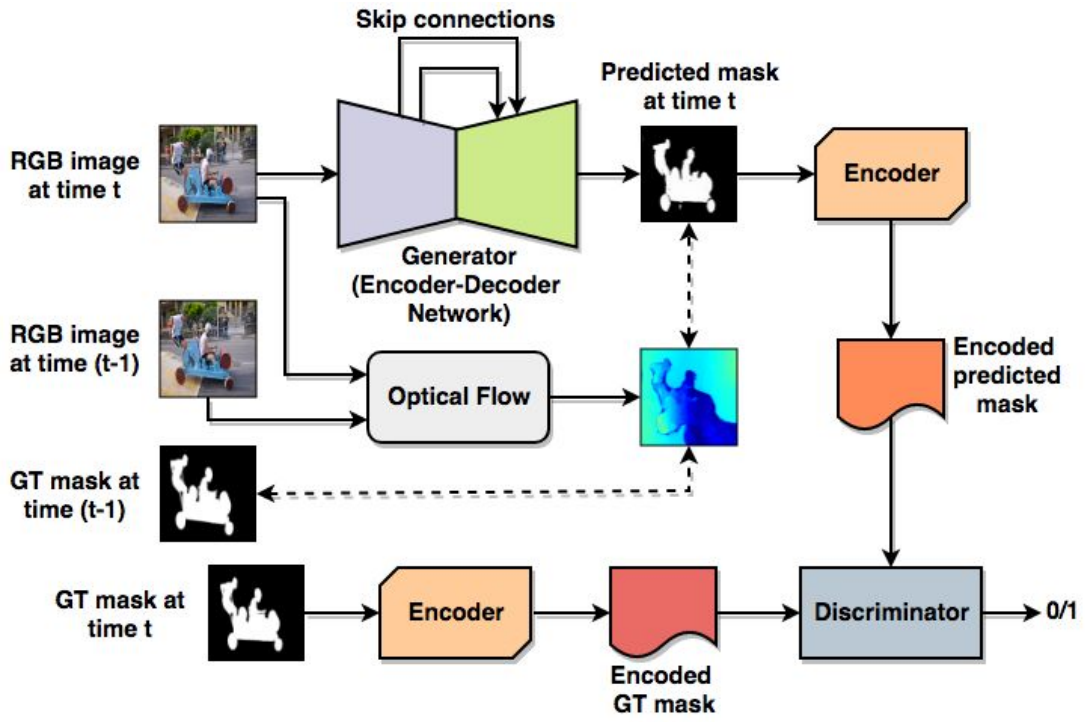


Figure 4.1: Proposed TempSeg-GAN framework. GT denotes the ground-truth and ←- - -→ refers to the Inter-frame Temporal Symmetric Difference Loss ($\mathcal{L}_{ITSDL}$) estimation using GT mask at time t, predicted mask at time $(t-1)$ and optical flow vectors between RGB input images at time $(t-1)$ and t.

4.1 Temporally aided Segmentation Network

The proposed architecture of Temporally aided Segmentation GAN (TempSeg-GAN) is illustrated in figure 4.1. The framework follows the VidSeg-GAN network as discussed earlier in section 3.1, with the addition of an optical flow vector generation module inspired from *FlowNet 2.0* (Ilg *et al.*, 2017), employed to capture the motion patterns existing between the consecutive frames of the video sequences. Though theoretically the process of alternative optimization of generator and discriminator is logically firm and well-established, but in reality it is prone to mode collapse leading to instabilities in training. This implicit instability is overcome by the proposed two novel loss functions based on Intersection-over-Union (IoU) and optical flow vectors (discussed later in sections 4.2, 4.3) to produce improved segmented masks as output. These objective functions are used in addition to the conventional GAN losses. The flow vectors, required for formulating the proposed loss functions, are obtained by passing consecutive RGB input frames through the optical flow generation module during training of the network.

Algorithm 4.1: $Optical\_Flow\_Warp(P_{t-1}, W^*_{t-1}, \hat{P}_t)$

Input: Ground-truth mask patch (P_{t-1}) at time $t-1$, Predicted mask patch ($\hat{P}_t$) at time t, Optical flow vector map patch (W^*_{t-1}).

Output: Optical flow warped ground-truth mask patch (P^*_t) at time t of dimension same as $\hat{P}_t$.

// s = height and width of $\hat{P}_t$

// $s+4$ = height and width of P_{t-1} and W^*_{t-1}

Initialize: patch P^*_t with each pixel value equal to 0.;

for $u = 0$ *to* $s+4$, $i \leftarrow u+1$ **do**

  for $v = 0$ *to* $s+4$, $j \leftarrow v+1$ **do**

    /* (u, v) is the spatial location of pixel at time $(t-1)$. */

    $u' \leftarrow u + V_u \Delta t$;

    $v' \leftarrow v + V_v \Delta t$;

    /* V_u and V_v are horizontal and vertical flow vectors of (u, v) obtained from W^*_{t-1}, (u', v') is the new spatial location of (u, v) at time t. Here, $\Delta t = 1$. */

    Assign the label of (u', v') same as that of (u, v) using ground-truth mask;

    /* The label of (u', v') is updated with that of (u, v). */

    if (u', v') *lies within the patch* P^*_t **then**

      Update corresponding pixel value of P^*_t with label of (u', v').

    end

  end

end

4.2 Inter-frame Temporal Symmetric Difference Loss

Unlike images, the advantage of the videos lies in the fact that it provides a latent space of data distribution by combining the temporal information with the spatial one. CNNs are capable of capturing short-range consistencies in the spatial domain, which only forms a small part of the rich input data. Thus, to maintain the temporal coherency between the masks along with enhancing the segmentation quality, an Intersection-over-Union (IoU) based temporal objective function has been incorporated. It measures the region of similarity between the predicted mask ($\hat{Y}$) and the ground-truth mask (Y), by computing the number of overlapping pixels. In other words, it gives an idea of the number of mispredicted pixels present in the estimated mask compared to the ground-truth. To formulate the function in a patch-wise manner, we calculate the number of mispredicted pixels ($|M_p|$) in a patch of the segmented mask. The set of mispredicted pixels (M_p) for class $c \in \{0, 1\}$ obtained by using the symmetric difference of the two vectors $\boldsymbol{p}$ and $\hat{\boldsymbol{p}}$,

is as follows (similar to equation 3.8):

$$\boldsymbol{M}_p(\boldsymbol{p}, \hat{\boldsymbol{p}}) = \{\boldsymbol{p}_k = c, \hat{\boldsymbol{p}}_k \neq c\} \cup \{\boldsymbol{p}_k \neq c, \hat{\boldsymbol{p}}_k = c\} \quad \forall k = 1, \cdots, s^2 \tag{4.1}$$

where, $\hat{\boldsymbol{p}}$ denotes the vector of predicted labels in the patch $\hat{P}$, with top-left pixel index (i, j), of the estimated mask ($\hat{Y}$) and $\boldsymbol{p}$ is the vector of ground-truth labels in the corresponding patch P of the target mask (Y); $\boldsymbol{p}, \hat{\boldsymbol{p}} \in \{0, 1\}^{s^2}$; s is the height and width of the patches P and $\hat{P}$.

In Video Object Segmentation, we implement the symmetric difference by extracting non-overlapping patches of dimension $s \times s$ ($1 < s \leq 4$), represented by $\hat{P}_t\{i, j, s\}$, where (i, j) is the top-left index of the patch, from the predicted mask at time t and then evaluating the number of mislabelled pixels with the corresponding target patch $P^*_t\{i, j, s\}$ at same time t. The mechanism of formation of the target patch from the ground-truth patch at time $(t-1)$, denoted by $P_{t-1}\{i-2, j-2, s+4\}$, by warping with optical flow vector patch $W^*_{t-1}\{i-2, j-2, s+4\}$, at time $(t-1)$, is explained step-wise in Algorithm 4.1.

In simpler terms, the symmetric difference score is calculated between small portions of the predicted mask and the corresponding optical flow warped ground-truth mask. It is assumed that the motion features are effectively transferred from the ground-truth mask of the previous time step to the warped target mask of the current time step using optical flow vectors. Thus the motion-related features can be well approximated with the low-resolution patches both in the spatial as well as temporal domains. The smoothness of the features is also guaranteed, unless there is a sudden change of scene or rapid movement in the videos. To enhance the attention over small disjoint sections, formed as a result of occlusions in the segmented masks, the loss function is computed in a patch-wise manner. The artefacts, produced as a part of the network output, are often ignored when estimated globally.

The Inter-frame Temporal Symmetric Difference score for each patch ($ITSD_{patch}$) of a predicted mask with top-left index (i, j) is denoted as:

$$ITSD_{patch} = \frac{|\boldsymbol{M}_{p^*}|}{|\{\hat{\boldsymbol{p}}_k = 1\} \cup \boldsymbol{M}_{p^*}|} \tag{4.2}$$

which computes a ratio between the mispredicted pixels and the total number of pixels

for the patch with top-left pixel index (i, j) (refer to line 11 in algorithm 4.2).

Algorithm 4.2: Inter-frame Temporal Symmetric Difference score to estimate the similarity between optical flow warped ground-truth mask(s) and predicted mask(s).
$extract\_patch(X_t, a, b, d)$ is used to extract a patch of dimension $d \times d$ starting from the top-left pixel index (a,b) of the frame X_t at time t.

Input: Ground-truth masks (Y), Predicted masks ($\hat{Y}$), Optical flow vector maps (W)
Output: Inter-frame Temporal Symmetric Difference score ($Score_{ITSD}$)

// $s =$ height and width of an patch on the mask
// $S =$ height and width of the masks
// $t =$ current time
// $T =$ Number of masks predicted

Initialize: $Score_{ISD} = 0$;
for $t = 1$ *to* T **do**
  for $i = 0$ *to* S, $i \leftarrow i + s$ **do**
    for $j = 0$ *to* S, $j \leftarrow j + s$ **do**
      $\hat{P}_t \leftarrow extract\_patch(\hat{Y}_t, i, j, s)$;
      $P_{t-1} \leftarrow extract\_patch(Y_{t-1}, i-2, j-2, s+4)$;
      $W^*_{t-1} \leftarrow extract\_patch(W_{t-1}, i-2, j-2, s+4)$;
      $P^*_t \leftarrow optical\_flow\_warp(P_{t-1}, W^*_{t-1}, \hat{P}_t)$ (refer to Algo. 4.1);
      $\hat{\boldsymbol{p}} \leftarrow$ vector of predicted labels of pixels obtained from the patch $\hat{P}_t$;
      $\boldsymbol{p}^* \leftarrow$ vector of labels of pixels obtained from the patch P^*_t;
      $\boldsymbol{M}_{p^*} \leftarrow \{\boldsymbol{p}^*_k = 1, \hat{\boldsymbol{p}}_k = 0\} \cup \{\boldsymbol{p}^*_k = 0, \hat{\boldsymbol{p}}_k = 1\}, \quad \forall k = 1, \cdots, s^2$ (see equation 4.1);
      /* $\boldsymbol{M}_{p^*}$ denotes the set of mislabelled pixels in the $\hat{\boldsymbol{p}}$ compared with $\boldsymbol{p}^*$ */
      $Score_{ITSD} \leftarrow Score_{ITSD} + \frac{|\boldsymbol{M}_{\boldsymbol{p}^*}|}{|\{\hat{\boldsymbol{p}}=1\} \cup \boldsymbol{M}_{\boldsymbol{p}^*}|}$ (see equation 4.2);
    end
  end
  $Score_{ITSD} \leftarrow {}^{Score_{ITSD}}/_{\lfloor S/s \rfloor^2}$; // Average over all the patches
end
$Score_{ITSD} \leftarrow {}^{Score_{ITSD}}/_{T}$; // Average over all the masks

The evaluation procedure of ITSD score by matching the corresponding local patches of the estimated and optical flow warped masks is explained in a step-wise manner in algorithm 4.2. The objective function modelled by calculating the ITSD score for the generator ($\mathbb{G}$) network not only minimizes the score for batch inputs (during training) but also maintains the temporal data distribution by stabilizing the motion features generated by the network with the help of flow vectors. The loss function, $\mathcal{L}_{ITSDL}$ is defined as:

$$\mathcal{L}_{ITSDL}(Y, \hat{Y}) = Score_{ITSD}(Y, \hat{Y}) \tag{4.3}$$

where, $\hat{Y}$ and Y represent the predicted and optical flow warped mask, and $Score_{ITSD}$ denotes the mean symmetric difference score over all the masks, obtained using the process mentioned in algorithm 4.2.

4.3 Intra Frame Temporal Loss

The ITSD Loss, mentioned in section 4.2, estimates the motion features that change slowly with respect to time using the local symmetric difference measures, which in turn also enhances the segmentation quality. Thus to maintain the temporal relationship between frames globally, we introduce the idea of Intra Frame Temporal Loss over the network output masks. A few works (Goroshin *et al.*, 2015; Mobahi *et al.*, 2009) in the recent past exploits the idea of the temporal coherence to learn the motion features. Assuming slow variation of motion features over time, we consider two consecutive frames $\hat{Y}_t$ and $\hat{Y}_{t+1}$ as a temporal pair, where $\hat{Y}_t$ and $\hat{Y}_{t+1}$ are the TempSeg-GAN generated output masks at time t and $t+1$ respectively with q_t and q_{t+1}, as the value of the discriminator ($\mathbb{D}$) outputs, being equal to 1 for both the masks. The slow variation of motion features is modelled through an objective function as:

$$\begin{aligned}\mathcal{L}_{IFTL}(\hat{Y}, \overrightarrow{q}) &= \sum_{t=0}^{T-1} d_{\delta}(\hat{Y}_t, \hat{Y}_{t+1}, q_t \times q_{t+1}) \\ &= \sum_{t=0}^{T-1} \Big(q_t \times q_{t+1} \times d(\hat{Y}_t, \hat{Y}_{t+1}) \\ &\quad + (1 - q_t \times q_{t+1}) \times \max(0, \delta - d(\hat{Y}_t, \hat{Y}_{t+1})) \Big)\end{aligned} \tag{4.4}$$

where, T is the total time duration of the masks generated by the network, $q_t \in \{0, 1\}$ gives the value of the discriminator output, $d(x, y)$ is the measure for Euclidean distance and δ is a positive constant. Thus speaking in simpler terms, equation 4.4 minimizes the intra-frame distance between the predicted masks which have been generated correctly while penalizing the disparity between the incorrectly predicted frames with a positive margin δ.

Long-range Intra Frame Temporal Loss: Though the IFTL maintains the temporal consistency between the consecutive frames, it does not guarantee the same for the long term frames. Thus, to keep the stability intact in the spatio-temporal feature space

a Long-range Intra Frame Temporal Loss (L-IFTL) is incorporated by extending the IFTL function as an estimation of the distance between initial predicted mask ($\hat{Y_0}$) and all other predicted masks ($\hat{Y_t}$) at time $t(> 0)$. The proposed loss is defined as:

$$\begin{aligned}\mathcal{L}_{L-IFTL}(\hat{Y}, \overrightarrow{q}) &= \sum_{t=1}^{T} d_\delta(\hat{Y_0}, \hat{Y_t}, q_0 \times q_t) \\ &= \sum_{t=1}^{T} \Big(q_0 \times q_t \times d(\hat{Y_0}, \hat{Y_t}) \\ &\qquad + (1 - q_0 \times q_t) \times \max(0, \delta - d(\hat{Y_0}, \hat{Y_t})) \Big)\end{aligned} \tag{4.5}$$

where, the symbols have the same meaning as in equation 4.4.

Thus, L-IFTL preserves the temporal coherency among the distant frames by estimating the distance between the initial and rest of the generated frames.

4.4 Multi-Component Objective Function

Finally, the overall objective function is formed by combining the loss functions given in equations 4.3 - 4.5 with the adversarial loss (refer equation 3.4) and the traditional $\mathcal{L}_{L_1}$ (L_1 loss) objective with respective weights as follows:

$$\begin{aligned}\mathcal{L}_{combined} = \alpha_{adv}\mathcal{L}^{\mathbb{G}}_{adv}(I) + \alpha_{L_1}\mathcal{L}_{L_1}(Y, \hat{Y}) + \alpha_{ITSDL}\mathcal{L}_{ITSDL}(Y, \hat{Y}) \\ + \alpha_{IFTL}\mathcal{L}_{IFTL}(\hat{Y}, \overrightarrow{q}) + \alpha_{L-IFTL}\mathcal{L}_{L-IFTL}(\hat{Y}, \overrightarrow{q})\end{aligned} \tag{4.6}$$

where, the weights *viz.* α_{L_1}, α_{ITSDL}, α_{IFTL} and α_{L-IFTL} are set to 0.25 while α_{adv} is kept at 0.1 (all values obtained empirically, for best performance). This combined loss is minimized during the training of TempSeg-GAN using Adam optimizer (Kingma and Ba, 2014).

4.5 Experimental Results and Discussions

In this section, the performance of our proposed TempSeg-GAN model is analyzed on three popular benchmark real-world datasets for the task of video object segmentation. Sequences of video frames along with their corresponding ground-truth masks are used

for training the framework. During validation, the network generates segmented masks of the object of interest when provided with frames of video sequences as input. To specify the particular object to be segmented, the model is fine-tuned with first two annotated frames of the videos. The network generated segmented masks are again used by our proposed architecture as reference masks to produce the predicted masks of the next time steps. Optical flow vectors are also calculated using *FlowNet 2.0* (Ilg *et al.*, 2017) between the consecutive frames, which helps in formulating the ITSD Loss (described in section 4.2). Three metrics: Region similarity ($\mathcal{J}$), Contour accuracy ($\mathcal{F}$) and Temporal (in-)stability ($\mathcal{T}$) (Perazzi *et al.*, 2016), have been used to evaluate the results of our network with the existing state-of-the-art methods.

4.5.1 Performance Analysis of Video Object Segmentation

The proposed TempSeg-GAN model is first pre-trained on the ImageNet (Deng *et al.*, 2009) dataset. 30 sequences of annotated video frames of DAVIS-2016 (Perazzi *et al.*, 2016) dataset are used for training. Random rotation, flipping and zooming of frames are done as a part of data augmentation, in such a way that the temporal stability among the frames remains undisturbed. During validation, the network is first fine-tuned with two initial annotated frames of the test video followed by generation of segmentation masks for the remaining frames of the video, when provided with the rest of the RGB frames sequentially as input. The appearance of the specific object of interest to be segmented is captured during fine-tuning. Also, to maintain the temporal relationship between the predicted masks, the network-output masks of previous time steps are used as reference by the framework, during generation of next time step masks. For a fair comparative study of results with the recent and state-of-the-art methods, a well-tuned Conditional Random Field (CRF) (Krähenbühl and Koltun, 2011) is used as a post-processing module on top of our method. To enhance the quality of the segmented masks, augmentation of the annotated frames is performed, at the fine-tuning stage (Test-time augmentation). TempSeg-GAN++ refers to the modified version of our baseline model with the above-mentioned add-ons attached to it.

Ablation studies on DAVIS-2016

The performance of the proposed model has been studied by varying the architecture and

Table 4.1: Ablation studies of our proposed method on DAVIS-2016 dataset. Keeping the entire system intact, one variation is made at a time to observe the contribution of the respective module. The last row exhibit the result after adding test-time augmentation and CRF on the top of our baseline method. The result of the best configuration is shown in **bold**. The right-most column gives the $\mathcal{J}_{mean}$ difference ($\Delta\mathcal{J}_{mean}$) of performance of different settings in comparison with the baseline method (in row 6).

Aspect	System variant	$\mathcal{J}_{mean}$	$\Delta\mathcal{J}_{mean}$
Training	w/o fine-tune	76.8	-8.3
	w/o DAVIS-2016 training	68.7	-16.4
	w/o ImageNet weights	79.3	-5.8
Loss	$\mathcal{L}_1$ loss	81.2	-3.9
	$\mathcal{L}_1 + ITSDL$	84.6	-0.5
	TempSeg-GAN (baseline model)	85.1	-
Add-ons	TempSeg-GAN++	**86.3**	+ 1.2

the experimental setup. The results obtained in different cases are exhibited in table 4.1. First, the output on the validation set is studied in an unsupervised setup by removing the fine-tuning based on the initial annotated frames of the videos. A significant decrease in $\mathcal{J}_{mean}$ was noticed depending only on pre-trained ImageNet (Krizhevsky *et al.*, 2012) weights and DAVIS-2016 training data, thus making fine-tuning indispensable for expanding the tracking capabilities in the video sequences. Again, skipping the training on DAVIS-2016 (Perazzi *et al.*, 2016) dataset and relying only on fine-tuning and pre-trained ImageNet weights show a substantial drop (68.7 $\mathcal{J}_{mean}$) in the performance of the model. Loss of scale information is observed, resulting in a decrease in $\mathcal{J}_{mean}$ on the removal of ImageNet pre-trained weights. Thus it is evident that segmenting a specific object in a sequence of frames requires a reasonable amount of pre-knowledge which comes from pre-training the network on ImageNet (Deng *et al.*, 2009) dataset which consists of $\sim$10 M objects belonging to 1000 categories. The general objectness prior is learnt by TempSeg-GAN from the pre-trained weights, while the training set of DAVIS-2016 aids the model to adapt the dataset characteristics. The initial frames of the validation set videos guide the network to track the specific object of interest throughout the sequences.

The proposed objective functions play an important role in the generation of segmentation masks in the sequence of video frames. Using only $\mathcal{L}_1$ loss produces holes in the segmented mask and some of them contain small blobs generated outside the region of interest, causing inaccurate segmentation. On the other hand, the proposed Inter-frame Temporal Symmetric Difference Loss (ITSDL), combined with $\mathcal{L}_1$ and Intra

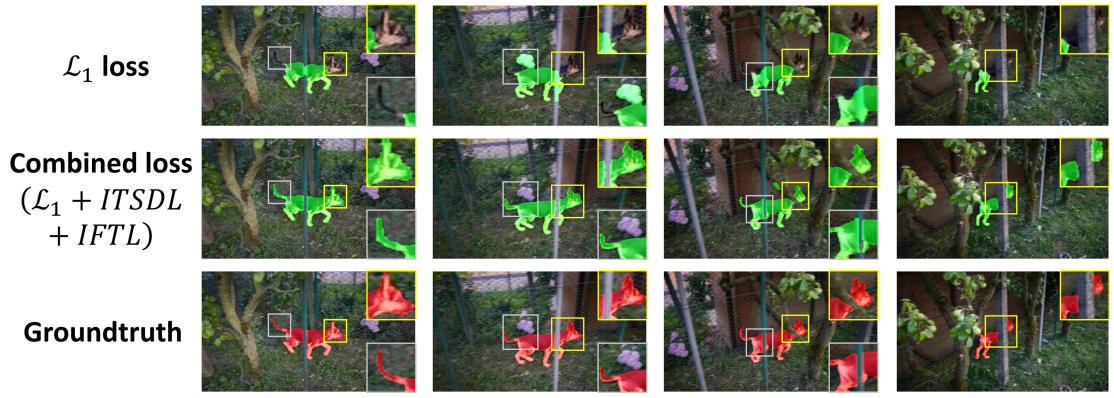


Figure 4.2: Comparative study of predicted segmentation results on DAVIS-2016 dataset obtained using our proposed TempSeg-GAN models, with only $\mathcal{L}_1$ loss and Combined ($\mathcal{L}_1 + ITSDL + IFTL$) loss (refer equation 4.6) function. Figures in insets show zoomed-in patches for better visibility of the estimated segmented masks in areas with background clutter, occlusion and significant motion blur (best viewed in color).

Frame Temporal Loss (IFTL) produces impressive results (see figure 4.2), where the ITSDL helps in removing the blob-like artifacts, thereby improving the contours of the output masks. We also add a well-tuned post-processing CRF on top of our proposed method along with augmentation of initial frames during the fine-tuning stage to boost the $\mathcal{J}_{mean}$ value further. It is evident from table 4.1 that, each of the above factors is important and removing any one of them degrades the results quantitatively as well as qualitatively.

Quantitative analysis with existing methods

The major part of our experiments are performed on the DAVIS-2016 (Perazzi *et al.*, 2016) dataset, which consists of high-resolution video sequences with all of their frames annotated with pixel-level segmentation. For DAVIS-2016, three metrics: (i) region similarity in terms of mean Jaccard index ($\mathcal{J}_{mean}$), (ii) mean contour accuracy ($\mathcal{F}_{mean}$) and (iii) mean temporal (in-)stability of the segmented masks ($\mathcal{T}_{mean}$), have been relied upon for evaluation. The validation set of DAVIS-2016 has been used for computation and comparison purposes.

The proposed method is compared with a few recent and state-of-the-art semi-supervised methods like OnAVOS (Voigtlaender and Leibe, 2017), Masktrack (Perazzi *et al.*, 2017), OSVOS (Caelles *et al.*, 2017), LT (Khoreva *et al.*, 2017), CINM (Bao *et al.*, 2018), RGMP (Wug Oh *et al.*, 2018), FAVOS (Cheng *et al.*, 2018), OFL (Tsai *et al.*, 2016)

Table 4.2: Quantitative analysis of TempSeg-GAN with other existing semi-supervised methods on DAVIS-2016 validation set, YouTube-Objects and SegTrack-v2 datasets. Other results used for comparison are quoted from the respective previous works. Best results are in **bold**. Values underlined represents the next best results. ↑: 'higher the value better'; ↓: 'lower the value better'.

Method	DAVIS-2016			YouTube-Objects	SegTrack-v2
	$\mathcal{J}_{mean}\uparrow$	$\mathcal{F}_{mean}\uparrow$	$\mathcal{T}_{mean}\downarrow$	$\mathcal{J}_{mean}\uparrow$	$\mathcal{J}_{mean}\uparrow$
BVS (Märki *et al.*, 2016)	60.0	58.8	34.7	59.7	58.4
OFL (Tsai *et al.*, 2016)	68.0	63.4	22.2	70.1	67.5
OSVOS (Caelles *et al.*, 2017)	79.8	80.6	37.8	72.5	65.4
Masktrack (Perazzi *et al.*, 2017)	80.3	75.8	18.6	72.6	70.3
RGMP (Wug Oh *et al.*, 2018)	81.5	82.0	**13.3**	-	71.1
LucidTracker (Khoreva *et al.*, 2017)	80.5	-	-	76.2	77.6
FAVOS (Cheng *et al.*, 2018)	82.4	79.5	26.3	-	-
OnAVOS (Voigtlaender and Leibe, 2017)	85.7	84.2	18.5	77.4	-
CINM (Bao *et al.*, 2018)	83.4	<u>85.0</u>	28.0	**78.4**	77.1
VideSeg-GAN w/ online adaptation (ours from chapter 3)	<u>86.2</u>	84.9	20.7	<u>77.8</u>	<u>77.7</u>
TempSeg-GAN (ours)	85.1	83.3	15.1	77.6	76.8
TempSeg-GAN++ (ours)	**86.3**	**85.2**	<u>14.2</u>	**78.4**	**77.9**

and BVS (Märki *et al.*, 2016). The quantitative results of our method in comparison with other techniques are shown in table 4.2. In terms of region similarity metric, $\mathcal{J}_{mean}$, our baseline *TempSeg-GAN w/o adapt* model beats all other existing techniques except OnAVOS (Voigtlaender and Leibe, 2017) which uses online adaptation, test time augmentation and CRF. On using CRFs and Test-time augmentation on the top of our base network (*TempSeg-GAN++*), the result obtained surpasses OnAVOS. In terms of contour accuracy, $\mathcal{F}_{mean}$, *TempSeg-GAN++* outperforms all other methods, though the base model falls short when compared with CINM (Bao *et al.*, 2018) and OnAVOS. Temporal (in-)stability measure $\mathcal{T}_{mean}$ of both *Temp-GAN* and *TempGAN++* exhibits dominant performance over all the recent and state-of-the-art semi-supervised methods except RGMP (Wug Oh *et al.*, 2018) which used guided mask propagation as a part of the model (refer column 3 of results under DAVIS-2016, in table 4.2). Thus, the overall performance of TemSeg-GAN base model along with its modified variant (TemSeg-GAN++) is better than most of the existing methods with small exceptions in a few cases. The *Inter-frame Temporal Symmetric Difference Loss (ITSDL)* used in our network can be accounted for the success of our base model which has outperformed majority of the existing state-of-the-art techniques by minimizing the number of mispredicted pixels

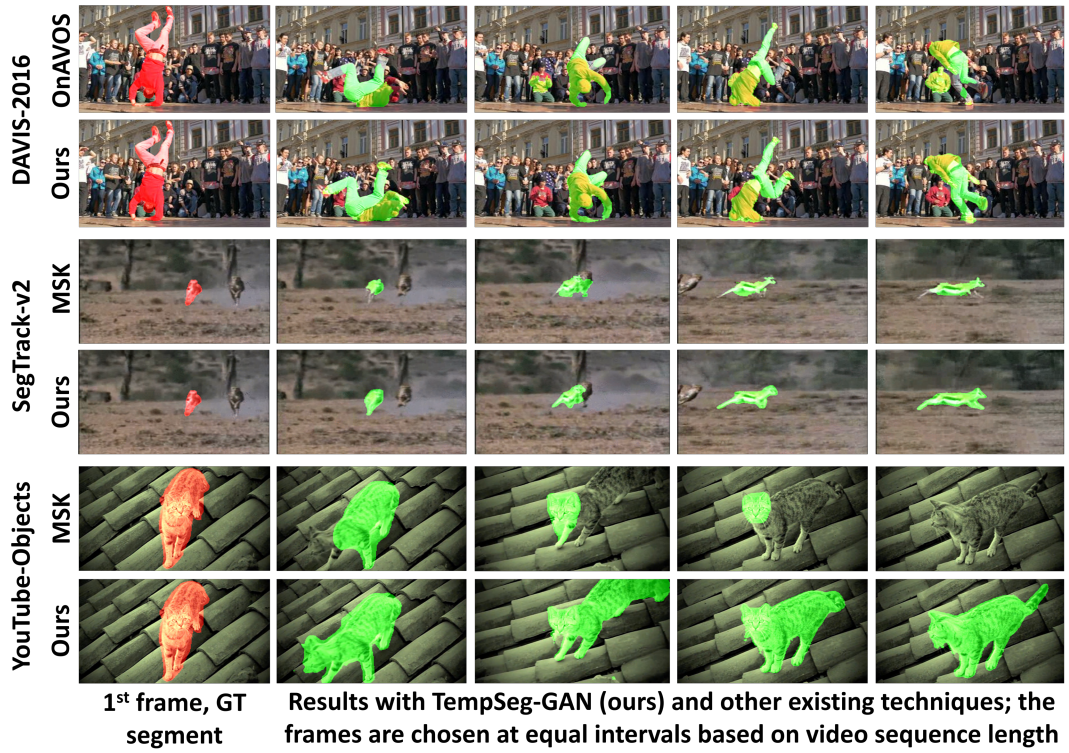


Figure 4.3: Qualitative results on three benchmark real-world datasets exhibit that our proposed method gives impressive results in challenging situations like change of appearance, occlusions, background clutter and motion blur, when compared to OnAVOS (Voigtlaender and Leibe, 2017) and MSK (Perazzi *et al.*, 2017), using one sample video for each of the 3 datasets (best viewed in color).

in segmentation. Thus it has not only increased the $\mathcal{J}_{mean}$ value, but also has improved the segmented contour by working on small patches. Again, ITSDL along with Intra Frame Temporal Loss (IFTL) and its long-range variant have contributed to the temporal stability in between the generated masks. The quantitative results in table 4.2 clarify the effectiveness of our proposed ITSDL and IFTL objective functions in improving the result of TempSeg-GAN++ over our previous method (VidSeg-GAN) in chapter 3.

Experiments are also done on SegTrack-v2 (Li *et al.*, 2013) and YouTube-Objects (Prest *et al.*, 2012) datasets and compared our results (refer table 4.2) with existing and state-of-the-art methods, for the purpose of complete evaluation. Due to insufficient training data in YouTube-Objects dataset, the same parameters as of DAVIS-2016 have been used and the pre-training step on DAVIS-2016 training set has been omitted to evaluate the generalization capability of our method. While evaluating on this dataset, we have been consistent with (Khoreva *et al.*, 2017) i.e. the frames in which the object of interest are absent has also been included. Both TempSeg-GAN base network (77.6

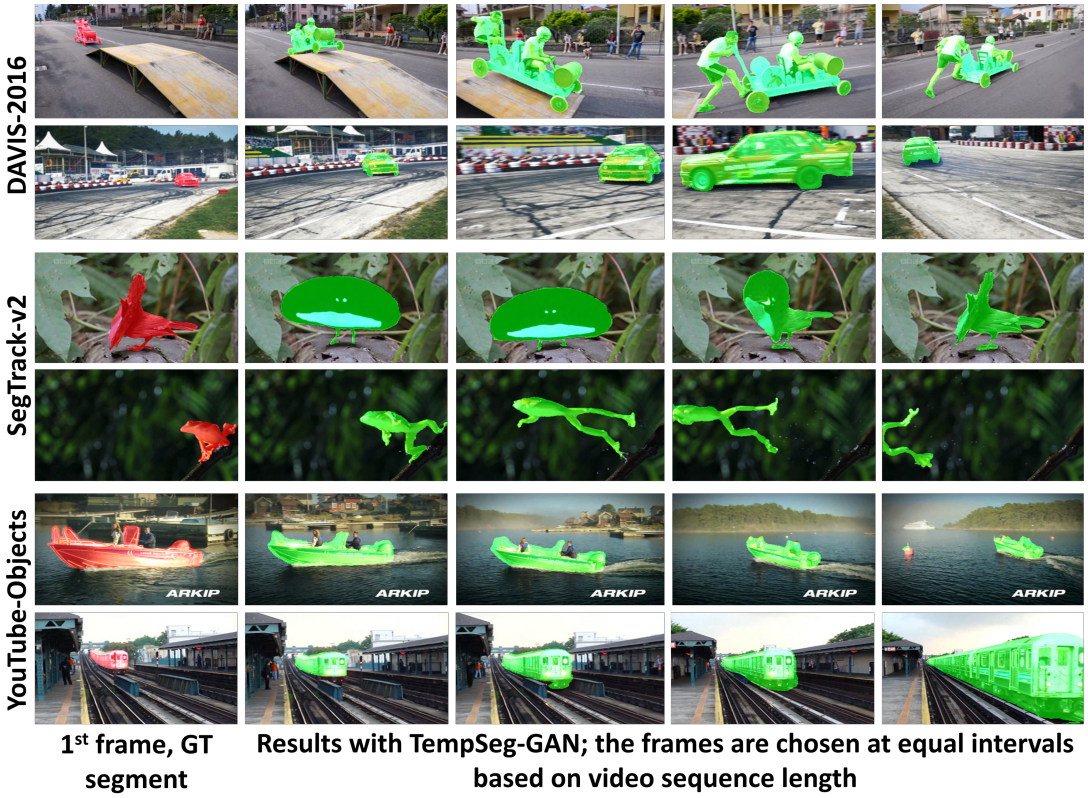


Figure 4.4: More qualitative results on three benchmark real-world datasets exhibit that our proposed method gives impressive results in challenging situations like change of appearance, occlusions, camera view change, background clutter and motion blur, using two sample videos for each of the 3 datasets (best viewed in color).

$\mathcal{J}_{mean}$) and its modified variant (78.4 $\mathcal{J}_{mean}$) give better results than OnAVOS (refer table 4.2). TempSeg-GAN++ performs at par with the state-of-the-art CINM model.

Similar choice of setting as of (Perazzi *et al.*, 2017; Khoreva *et al.*, 2017) is relied upon during the evaluation on SegTrack-v2 dataset. Fine-tuning on the initial frame of the video sequences is performed, following the same protocols of DAVIS-2016 evaluation. From table 4.2, it is evident that the modified version of our network (*TempSeg-GAN++*) performs better in comparison with the existing and state-of-the-art models. Qualitative comparison our proposed method with OnAVOS (Voigtlaender and Leibe, 2017) and MaskTrack (Perazzi *et al.*, 2017) on three benchmark real-world datasets is shown in figure 4.3, where our framework performs substantially well in segmenting object of interest under challenging conditions like occlusions, viewpoint change, background clutter, motion blur and shape deformation of object.

TempSeg-GAN subdues the drawbacks of OnAVOS (Voigtlaender and Leibe, 2017) by retaining the contextual information using the encoder-decoder arrangement of the

generator sub-network of GAN, while the patch-wise symmetric difference objective function enhances the minor details of the predicted results. Thus, it overcomes not only background clutter but also issues related with change of appearance of object also (refer figure 4.3, rows 1 and 2). For SegTrack-v2 dataset (see figure 4.3, rows 3 and 4), the segmentation quality of MSK (Perazzi *et al.*, 2017) decreases for the latter part of the video with the introduction of random motion. TempSeg-GAN manages to generate better segmentation output by maintaining the temporal consistency among the frames incorporating the temporal relations in the objective functions (Inter-frame Temporal Symmetric Difference Loss and Intra Frame Temporal Loss), rather than learning only in the spatial domain. Similar phenomenon is observed in YouTube-Objects dataset (refer figure 4.3, last 2 rows), where our method shows superiority over MSK (Perazzi *et al.*, 2017) in segmenting the object undergoing change of pose and position quite rapidly. All such cases fuel the improvement of the quantitative results (refer table 4.2) of our method over the existing algorithms. More qualitative results of TempSeg-GAN on the aforementioned datasets are shown in figure 4.4. Qualitative results in the form of video-clips for the proposed TempSeg-GAN model are available in (TempSeg-GAN_Res).

4.6 Summary

This chapter proposes a novel temporally aided Generative Adversarial Network for the task of Video Object Segmentation. The generator of the model is modified by implementing an encoder-decoder type architecture with skip connections, along with a variation in the discriminator training by introducing an additional encoder module. Introduction of Inter-frame Temporal Symmetric Difference Loss (ITSDL) and Intra Frame Temporal Loss (IFTL) not only provides a significant improvement in the segmentation results over the existing state-of-the-art techniques, but also preserves the motion features among the generated masks. Quantitative results on three benchmark datasets reveal the superiority of TempSeg-GAN over other existing and state-of-the-art methods. Though our proposed model exhibits satisfactory results, there is still room for improvement as it relies on purpose specific objective functions and fails in complex situations of multiple objects of different scales moving with varying velocities.

CHAPTER 5

Motion-based and Occlusion-aware Pixel Graph Convolutional Network for Video Object Segmentation

In this chapter, a variant of Graph Convolutional Networks (GCN) (Kipf and Welling, 2016) combined with a couple of similarity-based aggregator functions has been introduced to model the spatio-temporal relationships in a video for object segmentation. Recent methods show suboptimal performance in segmenting videos involving multiple objects moving in various directions with different speeds. In the previous chapters, we introduced novel objective functions to model the motion patterns of a single object or connected objects moving in the same direction, in the videos. However, such explicit objective functions fail to work optimally in situations involving multiple objects moving with different motion patterns and velocities. Also, real-world scenarios involve scene change in videos, occlusion as well as objects reappearing in the frames after prolonged disappearance. These challenges require independent modelling of each of the objects, which is not possible only through the use of specific tailored objective functions.

Recently introduced Graph Convolutional Networks are perfectly suitable for overcoming these challenges in a far more elegant way. GCNs perform complicated graph computation tasks by decomposing the data over a series of localized computations. The proposed direction oriented motion-based aggregator enables the network to give targeted importance to the selected neighbouring pixels along multiple directions based on their similarity in an intermediate feature space produced by a convolutional neural network. Whereas, the bi-directional computation of the forward and backward flow of motion features through the occlusion aware aggregation mechanism aids the model to effectively handle the situations of objects undergoing occlusion as well as disappearing and resurfacing after some frames. The proposed model is trained end-to-end in adversarial setting (Goodfellow *et al.*, 2014) and performs graph-based interference in both training and evaluation phases. The segmentation outcomes of the model show substantial improvement over the recent and state-of-the-art methods when evaluated on real-word benchmark video object segmentation datasets: DAVIS-2016 (Perazzi

et al., 2016) and DAVIS-2017 (Pont-Tuset *et al.*, 2017*b*). We also performed controlled experiments on semantic segmentation of traffic scenes in videos using CamVid Dataset (Brostow *et al.*, 2009) for complete evaluation and to present the ability of our proposed model in varied domains.

5.1 Overview and Formulation of Pixel-GCN

The proposed video object segmentation framework is exhibited in figure 5.1. The pixel graph convolutional network is trained adversarially in a generator-discriminator setting because of the success of the GANs (Goodfellow *et al.*, 2014) as a generative modelling architecture. Given a sequence of M consecutive RGB frames as input, the generator $(\mathbb{G})$ produces segmentation masks of objects of interest. The dimension of the generated segmentation masks being $W \times H \times 1$, while that of the input RGB frames are $W \times H \times 3$. However, transposed convolutional layers can also be added to the proposed network to produce segmentation masks of higher resolution. The generator consists of 2 channels to produce RGB and optical flow based features respectively. The RGB channel is composed of an instance based segmentation module, Mask R-CNN (He *et al.*, 2017) with ResNet-101 (He *et al.*, 2016) backbone. RGB image features from $C4$ module (last convolutional layer of the 4^{th} stage) of ResNet-101 (He *et al.*, 2016) are extracted to produce an intermediate feature map of dimension $W \times H \times D$, where D is the number of features. Likewise, optical flow based feature maps of the same dimensions are generated from the other channel formed using FlowNet 2.0 (Ilg *et al.*, 2017). To facilitate the graph formation, feature maps of M consecutive frames are combined together to elevate the features from $W \times H \times D$ to $M \times W \times H \times D$ dimension.

The RGB image and optical flow based 3D feature maps are used to construct the image feature based pixel graph ($\mathcal{G}_{rgb}$) and optical flow feature based pixel graph ($\mathcal{G}_{opt}$) respectively. The input as well as the output of each of the individual graphs are the same, and they are concatenated to form the combined graph-based feature map. These synthesized spatio-temporal features are then passed through the rest of the modules of Mask R-CNN (He *et al.*, 2017) to produce the segmentation masks of the objects. Since the end-to-end framework is trained adversarially, the discriminator $(\mathbb{D})$ differentiates

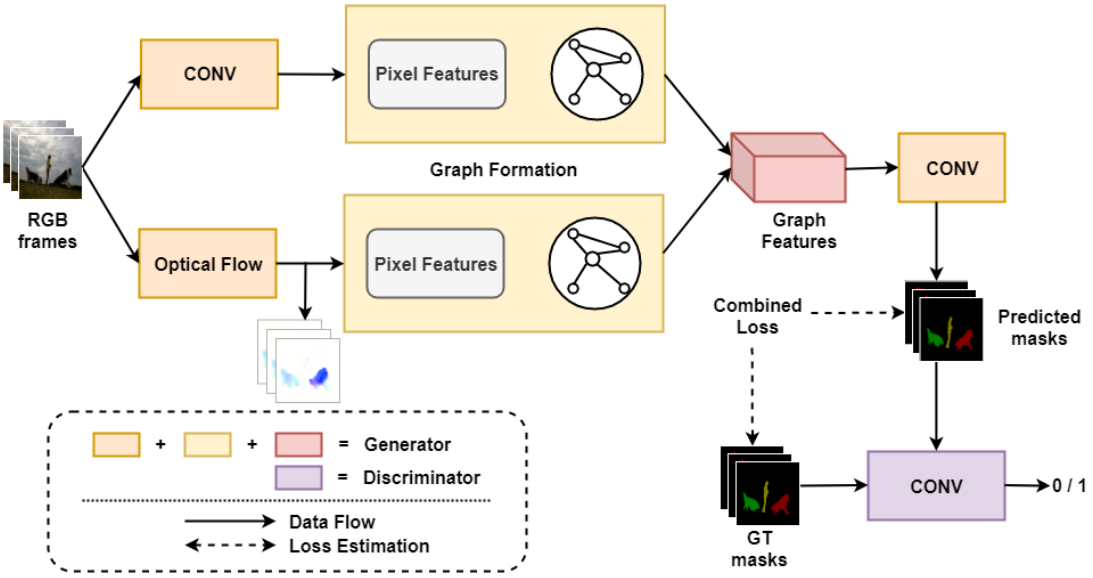


Figure 5.1: Proposed Pixel-GCN framework for Video Object Segmentation.

the generated masks from the ground-truth.

5.1.1 Graph Formulation

Given a video clip, the pixel graphs $\mathcal{G}_{rgb}$ & $\mathcal{G}_{opt}$ are formed using the internal $3D$ feature maps extracted from the ResNet-101 (He *et al.*, 2016) backbone model of the segmentation module and optical flow vector generation network respectively (see figure 5.1). Throughout the chapter, the two graphs ($\mathcal{G}_{rgb}$ & $\mathcal{G}_{opt}$) have been generalized as $\mathcal{G}_{pix}$ for ease of discussion, unless mentioned otherwise.

Using $M \times W \times H \times D$ dimensional intermediate feature map, an undirected graph $\mathcal{G}_{pix}(V, E)$ is formed, where V/E is the set of nodes/edges. As the motivation behind the graph-based framework is to model the content as well as motion pattern by learning the inter-pixel relationships in the scene across space-time, $1 \times D$ feature vectors are extracted for the node corresponding to a particular spatio-temporal pixel location of the 3D feature map.

We assume slow and smooth motion patterns across time, i.e., the videos do not involve sudden viewpoint changes or significant amount of jitter. However, implementation of the proposed aggregations along with the pixel graph ($\mathcal{G}_{pix}$) formation not only enables the model to recover itself from such challenges, but also aid to resolve situations like occlusion, shape deformation and re-appearance of objects in the video.

5.2 Proposed Aggregation Mechanisms

Following the pixel feature based graphs ($\mathcal{G}_{rgb}$ & $\mathcal{G}_{opt}$) formation, direction oriented motion based and occlusion aware aggregation functions are introduced.

The commonly used aggregation function of a GCN (Kipf and Welling, 2016) is:

$$v_i' = \sigma\Big(\phi(v_i) + \sum_{j \in \mathcal{N}_{v_i}} \mathbb{S}_{ij} \mathbb{W} \phi(v_j)\Big) \tag{5.1}$$

where, $\phi(v_i)$ denotes the input feature vector corresponding to node v_i, $\mathcal{N}_{v_i}$ is the set of its neighbouring nodes , while v_i' is the output vector. σ is a non-linear activation function (such as ReLU), $\mathbb{W}$ denotes the learnable transformation weights, and $\mathbb{S}_{ij}$ represents the adjacency function. The neighbouring nodes yield varying impact in modelling a particular node feature, depending on the relationships shared by different objects or even various parts of a single deformable object. Incorporation of features from the unrelated nodes during estimation often leads to sub-optimal outcomes from the model. To overcome this limitation, the Pixel-GCN is extended by two novel aggregator schemes: **(i) motion-based aggregation**, and **(ii) occlusion-aware aggregation**. The proposed mechanisms are discussed below.

5.2.1 Motion based Aggregation ($\mathcal{A}_{motion}$)

Learning motion patterns from short clips is a challenging task when multiple objects are moving with different velocities in various directions. We propose a motion based direction oriented aggregation to aid the segmentation network for modelling the intricate motion patterns. To estimate this, the Pixel-GCN is modified to use directional weights $\gamma_i \in [0, 1]$ for assigning relative significance on each of the neighbouring nodes. Based on the respective location from the reference node in the feature space, the neighbouring nodes are divided into 4 spatial quadrants (NW, NE, SE & SW). Direction oriented motion based categorization is incorporated in graph modelling by a set of 4 adaptive weights $\Gamma^i = \{\gamma_1^i, \gamma_2^i, \gamma_3^i, \gamma_4^i\}$ for each node v_i. The $\mathbb{S}_{ij}$ values in the standard GCN algorithm (see equation 5.1) is replaced with these learned weights, to obtain a modified

aggregation function:

$$v_i' = \sigma\Big(\phi(v_i) + \sum_{k=1}^{4} \gamma_k^i \sum_{j \in \mathcal{N}_{v_i}^k} e^{\phi(v_i)\cdot\phi(v_j)^T} \mathbb{W}_1 \phi(v_j)\Big) \tag{5.2}$$

where, $\mathcal{N}_{v_i}^k$ denotes the set of neighbouring nodes corresponding to the reference node v_i in one of the four direction based quadrants and $e^{\phi(v_i)\cdot\phi(v_j)^T}$ computes the similarity between pair of nodes. $\mathbb{W}_1$ is the matrix of learnable weights.

To model the adaptive weights, a convolutional network Ω has been incorporated, which is trained on the node features along with corresponding neighbouring nodes in the four spatial sections, described as

$$\Gamma^i = \{\gamma_k^i | k = [1, 4]\} = \Omega(v_i, \mathcal{N}_{v_i}^k) \tag{5.3}$$

where, the symbols bear usual meaning. Though the convolutional network for learning the adaptive weights can be constructed in various ways (Atwood and Towsley, 2016; Fout *et al.*, 2017; Schütt *et al.*, 2017), we designed a two-layer network with average pooling on nodes within each of the spatial quadrants followed by a max-pooling and ReLU activation with fully connected layer for our proposed framework, defined as:

$$\Gamma^i = \sigma\Big(\phi(v_i) + ||\ \lambda_k\big(\underset{j \in \mathcal{N}_{v_i}^k}{avg}\ (FC_\theta(\phi(v_j)))\big)\ ||_k\Big), \quad k = [1, 4] \tag{5.4}$$

where, λ_k is the parameter learned for each direction to find the corresponding soft attention weights; θ denotes the learnable parameters of FC-layer. The input node features are dimensionally reduced by the fully-connected layer inside the average pooling function, to diminish the computations involved during training of the sub-network. This module computes the similarity of the reference node with its neighbours in all the four directions and sets relative importance on all of them depending on their adaptive weights (Γ^i). Thus, the direction oriented motion based strategy not only aids to capture the relative motion pattern but also the long-range dependencies involved among objects in videos.

The local motion flow of the objects along with their directions and relative velocities is learned more precisely in the space-time domain by extending the entire network to include the temporal dimension. In addition, the inter-pixel similarity-based relationship

also models the long-range interaction between nearby objects as well as change of appearance of the objects due to shape and scale deformation across space-time. Thus, to implement this, adaptive weights for all the nodes, based on the four spatial quadrants (as proposed earlier), are introduced. Equation 5.4 of the sub-network in spatio-temporal domain is thus:

$$\Gamma^i = \sigma\Big(\phi(v_i) + ||\ \lambda_k\big(\underset{j \in \mathcal{N}_{v_i}^{k,t}}{avg}\ (FC_\theta(\phi(v_j)))\big)\ ||_{k,t}\Big), \quad k = [1, 4] \tag{5.5}$$

where, $\mathcal{N}_{v_i}^{k,t}$ is the set of neighbouring nodes corresponding to v_i at time-step t in the spatial section k and $||\cdot||_t$ denotes the concatenation across time-steps. Thus, the modified set of weights consisting of several sets of adaptive values per time-step is expressed as:

$$\Gamma^i = \{\{\gamma_{k,t}^i\}\ \forall t \in [\mathbb{T} - M + 1, \mathbb{T}]\} \tag{5.6}$$

where, $\mathbb{T}$ represents the current time-step.

5.2.2 Occlusion aware Aggregation ($\mathcal{A}_{occ}$)

Occlusion has always been a challenge to cope with while segmenting objects in videos. Thus, to deal with it, we propose an occlusion-aware aggregator scheme which aids to segment the objects partially or almost fully obstructed by another object. To incorporate less ambiguity in modelling the network, multiple frames ($n > 2$) have been used in formulating the aggregator function. For simplicity, 3 consecutive frames $\mathcal{I}_{t-1}$, $\mathcal{I}_t$ and $\mathcal{I}_{t+1}$ have been used for estimation. However, it can be extended for videos having more frames.

According to our approach, it is assumed that the unoccluded pixels in the initial frame should be similar to its corresponding pixels in the destination frame. But, this implication does not hold for the occluded pixels which results in three possible events: (i) occluded in the past ($\mathcal{I}_{t-1}$) and visible in the current frame ($\mathcal{I}_t$), (ii) occluded in future ($\mathcal{I}_{t+1}$) and visible in the current frame ($\mathcal{I}_t$), and (iii) occluded across all frames. Thus, to deal with these kinds of events, a bi-directional computation is done involving forward estimation of similarity between neighbouring nodes of a particular node ($v_i^{(t-1)}$) in the previous frame ($\mathcal{I}_{t-1}$) and the corresponding node ($v_i^{(t)}$) in the current frame ($\mathcal{I}_t$).

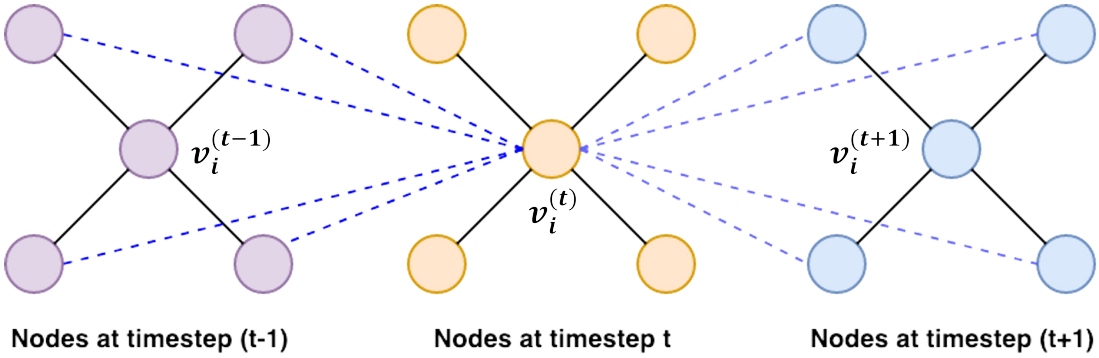


Figure 5.2: Pictorial illustration of occlusion-aware aggregation. Dotted lines represent the forward similarity estimation between the neighbouring nodes of $v_i^{(t-1)}$ and node $v_i^{(t)}$, while backward similarity estimation is done between neighbouring nodes of $v_i^{(t+1)}$ and node $v_i^{(t)}$.

Whereas, reverse (backward) estimation is done by similarity computation between the node $v_i^{(t)}$ and the neighbouring nodes corresponding to the node $v_i^{(t+1)}$ in the future frame ($\mathcal{I}_{t+1}$) (see figure 5.2). Thus, the occlusion aware aggregator function is defined as:

$$v_i'^{(t)} = \sigma\left(\phi(v_i^{(t)}) + \mathcal{U}_{occ}^t\right) \tag{5.7}$$

$$\mathcal{U}_{occ}^t = \left(\max\left(0, \delta - |\mathbb{F}(v_i^{(t)}, v_a^{(t-1)}) - \mathbb{F}(v_i^{(t)}, v_b^{(t+1)})|^2\right)\right)\mathbb{W}_2\phi(v_i) \tag{5.8}$$

where, $\mathbb{F}(v_i^{(t)}, v_a^{(t-1)}) = \sum_{a \in \mathcal{N}_{v_i}^{(t-1)}} e^{(v_i^{(t)}) \cdot (v_a^{(t-1)})^T}$ refers to the similarity estimation between node v_i at current time-step t and neighbouring nodes v_a of the node v_i at previous time-step $(t-1)$. v_b are the neighbouring nodes of v_i at next time-step $(t+1)$; δ is a positive constant and $\mathbb{W}_2$ is the shared transformation weight matrix. Thus, equations 5.7, 5.8 model the consistency between the forward and backward estimation to learn the motion pattern of the objects while penalizing the large disparity in case of occlusion with δ.

The motivation behind the two-way estimation keeping the centre frame as reference is due to the fact that, the motion is often non-linear across space-time and contain sudden variations. Our goal is to estimate the forward similarity from $\mathcal{I}_t$ to $\mathcal{I}_{t+1}$ using the occlusion aware aggregation mechanism by updating the feature vectors of the nodes. It also combines re-identification functionalities with temporal propagation which assists the network to identify missing targets re-appearing in the video after prolonged occlusion despite change in appearances.

5.3 Training of Pixel-GCN

For implementing the proposed graph-based strategy, a series of L graph aggregation layers are stacked together. Thus, the overall aggregator function used for reasoning on the graphs, formed by combining equations 5.2 and 5.7 is as follows:

$$v_i^{l+1} = \sigma\Big(\phi(v_i^l) + \mathcal{U}_{motion}^l + ||\, \mathcal{U}_{occ}^{l(t)} \,||_t\Big), \quad t = [\mathbb{T} - M + 1, \mathbb{T}] \tag{5.9}$$

where, $\mathcal{U}_{motion}^l = \sum_{k=1}^{4} \gamma_k^i \sum_{j \in \mathcal{N}_{v_i}^{k,l}} e^{\phi(v_i^l)\cdot\phi(v_j^l)^T} \mathbb{W}_1 \phi(v_j^l)$ is the motion-based aggregator update, whereas $||\, \mathcal{U}_{occ}^l \,||_t$ represents the update term of occlusion-aware aggregation function calculated over the frames, $t = [\mathbb{T} - M + 1, \mathbb{T}]$ (refer equation 5.8). $l = \{0, 1, \cdots, L\}$ are layers of the graph with $l = 0$ and $l = L$ being the input and output layers respectively. v_i^{l+1} denotes the node features at the $(l+1)^{th}$ graph layer and $v_i^0 = \phi(v_i)$. Thus, the features related to a particular node is updated based on the relations in the graph with neighbouring nodes, and this process of reasoning is propagated in a message passing manner throughout the entire graph. The similarity-based graph relations involve learnable parameters which are updated through back-propagation. Finally, the consolidated graph feature is obtained by combining the output features of both the graphs ($\mathcal{G}_{rgb}$ & $\mathcal{G}_{opt}$) from the two channels, as:

$$V_{combined}^L = V_{rgb}^L \oplus V_{opt}^L \tag{5.10}$$

where, V_{rgb}^L and V_{opt}^L refer to the features related to the nodes in the final layer L of the respective graphs and $\oplus$ is the concatenation operator.

To cope up with the large number of nodes and edges in the pixel based graphs ($\mathcal{G}_{rgb}$ & $\mathcal{G}_{opt}$), a random sampling based strategy has been adopted. It samples the neighbourhood nodes at each step in such a way that part of the neighbourhood nodes sampled and the neighbouring nodes of those mini-batch of sampled nodes are only considered in the subsequent forward passes through the graph nodes. Thus, sampling decreases the computational cost of the network significantly.

The combined feature map $V_{combined}^L$ is then passed a series of convolutional and fully-connected modules (refer figure 3.2) to generate the segmentation maps. The entire framework is trained in an adversarial setting (Goodfellow *et al.*, 2014) by minimizing

the overall objective function through back-propagation. The overall loss function is formed using a combination of $\mathcal{L}_{L_1}$ loss on the segmentation output, the Gradient Divergence Loss ($\mathcal{L}_{gdl}$) (Mathieu *et al.*, 2016) and the standard pixel-wise categorical cross-entropy loss ($\mathcal{L}_{cce}$) components, apart from the general adversarial objective ($\mathcal{L}_{adv}$). The final combined objective is thus:

$$\mathcal{L}_{combined} = \mathcal{L}_{adv} + \mathcal{L}_{L_1}(\hat{Y}, Y) + \mathcal{L}_{gdl}(\hat{Y}, Y) + \mathcal{L}_{cce}(\hat{Y}, Y) \tag{5.11}$$

where, $\hat{Y}, Y$ are the generated and ground-truth segmentation respectively. $\mathcal{L}_{gdl}$ (Mathieu *et al.*, 2016) is used to penalize the model for producing blurry edges.

Assume that there are $\mathbb{N}$ nodes in $\mathcal{G}_{pix}$; each having average degree d. Thus, for a GCN with L layers, features from $O(d^L)$ nodes are aggregated to update a single node, whereas computation of each embedding associated with a node takes $O(D^2)$ time due to multiplication with the weight matrix, $\mathbb{W}$; D is the number of features. Therefore, the average time complexity for each epoch is $O(\mathbb{N}d^L D^2)$.

5.4 Experimental Results and Discussions

Experiments are performed on two real-word Video Object Segmentation (VOS) datasets: (a) DAVIS-2016 (Perazzi *et al.*, 2016) and (b) DAVIS-2017 (Pont-Tuset *et al.*, 2017*b*), to compare the performance of proposed Pixel-GCN with the recent and state-of-the-art methods. Controlled experiments are performed on CamVid (Brostow *et al.*, 2009) dataset to exhibit the ability of the proposed model in Video Semantic Segmentation (VSS).

Implementation details. For the task of VOS, the Mask R-CNN (He *et al.*, 2017) network is adopted to the DAVIS (Perazzi *et al.*, 2016; Pont-Tuset *et al.*, 2017*b*) datasets by using the pre-trained ImageNet (Deng *et al.*, 2009) weights and then training with COCO (Lin *et al.*, 2014) dataset, followed by fine-tuning with the training set of DAVIS datasets. Next, the intermediate features of 4 consecutive RGB images, from ResNet-101 backbone of Mask R-CNN model, are used as input to the proposed Pixel-GCN. Updated features from the graph-based model, are then passed through the rest of the layers of Mask R-CNN to produce the segmented output. Before validation, Mask R-CNN model

is fine-tuned on each video sequence with synthetic in-domain images, generated from the first frame of the respective videos using Lucid Dreaming (Khoreva *et al.*, 2017). It is done to provide the contextual information regarding the objects of interest to the model. Finally, consecutive RGB frames are fed to the proposed network to produce segmentation masks of objects present in these frames.

DeeplabV3+ (Chen *et al.*, 2018*a*) network with Xception65 (Chollet, 2017) backbone is used for semantic segmentation of videos. As the number of segmented ground-truth images are quite less in CamVid (Brostow *et al.*, 2009) dataset, Cityscape (Cordts *et al.*, 2016) pre-trained weights has been deployed followed by fine-tuning on CamVid. The internal features of 4 RGB input images with equal time intervals, obtained from encoder output of DeeplabV3+, having atrous rates of 6, 12 and 18 with output stride of 16 (just before the final 1×1 convolution), is used as input to the graph network while the updated output features are passed through the decoder part of DeeplabV3+ to generate the semantic segmentation masks (for architecture details of Deeplab V3+ refer (Chen *et al.*, 2018*a*)). During testing, consecutive RGB images are sent to the framework for semantic segmentation of the same. All the experiments have been performed on 2 NVIDIA 1080 GPUs.

Datasets. Recently proposed **DAVIS-2016** (Perazzi *et al.*, 2016) dataset used for evaluation of VOS methods, consists of 50 high-resolution video sequences with a total of 3455 frames. Among them, 30 is used for training and remaining 20 for validation purposes. Each of the video sequences contains single or multiple connected objects per frame, provided with pixel-level segmentation. **DAVIS-2017** (Pont-Tuset *et al.*, 2017*b*), the extended version of DAVIS-2016 dataset, consists of multiple objects annotated corresponding to different instances. It consists of 60 video sequences for training and 30 videos reserved for validation. Background clutter, occlusion, shape deformation, re-appearance and dis-appearance of objects from frames are the main challenges in DAVIS datasets. On the other hand, **CamVid** (Brostow *et al.*, 2009) is a small road scene dataset consisting of 600 ground-truth segmented RGB images from various videos of resolution 360×480. The dataset contains scenes of day and dusk with 367 images as training data and 233 images for testing.

Evaluation Metrics. Following protocols of (Perazzi *et al.*, 2016), two evaluation metrics have been used in performance assessment of VOS: (a) **Region similarity** ($\mathcal{J}$),

Table 5.1: Quantitative comparison of the proposed Pixel-GCN with existing state-of-the-art methods on DAVIS-2016 and DAVIS-2017 validation sets for VOS. " $\uparrow$ " - higher is better. " $*$ " - w/o proposed aggregation functions ($\mathcal{A}_{motion}$ & $\mathcal{A}_{occ}$) in rows 9 & 10 (Best results in **bold**).

Methods	DAVIS-2017			DAVIS-2016		
	$\mathcal{G}_{mean}\uparrow$	$\mathcal{J}_{mean}\uparrow$	$\mathcal{F}_{mean}\uparrow$	$\mathcal{G}_{mean}\uparrow$	$\mathcal{J}_{mean}\uparrow$	$\mathcal{F}_{mean}\uparrow$
OSMN (Yang *et al.*, 2018)	54.8	52.5	57.1	73.5	74.0	72.9
FAVOS (Cheng *et al.*, 2018)	58.2	54.6	61.8	81.0	82.4	79.5
OSVOS (Caelles *et al.*, 2017)	60.3	56.6	63.9	80.2	79.8	80.6
OnAVOS (Voigtlaender and Leibe, 2017)	65.4	61.6	69.1	86.6	86.1	84.9
RGMP (Wug Oh *et al.*, 2018)	66.7	64.8	68.6	81.8	81.5	82.0
OSVOS-S (Maninis *et al.*, 2018)	68.0	64.7	71.3	86.6	85.6	87.5
CINM (Bao *et al.*, 2018)	70.6	67.2	74.0	84.2	83.4	85.0
PReMVOS (Luiten *et al.*, 2018)	77.8	73.9	81.7	86.8	84.9	**88.6**
VideSeg-GAN w/ online adaptation (ours)	-	-	-	85.6	86.2	84.9
TempSeg-GAN++ (ours)	-	-	-	85.8	86.3	85.2
Vanilla GCN* ($\mathcal{G}_{rgb}$, ours)	77.0	73.6	80.5	-	-	-
Pixel-GCN* ($\mathcal{G}_{rgb} + \mathcal{G}_{opt}$, ours)	77.6	74.1	81.2	-	-	-
Pixel-GCN-FS (ours)	78.4	74.9	81.8	-	-	-
Pixel-GCN-FF (ours)	**78.9**	**75.6**	**82.2**	**87.3**	**86.5**	88.1

and (b) **Contour accuracy** ($\mathcal{F}$). $\mathcal{J}$ or *intersection-over-union* (IoU) computes the region overlap between the predicted and ground-truth masks to measure the similarity in segmentation. In table 5.1, global metric $\mathcal{G}$ is the mean of $\mathcal{J}$ and $\mathcal{F}$. Whereas, $\mathcal{F}$ utilizes the contour points of the segmentation boundary to give an evaluation based on their precision and recall. Performance of video semantic segmentation has been evaluated using mIoU metric only.

Performance Analysis in VOS. Quantitative comparison on DAVIS-2017 validation set given in table 5.1 shows that our proposed Pixel-GCN outperforms each of the existing and the state-of-the-art methods in terms of $\mathcal{G}_{mean}$, $\mathcal{J}_{mean}$ and $\mathcal{F}_{mean}$, without using any post-processing module like CRF (Krähenbühl and Koltun, 2011). Experiments on DAVIS-2016 also proves the superior performance of our model in terms of $\mathcal{J}_{mean}$, but it only falls short in terms of $\mathcal{F}_{mean}$ to PReMVOS (Luiten *et al.*, 2018) where ours is a close second. Previous works exhibit suboptimal performance in situations like rapid motion, occlusion and shape deformation. In our proposed graph network, the directional motion-based aggregation scheme ($\mathcal{A}_{motion}$) aids the model to generate impressive results over PReMVOS (Luiten *et al.*, 2018) (see results on DAVIS-2017 in figure 5.3). Pixel-GCN not only succeeds in segmenting multiple objects with dissimilar motion patterns, but

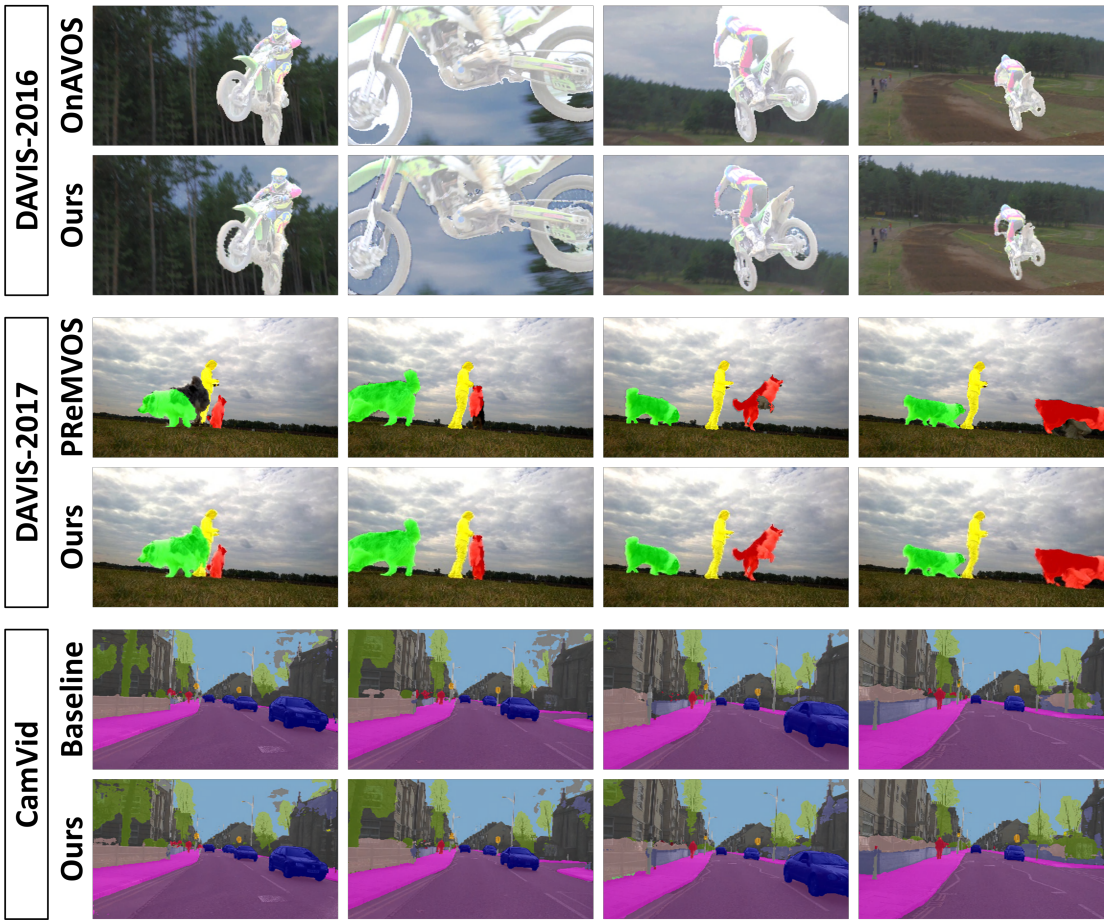


Figure 5.3: Qualitative comparison of the performance of our Pixel-GCN framework with the existing state-of-the-art methods on DAVIS-2016, DAVIS-2017 and CamVid datasets. The frames are taken at equal intervals of time (Best viewed in colour).

also remains robust to the appearance change of the objects, while PReMVOS (Luiten *et al.*, 2018) fails in segmenting objects in a few cases. Incorporation of the novel occlusion-aware aggregation function ($\mathcal{A}_{occ}$) has also improved the performance of the proposed model in situations like partial occlusion of objects of interest and also identification of objects or parts re-appearing in the video (refer DAVIS-2016 in figure 5.3).

Apart from these, we have performed ablation study on DAVIS-2017 dataset to investigate the individual contribution of each module in our proposed Pixel-GCN (see last 4 rows in table 5.1). Using only RGB feature based graph ($\mathcal{G}_{rgb}$) in the vanilla GCN (Kipf and Welling, 2016) we attain a $\mathcal{J}_{mean}$ of 73.6. Introduction of optical flow feature based graph ($\mathcal{G}_{opt}$) provides the model with more spatio-temporal information to capture the relative motion patterns among objects. We have also experimented with two different models where the first one learns to produce the segmented masks

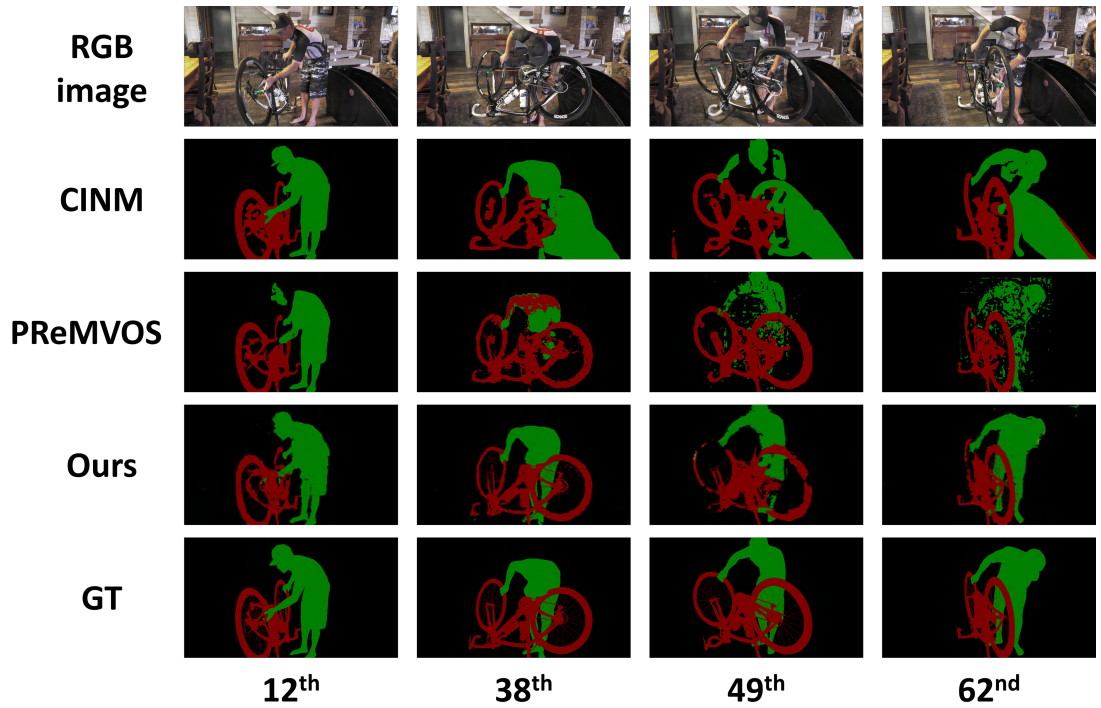


Figure 5.4: Comparison of qualitative results of our Pixel-GCN framework with the existing methods on DAVIS-2017 dataset. GT refers to the Ground-Truth. The numbers at the bottom denote time-step of the frames in the video (Best viewed in colour).

directly, while the other relies on the updated features to be passed through the Mask R-CNN (He *et al.*, 2017) to generate the predicted masks. The former model consists of a series of convolutional layers and a fully-connected layer attached to the graph network along with class-wise cross-entropy loss with adversarial objective for inference purposes. From table 5.1 (last 2 rows), it is evident that better results are obtained by processing the updated features than producing the segmentation masks directly. These two models also include the novel aggregation functions ($\mathcal{A}_{motion}$ & $\mathcal{A}_{occ}$) which improve the results of the frameworks by making it robust to complex motions, shape deformation and occlusion in comparison with the vanilla GCN and Pixel-GCN bare-model (without $\mathcal{A}_{motion}$ & $\mathcal{A}_{occ}$) mentioned earlier (see rows 11 & 12 of table 5.1). More comparative visual results of Pixel-GCN with the existing and state-of-the-art methods on DAVIS-2017 dataset for multi-object scenario are shown in figures 5.4 and 5.5.

From figure 5.3 (rows 1 and 2), it is evident that OnAVOS (Voigtlaender and Leibe, 2017) fails miserably when the object undergoes partial occlusion and change of appearance. These issues are subdued by the rich intermediate spatio-temporal feature space of our Pixel-Graph network, which captures the contextual information as well as motion

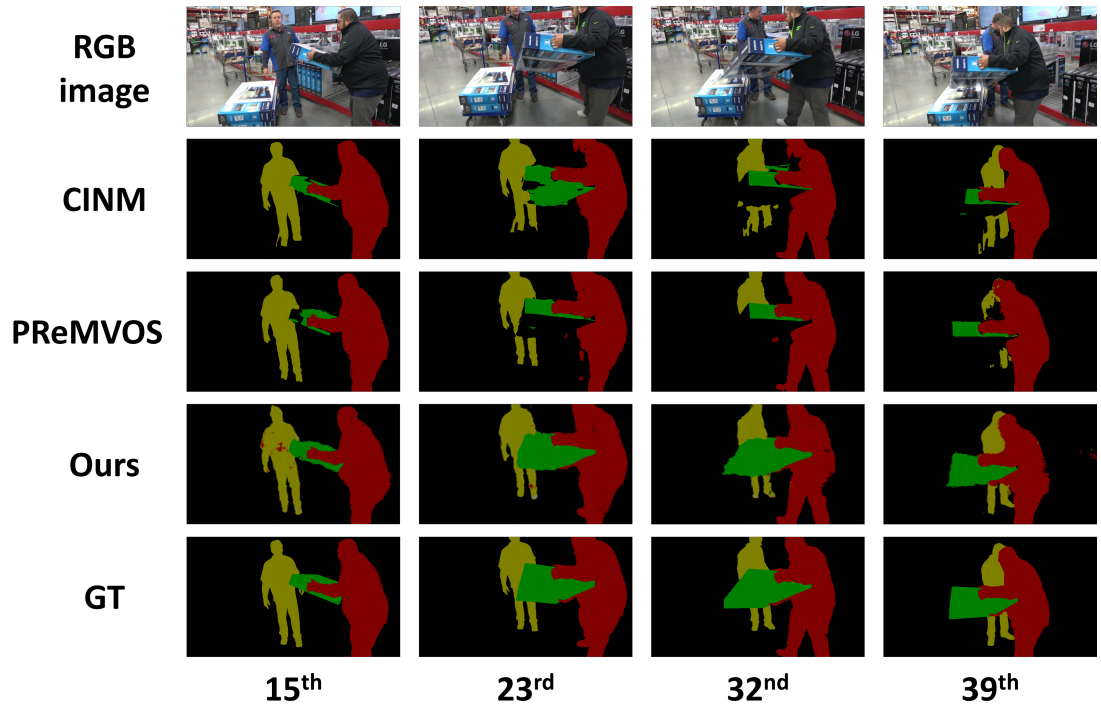


Figure 5.5: Comparison of qualitative performance of our Pixel-GCN framework with the existing methods on DAVIS-2017 dataset. GT refers to the Ground-Truth. The numbers at the bottom denote time-step of the frames in the video (Best viewed in colour).

with the object effectively. The occlusion aware aggregator mechanism ($\mathcal{A}_{occ}$) also plays a vital role in segmenting the object (in this example: motor-cycle) suffering from partial occlusion in the intermediate video frames. In DAVIS-2017 dataset, Pixel-GCN exhibits superior performance over its closest competitor PReMVOS (Luiten *et al.*, 2018) in terms of both region similarity ($\mathcal{J}$) and contour accuracy ($\mathcal{F}$). It is noted that the drop in segmentation quality is substantially less than other methods, as we move towards the latter frames in a video sequence involving multiple objects (refer figures 5.3 (rows 3 and 4), 5.4 and 5.5). This can be attributed to the introduction of a weighted direction oriented motion based aggregator mechanism ($\mathcal{A}_{motion}$), which not only captures the variations in motion patterns involved with multiple objects moving in different directions with dissimilar speeds, but also efficiently handles the dynamic (temporal) changes in appearance of the same in terms of pose and scale across the frames. PReMVOS (Luiten *et al.*, 2018) solely relies on segmenting and merging of the object proposals obtained from the object detection algorithm used, which only captures the spatial information. Thus, though the complex framework of PReMVOS learns to segment multiple objects, it often fails to perform in case of complex scenes involving objects with highly non-smooth, non-linear and random-type motion. On the other hand, as our method (Pixel-GCN) learns an

Table 5.2: Quantitative comparison of the proposed Pixel-GCN with methods developed in 2019 on DAVIS-2016 and DAVIS-2017 validation sets for VOS. " ↑ " - higher is better (Best results in **bold**).

Methods	DAVIS-2017			DAVIS-2016		
	$\mathcal{G}_{mean}\uparrow$	$\mathcal{J}_{mean}\uparrow$	$\mathcal{F}_{mean}\uparrow$	$\mathcal{G}_{mean}\uparrow$	$\mathcal{J}_{mean}\uparrow$	$\mathcal{F}_{mean}\uparrow$
VideSeg-GAN w/ online adaptation (ours)	-	-	-	85.6	86.2	84.9
TempSeg-GAN++ (ours)	-	-	-	85.8	86.3	85.2
Pixel-GCN-FF (ours)	78.9	75.6	82.2	87.3	86.5	88.1
STMN (Oh *et al.*, 2019*b*)	**81.8**	**79.2**	**84.3**	**89.3**	**88.7**	**89.9**
FEELVOS (Voigtlaender *et al.*, 2019)	71.6	69.1	74.0	81.6	81.1	82.2
MHP-VOS (Xu *et al.*, 2019)	75.3	71.8	78.8	86.9	85.7	88.1

intermediate spatio-temporal Graph feature representation that is explicitly trained to fuel the segmentation quality, the output masks look much more superior than that of the existing methods. Similarly, for Video Semantic Segmentation (VSS) on CamVid dataset (refer figure 5.3, last 2 rows), our method demonstrates comparatively better results than the baseline. This confirms that the improvement stems from the introduction of the graph layers along with the aggregation functions. Thus, the quantitative (refer tables 5.1 and 5.3) and qualitative results (refer figures 5.3 - 5.5) prove that our proposed framework has generalizing stability for different motion-based tasks across both VOS and VSS domains.

Space-Time Memory Networks (STMN) (Oh *et al.*, 2019*b*) being trained on YouTube-VOS (Xu *et al.*, 2018) dataset exhibits superior performance over our Pixel-GCN on both DAVIS-2016 ($\mathcal{J}_{mean}$: 88.7, $\mathcal{F}_{mean}$: 89.9) and DAVIS-2017 ($\mathcal{J}_{mean}$: 79.2, $\mathcal{F}_{mean}$: 84.3) validation sets. But it has not been included in table 5.1 (instead has been mentioned in table 5.2) to maintain fair comparison since it is trained on a completely different dataset and also is published after our Pixel-GCN submitted for publication. Similarly, though Pixel-GCN outperforms FEELVOS (Voigtlaender *et al.*, 2019) (DAVIS-2016: $\mathcal{J}_{mean}$: 81.1, $\mathcal{F}_{mean}$: 82.2; DAVIS-2017: $\mathcal{J}_{mean}$: 69.1, $\mathcal{F}_{mean}$: 74.0) and MHP-VOS (Xu *et al.*, 2019) (DAVIS-2016: $\mathcal{J}_{mean}$: 85.7, $\mathcal{F}_{mean}$: 88.1; DAVIS-2017: $\mathcal{J}_{mean}$: 71.8, $\mathcal{F}_{mean}$: 78.8) on DAVIS-2016 and 2017 validation sets, still these methods have not been included in table 5.1 (instead have been mentioned in table 5.2), since they appear chronologically later than our publications and completion of work. Among them, FEELVOS (Voigtlaender *et al.*, 2019) is also trained on YouTube-VOS (Xu *et al.*, 2018) dataset and evaluated on DAVIS validation sets like Space-Time Memory Networks

Figure 5.6: Comparative study of qualitative performance of our three proposed networks *viz.* VidSeg-GAN, TempSeg-GAN and Pixel-GCN on DAVIS-2016 dataset. GT refers to the Ground-Truth. The frames are chosen at equal intervals of time (Best viewed in colour).

(Oh *et al.*, 2019*b*). However, MHP-VOS (Xu *et al.*, 2019) ($\mathcal{F}_{mean}$: 88.1) performs at par with Pixel-GCN in terms of $\mathcal{F}_{mean}$ on DAVIS-2016 validation set. For Pixel-GCN quantitative results refer to the 3^{rd} row of table 5.2.

Figure 5.6 and table 5.1 (rows 9, 10 & 14) exhibit the incremental improvement of qualitative and quantitative performances of our three proposed networks *viz.* VidSeg-GAN (chapter 3), TempSeg-GAN (chapter 4) and Pixel-GCN on DAVIS-2016 dataset respectively. Experiments have also been done on DAVIS-2016 to evaluate the Temporal in-stability ($\mathcal{T}_{mean}$) of Pixel-GCN ($\mathcal{T}_{mean}$: 13.4), where it shows superior performance over both VidSeg-GAN ($\mathcal{T}_{mean}$: 20.7 (table 4.2: third last row, column 3)) and TempSeg-GAN ($\mathcal{T}_{mean}$: 14.2 (table 4.2: last row, column 3)). The capability of motion-based aggregation mechanism ($\mathcal{A}_{motion}$) in Pixel-GCN to capture the long-range motion patterns of multiple objects coupled with the occlusion-aware aggregation ($\mathcal{A}_{occ}$) can be accounted for the success of Pixel-GCN in maintaining temporal stability among the sequence of video frames. Qualitative results in the form of video-clips for the proposed Pixel-GCN model are available in (Pixel-GCN_Res).

Performance Analysis in VSS. For complete evaluation, our proposed Pixel-GCN model has been studied on Camvid (Brostow *et al.*, 2009) dataset for semantic segmenta-

Table 5.3: Quantitative comparison of various models on CamVid dataset having features as input for semantic segmentation in videos. "MO" refers to the moving object categories in the videos. "FS": Feature → Segmentation, whereas "FF": Feature → Feature. " $*$ " - without proposed aggregation functions ($\mathcal{A}_{motion}$ & $\mathcal{A}_{occ}$) in rows 2 & 3 (Best results in **bold**).

Models	IoU (SEG)	IoU-MO (SEG)
Baseline (DeeplabV3+ (Chen *et al.*, 2018*a*))	62.7	60.3
Vanilla GCN* ($\mathcal{G}_{rgb}$)	64.3	61.5
Pixel-GCN* ($\mathcal{G}_{rgb} + \mathcal{G}_{opt}$)	65.2	63.6
Pixel-GCN* + $\mathcal{A}_{motion}$	67.5	64.8
Pixel-GCN-FS	67.9	65.1
Pixel-GCN-FF	**70.1**	**67.8**

tion of videos (refer CamVid in figure 5.3). Quantitative comparison of various models along with the full-frame and moving objects segmentation baseline in the videos are shown in table 5.3. DeeplabV3+ (Chen *et al.*, 2018*a*) trained in Cityscapes (Cordts *et al.*, 2016) and fine-tuned in CamVid (Brostow *et al.*, 2009) dataset is used as the baseline model keeping the common categories of both the datasets. For evaluation of the proposed models, intermediate features of Xception65 (Chollet, 2017) model trained under the same settings are used for graph inference, followed by processing of the output features using decoder network of DeeplabV3+ (Chen *et al.*, 2018*a*). From rows 2-6 of table 5.3, it is evident that introduction of the aggregation functions ($\mathcal{A}_{motion}$ & $\mathcal{A}_{occ}$) contribute much in the improvement of the performance of the model in comparison with the vanilla or Pixel-GCN bare-model ($\mathcal{G}_{rgb} + \mathcal{G}_{opt}$). Also, segmentation using updated features through DeeplabV3+ model produces better results than direct segmentation (see last 2 rows in table 5.3). The models exhibit a dip in performance in case of segmentation of moving objects (IoU-MO) in the videos in comparison with general semantic segmentation (see column 1 and 2 in table 5.3). This is not at all surprising because the moving objects often produce quite different motion patterns and velocities, thus making it significantly hard to segment objects if equal importance is given to all neighbourhood pixels over all four directions. Formulation of motion-based aggregation ($\mathcal{A}_{motion}$) mechanism aided with direction oriented adaptive weights has improved the quantitative performance of Pixel-GCN over its vanilla variant and baseline (Chen *et al.*, 2018*a*) by a significant margin. Bi-directional estimation of flow features in Occlusion-aware aggregation ($\mathcal{A}_{occ}$) assists to segment objects in scenarios involving occlusion and re-identification of objects disappearing from the frame and surfacing after some time.

Table 5.4: Runtime analysis of our three proposed Video Object Segmentation networks viz. VidSeg-GAN, TempSeg-GAN and Pixel-GCN with respect to the existing and state-of-the-art methods on DAVIS-2016 and DAVIS-2017 validation sets. The comparison results of other methods are quoted from the respective previous works and FEELVOS paper (Voigtlaender *et al.*, 2019). †: Speed for DAVIS-2017 has been extrapolated from DAVIS-2016 assuming linear scaling in the number of objects as per FEELVOS paper (Voigtlaender *et al.*, 2019). Speed is measured in frames per second (fps) (Best results in **bold**).

Methods	Speed (fps)	
	DAVIS-2017	DAVIS-2016
OnAVOS (Voigtlaender and Leibe, 2017)	0.04†	0.08
OSVOS (Caelles *et al.*, 2017)	-	0.1
MaskTrack (Perazzi *et al.*, 2017)	-	0.1
PReMVOS (Luiten *et al.*, 2018)	0.02	0.03
FAVOS (Cheng *et al.*, 2018)	0.84†	1.67
RGMP (Wug Oh *et al.*, 2018)	**3.84**†	**7.69**
OSMN (Yang *et al.*, 2018)	3.57†	7.14
VideSeg-GAN (ours)	-	0.13
TempSeg-GAN (ours)	-	0.12
Pixel-GCN (ours)	0.07	0.08

Runtime analysis in VOS. Table 5.4 compares the average speed of different existing and state-of-the-art methods with our three proposed networks *viz.* VidSeg-GAN, TemSeg-GAN and Pixel-GCN, during testing on DAVIS-2016 and DAVIS-2017 validation sets. The runtime values of the existing and proposed networks have been measured on dual NVIDIA 1080 GPUs. The existing works of Video Object Segmentation that follow semi-supervised approach are generally initialized with a mask and mostly undergo computationally expensive techniques during test time like data augmentation (Perazzi *et al.*, 2017), fine-tuning (Caelles *et al.*, 2017; Voigtlaender and Leibe, 2017; Luiten *et al.*, 2018), optical flow (Perazzi *et al.*, 2017) and post-processing using CRF (Perazzi *et al.*, 2017). Thus, these mechanisms lead to a larger computation time even for a short sequence of video. On the other hand, RGMP (Wug Oh *et al.*, 2018), OSMN (Yang *et al.*, 2018) do not involve any computationally intensive fine-tuning steps, making them faster than the fine-tuned based methods by order of 2. FAVOS (Cheng *et al.*, 2018) utilizes object proposal information for tracking instead of initial masks. The proposed GAN based networks *viz.* VidSeg-GAN and TempSeg-GAN exhibit average speed of 0.13 fps and 0.12 fps respectively (see 3rd last and 2nd last row in table 5.4) on DAVIS-2016 dataset due to the presence of fine-tuning, data augmentation and CRF post-processing based stages in both the networks. Additional optical flow based module in TemSeg-

GAN leads to lesser computation speed compared to VidSeg-GAN model. Though the proposed graph-based Pixel-GCN architecture does not posses any post-processing step like VidSeg-GAN and TempSeg-GAN, still it shows much slower average processing speed (see last row in table 5.4) than the two. This is mainly due to the computational burden resulting from the image and optical flow vector based feature extractors and the graph aggregation layers used for forming the respective graphs on the top of those extractors.

Our methods show slower processing speed in comparison with FEELVOS (Voigtlaender *et al.*, 2019) (DAVIS-2016: 2.22 fps; DAVIS-2017: 1.96 fps) and Space-Time Memory Networks (Oh *et al.*, 2019*b*) (DAVIS-2016: 6.25 fps), but have not been included in table 5.4 since these are published chronologically after ours.

5.5 Summary

In this chapter, we have introduced a novel approach of segmenting objects in videos using a dual-channel Graph Convolutional Network. The RGB feature based graph provides the contextual information, while the motion patterns are captured through the optical flow based feature graph, by learning the inter-pixel relationship in the space-time domain. In addition, a motion-based aggregator scheme has been proposed to model the non-periodic object movements along with its change in appearance in terms of pose and scale. Use of novel occlusion-aware aggregator aids the network to identify targets under occlusion or re-appearing in the frame. Our model not only shows superior performance over existing and state-of-the-art methods on real-world VOS datasets viz. DAVIS-2016 and DAVIS-2017, but also produces impressive results on CamVid dataset for video semantic segmentation.

CHAPTER 6

Conclusion

In this thesis, we have proposed three efficient adversarial training-based methods to study the problem of Video Object Segmentation and overcome the short-comings of the existing works. First, a generative adversarial network based framework (VidSeg-GAN) with Intersection-over-Union score based Patch-wise Symmetric Difference Loss (PSDL) function has been used for training the model. Independent processing of the frames through the adversarial framework aids to generate optimal results in cases of scene change and random object motion, where explicit trajectory flow intensive methods fail to give satisfactory solution. In the second work, the VidSeg-GAN model is extended incorporating temporal information by formulating two novel cost functions: (a) Inter-frame Temporal Symmetric Difference Loss (ITSDL) and Intra Frame Temporal Loss (IFTL) to stabilize the training process. The objective functions not only assist in providing enhanced foreground segmentation of objects, but also captures the motion patterns of objects to maintain temporal coherency among long-range video frames. Finally, we propose a variant of Graph Convolutional Network (Pixel-GCN) coupled with motion and occlusion aware aggregation schemes to learn the relative independent motions involved with multiple objects in videos along with the change in appearance in complex scenarios like occlusion, motion blur, background clutter, etc. Experimental evaluations on the real-world datasets *viz.* DAVIS-2016, DAVIS-2017, SegTrack-v2, YouTube Objects and CamVid, reveal that the introduction of adversarial training based frameworks, with incorporation of novel temporal information aided objective functions and the formulation of Pixel Graph Convolutional Network have subsequently improved the segmentation quality.

6.1 Contribution

The key contributions of this thesis have been highlighted as follows:

(i) Introduction of an adversarial training based model (VidSeg-GAN) along-with proposed Patch-wise Symmetric Difference Loss (PSDL) to reduce the disparity between generated and ground-truth segmentation masks of objects in videos.

(ii) Formulation of two novel objective functions to incorporate temporal information (in TempSeg-GAN), as: Inter-frame Temporal Symmetric Difference Loss (ITSDL) and Intra Frame Temporal Loss (IFTL), not only to capture the motion patterns of objects involved in video frames, but also to enhance the performance of the network in segmenting objects smaller in size.

(iii) A dual-channel variant of Graph Convolutional Network (Pixel-GCN) is implemented where image feature based graph (G_{rgb}) provides the contextual information and the optical flow feature based graph (G_{opt}) captures the motion patterns involved with objects in the video, thus modelling the inter-pixel spatio-temporal relations.

(iv) Novel motion-based and occlusion-aware aggregation mechanisms are proposed for graph reasoning of the dual-channel graph convolutional network (Pixel-GCN). The former learns the spatio-temporal relationships among multiple objects moving in various directions with different speeds and change in appearance of objects in terms of scale, shape and pose. Whereas, the latter models the objects undergoing occlusion and re-identifying objects disappearing from the frames and resurfacing after some time.

6.2 Future Scope of Work

The various concepts and modules proposed in our work can be utilized by researchers to perform a variety of tasks to solve different problems. The semantic features obtained during Video Object Segmentation (VOS) can be applied as intermediate features to provide contextual information in several other tasks which are new in the field of Computer Vision and Machine Learning like video captioning, video summarization, visual question-answering, etc. Since our proposed model of Pixel-GCN also deals with the understanding of motion patterns of multiple moving objects in scenes, the framework can be modified to solve problems like action recognition, event categorization in videos, etc. Some other works include segmenting scenes or objects under challenging conditions like videos containing sufficient amount of jitter, roll-in camera effects and cases where videos have unnecessary noise and other perturbations. Cross-domain VOS can also be an interesting problem to work on (e.g. training on urban street scenes and using the same model to test in rural, hilly terrains or in adversarial scenarios like foggy weather conditions, during rain/snow, different times of a day at various illumination conditions including night-time). Use of spatio-temporal features opens the scope of dealing with

the future events where our model can be extended to complex tasks like future frame segmentation, video prediction, etc. which have an immense impact in understanding the unseen future in case of autonomous systems. Also, better mechanisms or metrics can be formulated to reflect the overall improvement of quality of segmentation masks produced and understand the relation between the qualitative and quantitative results.

REFERENCES

1. **Agarwala, A.**, **A. Hertzmann**, **D. H. Salesin**, and **S. M. Seitz**, Keyframe-based tracking for rotoscoping and animation. *In ACM Transactions on Graphics (ToG)*. 2004. 11

2. **Atwood, J.** and **D. Towsley**, Diffusion-convolutional neural networks. *In Advances in Neural Information Processing Systems (NIPS)*. 2016. 53

3. **Bai, X.**, **J. Wang**, **D. Simons**, and **G. Sapiro**, Video snapcut: robust video object cutout using localized classifiers. *In ACM Transactions on Graphics (ToG)*. 2009. 11

4. **Bao, L.**, **B. Wu**, and **W. Liu**, Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. 10, 44, 45, 59

5. **Benard, A.** and **M. Gygli**, Interactive video object segmentation in the wild. *In arXiv preprint arXiv:1801.00269*. 2017. 12

6. **Brostow, G. J.**, **J. Fauqueur**, and **R. Cipolla**, Semantic object classes in video: A high-definition ground truth database. *In Pattern Recognition Letters*. 2009. 13, 50, 57, 58, 64, 65

7. **Caelles, S.**, **K.-K. Maninis**, **J. Pont-Tuset**, **L. Leal-Taixé**, **D. Cremers**, and **L. Van Gool**, One-shot video object segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. vii, 1, 2, 9, 12, 15, 28, 29, 30, 31, 44, 45, 59, 66

8. **Chen, L.-C.**, **G. Papandreou**, **I. Kokkinos**, **K. Murphy**, and **A. L. Yuille**, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 2017. 9

9. **Chen, L.-C.**, **Y. Zhu**, **G. Papandreou**, **F. Schroff**, and **H. Adam**, Encoder-decoder with atrous separable convolution for semantic image segmentation. *In European Conference on Computer Vision (ECCV)*. 2018*a*. 10, 58, 65

10. **Chen, Y.**, **J. Pont-Tuset**, **A. Montes**, and **L. Van Gool**, Blazingly fast video object segmentation with pixel-wise metric learning. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018*b*. 9

11. **Cheng, J.**, **Y.-H. Tsai**, **W.-C. Hung**, **S. Wang**, and **M.-H. Yang**, Fast and accurate online video object segmentation via tracking parts. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. 9, 44, 45, 59, 66

12. **Chollet, F.**, Xception: Deep learning with depthwise separable convolutions. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 58, 65

13. **Cordts, M.**, **M. Omran**, **S. Ramos**, **T. Rehfeld**, **M. Enzweiler**, **R. Benenson**, **U. Franke**, **S. Roth**, and **B. Schiele**, The cityscapes dataset for semantic urban scene understanding. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. 58, 65

14. **Deng, J.**, **W. Dong**, **R. Socher**, **L.-J. Li**, **K. Li**, and **L. Fei-Fei**, Imagenet: A large-scale hierarchical image database. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2009. 2, 20, 24, 26, 27, 42, 43, 57

15. **Fout, A.**, **J. Byrd**, **B. Shariat**, and **A. Ben-Hur**, Protein interface prediction using graph convolutional networks. *In Advances in Neural Information Processing Systems (NIPS)*. 2017. 53

16. **Gkioxari, G.**, **R. Girshick**, **P. Dollár**, and **K. He**, Detecting and recognizing human-object interactions. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. 12

17. **Goodfellow, I.**, **J. Pouget-Abadie**, **M. Mirza**, **B. Xu**, **D. Warde-Farley**, **S. Ozair**, **A. Courville**, and **Y. Bengio**, Generative adversarial nets. *In Advances in Neural Information Processing Systems (NIPS)*. 2014. 2, 7, 17, 18, 35, 49, 50, 56

18. **Goroshin, R.**, **J. Bruna**, **J. Tompson**, **D. Eigen**, and **Y. LeCun**, Unsupervised learning of spatiotemporally coherent metrics. *In IEEE International Conference on Computer Vision (ICCV)*. 2015. 40

19. **He, K.**, **G. Gkioxari**, **P. Dollár**, and **R. Girshick**, Mask r-cnn. *In IEEE International Conference on Computer Vision (ICCV)*. 2017. 10, 50, 57, 61

20. **He, K.**, **X. Zhang**, **S. Ren**, and **J. Sun**, Deep residual learning for image recognition. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. 2, 50, 51

21. **Hu, H.**, **J. Gu**, **Z. Zhang**, **J. Dai**, and **Y. Wei**, Relation networks for object detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018*a*. 12

22. **Hu, P.**, **G. Wang**, **X. Kong**, **J. Kuen**, and **Y.-P. Tan**, Motion-guided cascaded refinement network for video object segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018*b*. 10

23. **Hu, Y.-T.**, **J.-B. Huang**, and **A. G. Schwing**, Videomatch: Matching based video object segmentation. *In European Conference on Computer Vision (ECCV)*. 2018*c*. 9

24. **Ilg, E.**, **N. Mayer**, **T. Saikia**, **M. Keuper**, **A. Dosovitskiy**, and **T. Brox**, Flownet 2.0: Evolution of optical flow estimation with deep networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 36, 42, 50

25. **Ioffe, S.** and **C. Szegedy**, Batch normalization: Accelerating deep network training by reducing internal covariate shift. *In International Conference on Machine Learning (ICML)*. 2015. 17

26. **Isola, P.**, **J.-Y. Zhu**, **T. Zhou**, and **A. A. Efros**, Image-to-image translation with conditional adversarial networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 3, 12, 15

27. **Jain, S. D.** and **K. Grauman**, Supervoxel-consistent foreground propagation in video. *In European Conference on Computer Vision (ECCV)*. 2014. 24

28. **Jain, S. D.**, **B. Xiong**, and **K. Grauman**, Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 11, 28, 29, 30

29. **Jampani, V.**, **R. Gadde**, and **P. V. Gehler**, Video propagation networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 10, 15, 28, 30

30. **Karras, T.**, **T. Aila**, **S. Laine**, and **J. Lehtinen**, Progressive growing of gans for improved quality, stability, and variation. *In International Conference on Learning Representations (ICLR)*. 2018. 2

31. **Khoreva, A.**, **R. Benenson**, **E. Ilg**, **T. Brox**, and **B. Schiele**, Lucid data dreaming for object tracking. *In The 2017 DAVIS Challenge on Video Object Segmentation - CVPR Workshops*. 2017. vii, 9, 28, 30, 31, 32, 44, 45, 46, 47, 58

32. **Kingma, D. P.** and **J. Ba**, Adam: A method for stochastic optimization. *In arXiv preprint arXiv:1412.6980*. 2014. 23, 41

33. **Kipf, T. N.** and **M. Welling**, Semi-supervised classification with graph convolutional networks. *In International Conference on Learning Representations (ICLR)*. 2016. 8, 49, 52, 60

34. **Krähenbühl, P.** and **V. Koltun**, Efficient inference in fully connected crfs with gaussian edge potentials. *In Advances in Neural Information Processing Systems (NIPS)*. 2011. 26, 42, 59

35. **Krizhevsky, A.**, **I. Sutskever**, and **G. E. Hinton**, Imagenet classification with deep convolutional neural networks. *In Advances in Neural Information Processing Systems (NIPS)*. 2012. 2, 26, 43

36. **Ledig, C.**, **L. Theis**, **F. Huszár**, **J. Caballero**, **A. Cunningham**, **A. Acosta**, **A. Aitken**, **A. Tejani**, **J. Totz**, **Z. Wang**, *et al.*, Photo-realistic single image super-resolution using a generative adversarial network. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 2

37. **Li, F.**, **T. Kim**, **A. Humayun**, **D. Tsai**, and **J. M. Rehg**, Video segmentation by tracking many figure-ground segments. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013. ix, 13, 16, 24, 31, 33, 46

38. **Li, W.**, **F. Viola**, **J. Starck**, **G. J. Brostow**, and **N. D. Campbell**, Roto++: Accelerating professional rotoscoping using shape manifolds. *In ACM Transactions on Graphics (TOG)*. 2016. 11

39. **Li, X.**, **Y. Qi**, **Z. Wang**, **K. Chen**, **Z. Liu**, **J. Shi**, **P. Luo**, **C. C. Loy**, **X. Tang**, **A. Khoreva**, *et al.*, Video object segmentation with re-identification. *In The 2017 DAVIS Challenge on Video Object Segmentation-CVPR Workshops*. 2017. 1, 10

40. **Lin, T.-Y.**, **M. Maire**, **S. Belongie**, **J. Hays**, **P. Perona**, **D. Ramanan**, **P. Dollár**, and **C. L. Zitnick**, Microsoft coco: Common objects in context. *In European Conference on Computer Vision (ECCV)*. 2014. 57

41. **Liu, W.**, **D. Anguelov**, **D. Erhan**, **C. Szegedy**, **S. Reed**, **C.-Y. Fu**, and **A. C. Berg**, Ssd: Single shot multibox detector. *In European Conference on Computer Vision (ECCV)*. 2016. 2

42. **Long, J.**, **E. Shelhamer**, and **T. Darrell**, Fully convolutional networks for semantic segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015. 9

43. **Luc, P.**, **C. Couprie**, **S. Chintala**, and **J. Verbeek**, Semantic segmentation using adversarial networks. *In NIPS Workshop on Adversarial Training*. 2016. 3, 12, 15

44. **Luiten, J.**, **P. Voigtlaender**, and **B. Leibe**, Premvos: Proposal-generation, refinement and merging for video object segmentation. *In Asian Conference on Computer Vision (ACCV)*. 2018. 1, 10, 59, 60, 62, 66

45. **Maninis, K.-K.**, **S. Caelles**, **Y. Chen**, **J. Pont-Tuset**, **L. Leal-Taixé**, **D. Cremers**, and **L. Van Gool**, Video object segmentation without temporal information. *In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 2018. 9, 59

46. **Maninis, K.-K.**, **J. Pont-Tuset**, **P. Arbeláez**, and **L. Van Gool**, Convolutional oriented boundaries. *In European Conference on Computer Vision (ECCV)*. 2016. 2, 28, 29, 30

47. **Märki, N.**, **F. Perazzi**, **O. Wang**, and **A. Sorkine-Hornung**, Bilateral space video segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. 10, 28, 30, 31, 45

48. **Mathieu, M.**, **C. Couprie**, and **Y. LeCun**, Deep multi-scale video prediction beyond mean square error. *In International Conference on Learning Representations (ICLR)*. 2016. 57

49. **Matthew, M. B. A. R. T.** and **B. Blaschko**, The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. 21

50. **Mobahi, H.**, **R. Collobert**, and **J. Weston**, Deep learning from temporal coherence in video. *In International Conference on Machine Learning (ICML)*. 2009. 40

51. **Oh, S. W.**, **J.-Y. Lee**, **N. Xu**, and **S. J. Kim**, Fast user-guided video object segmentation by interaction-and-propagation networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019*a*. 12

52. **Oh, S. W.**, **J.-Y. Lee**, **N. Xu**, and **S. J. Kim**, Video object segmentation using space-time memory networks. *In IEEE International Conference on Computer Vision (ICCV)*. 2019*b*. 10, 63, 64, 67

53. **Peng, C.**, **X. Zhang**, **G. Yu**, **G. Luo**, and **J. Sun**, Large kernel matters–improve semantic segmentation by global convolutional network. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 9

54. **Perazzi, F.**, **A. Khoreva**, **R. Benenson**, **B. Schiele**, and **A. Sorkine-Hornung**, Learning video object segmentation from static images. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. vii, ix, x, 1, 10, 15, 16, 28, 30, 31, 32, 33, 44, 45, 46, 47, 48, 66

55. **Perazzi, F.**, **J. Pont-Tuset**, **B. McWilliams**, **L. Van Gool**, **M. Gross**, and **A. Sorkine-Hornung**, A benchmark dataset and evaluation methodology for video object segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. vii, ix, 2, 13, 16, 20, 23, 24, 25, 26, 27, 28, 30, 33, 42, 43, 44, 49, 57, 58

56. **Pixel-GCN_Res**. Qualitative results of pixel-gcn model. `https://sites.google.com/view/iconip19-pixelgcn-vos/`. 64

57. **Pont-Tuset, J.**, **P. Arbelaez**, **J. T. Barron**, **F. Marques**, and **J. Malik**, Multiscale combinatorial grouping for image segmentation and object proposal generation. *In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 2017*a*. 13, 29, 30

58. **Pont-Tuset, J.**, **F. Perazzi**, **S. Caelles**, **P. Arbeláez**, **A. Sorkine-Hornung**, and **L. Van Gool**, The 2017 davis challenge on video object segmentation. *In arXiv:1704.00675*. 2017*b*. 50, 57, 58

59. **Prest, A.**, **C. Leistner**, **J. Civera**, **C. Schmid**, and **V. Ferrari**, Learning object class detectors from weakly annotated video. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012. ix, 13, 16, 24, 31, 33, 46

60. **Price, B. L.**, **B. S. Morse**, and **S. Cohen**, Livecut: Learning-based interactive video segmentation by evaluation of multiple propagated cues. *In IEEE International Conference on Computer Vision (ICCV)*. 2009. 11

61. **Radford, A.**, **L. Metz**, and **S. Chintala**, Unsupervised representation learning with deep convolutional generative adversarial networks. *In arXiv preprint arXiv:1511.06434*. 2015. 20

62. **Rahman, M. A.** and **Y. Wang**, Optimizing intersection-over-union in deep neural networks for image segmentation. *In International Symposium on Visual Computing (ISVC)*. 2016. 21

63. **Redmon, J.**, **S. Divvala**, **R. Girshick**, and **A. Farhadi**, You only look once: Unified, real-time object detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. 2

64. **Ren, S.**, **K. He**, **R. Girshick**, and **J. Sun**, Faster r-cnn: Towards real-time object detection with region proposal networks. *In Advances in Neural Information Processing Systems (NIPS)*. 2015. 2

65. **Ren, S.**, **K. He**, **R. Girshick**, and **J. Sun**, Faster r-cnn: Towards real-time object detection with region proposal networks. *In IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 2017. 10

66. **Ronneberger, O.**, **P. Fischer**, and **T. Brox**, U-net: Convolutional networks for biomedical image segmentation. *In International Conference on Medical image computing and computer-assisted intervention (MICCAI)*. 2015. 17

67. **Salimans, T.**, **I. Goodfellow**, **W. Zaremba**, **V. Cheung**, **A. Radford**, and **X. Chen**, Improved techniques for training gans. *In Advances in neural information processing systems*. 2016. 20

68. **Santoro, A.**, **D. Raposo**, **D. G. Barrett**, **M. Malinowski**, **R. Pascanu**, **P. Battaglia**, and **T. Lillicrap**, A simple neural network module for relational reasoning. *In Advances in Neural Information Processing Systems (NIPS)*. 2017. 12

69. **Schütt, K. T.**, **F. Arbabzadah**, **S. Chmiela**, **K. R. Müller**, and **A. Tkatchenko**, Quantum-chemical insights from deep tensor neural networks. *In Nature communications*. 2017. 53

70. **Shankar Nagaraja, N.**, **F. R. Schmidt**, and **T. Brox**, Video segmentation with just a few strokes. *In IEEE International Conference on Computer Vision (ICCV)*. 2015. 11

71. **Sharir, G.**, **E. Smolyansky**, and **I. Friedman**, Video object segmentation using tracked object proposals. *In The 2017 DAVIS Challenge on Video Object Segmentation-CVPR Workshops*. 2017. 10

72. **Shin Yoon, J.**, **F. Rameau**, **J. Kim**, **S. Lee**, **S. Shin**, and **I. So Kweon**, Pixel-level matching for video object segmentation using convolutional neural networks. *In IEEE International Conference on Computer Vision (ICCV)*. 2017. 9

73. **Simonyan, K.** and **A. Zisserman**, Very deep convolutional networks for large-scale image recognition. *In arXiv preprint arXiv:1409.1556*. 2014. 2

74. **Souly, N.**, **C. Spampinato**, and **M. Shah**, Semi supervised semantic segmentation using generative adversarial network. *In IEEE International Conference on Computer Vision (ICCV)*. 2017. 3, 12, 15

75. **TempSeg-GAN_Res**. Qualitative results of tempseg-gan model. `https://sites.google.com/view/visapp19-tempseggan-vos/`. 48

76. **Tokmakov, P.**, **K. Alahari**, and **C. Schmid**, Learning motion patterns in videos. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. 11, 28, 29, 30

77. **Tsai, Y.-H.**, **M.-H. Yang**, and **M. J. Black**, Video segmentation via object flow. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. 10, 15, 28, 30, 31, 44, 45

78. **VidSeg-GAN_Res**. Qualitative results of vidseg-gan model. `https://sites.google.com/view/icvgip18-vidseggan-vos/`. 34

79. **Voigtlaender, P.**, **Y. Chai**, **F. Schroff**, **H. Adam**, **B. Leibe**, and **L.-C. Chen**, Feelvos: Fast end-to-end embedding learning for video object segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. viii, 10, 63, 66, 67

80. **Voigtlaender, P.** and **B. Leibe**, Online Adaptation of Convolutional Neural Networks for Video Object Segmentation. *In British Machine Vision Conference (BMVC)*. 2017. vii, ix, x, 1, 2, 9, 15, 26, 27, 28, 29, 30, 31, 32, 44, 45, 46, 47, 59, 61, 66

81. **Wang, J.**, **P. Bhat**, **R. A. Colburn**, **M. Agrawala**, and **M. F. Cohen**, Interactive video cutout. *In ACM Transactions on Graphics (ToG)*. 2005. 11

82. **Wang, S.**, **Y. Zhou**, **J. Yan**, and **Z. Deng**, Fully motion-aware network for video object detection. *In European Conference on Computer Vision (ECCV)*. 2018. 10

83. **Wang, Z.**, **J. Xu**, **L. Liu**, **F. Zhu**, and **L. Shao**, Ranet: Ranking attention network for fast video object segmentation. *In IEEE International Conference on Computer Vision (ICCV)*. 2019. 11

84. **Watters, N.**, **D. Zoran**, **T. Weber**, **P. Battaglia**, **R. Pascanu**, and **A. Tacchetti**, Visual interaction networks: Learning a physics simulator from video. *In Advances in Neural Information Processing Systems (NIPS)*. 2017. 12

85. **Wug Oh, S.**, **J.-Y. Lee**, **K. Sunkavalli**, and **S. Joo Kim**, Fast video object segmentation by reference-guided mask propagation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. 10, 44, 45, 59, 66

86. **Xiao, F.** and **Y. Jae Lee**, Track and segment: An iterative unsupervised approach for video object proposals. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. 31

87. **Xiao, H.**, **J. Feng**, **G. Lin**, **Y. Liu**, and **M. Zhang**, Monet: Deep motion exploitation for video object segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. 10

88. **Xie, S.** and **Z. Tu**, Holistically-nested edge detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015. 2

89. **Xu, N.**, **B. Price**, **S. Cohen**, **J. Yang**, and **T. S. Huang**, Deep interactive object selection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. 12

90. **Xu, N.**, **L. Yang**, **Y. Fan**, **J. Yang**, **D. Yue**, **Y. Liang**, **B. Price**, **S. Cohen**, and **T. Huang**, Youtube-vos: Sequence-to-sequence video object segmentation. *In European Conference on Computer Vision (ECCV)*. 2018. 63

91. **Xu, S.**, **D. Liu**, **L. Bao**, **W. Liu**, and **P. Zhou**, Mhp-vos: Multiple hypotheses propagation for video object segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019. 11, 63, 64

92. **Yang, L.**, **Y. Wang**, **X. Xiong**, **J. Yang**, and **A. K. Katsaggelos**, Efficient video object segmentation via network modulation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018. 1, 10, 59, 66

93. **Zeiler, M. D.** and **R. Fergus**, Visualizing and understanding convolutional networks. *In European Conference on Computer Vision (ECCV)*. 2014. 17, 24

94. **Zhong, F.**, **X. Qin**, **Q. Peng**, and **X. Meng**, Discontinuity-aware video object cutout. *In ACM Transactions on Graphics (TOG)*. 2012. 11

LIST OF PAPERS

Publications Related to Thesis

1. **VidSeg-GAN: Generative Adversarial Network for Video Object Segmentation Tasks**, Saptakatha Adak and Sukhendu Das, In 11th Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP) [**Short Oral**], IIIT Hyderabad, India, December 18-22, 2018.
   DOI:10.1145/3293353.3293381

2. **TempSeg-GAN: Segmenting Objects in Videos Adversarially using Temporal Information**, Saptakatha Adak and Sukhendu Das; In 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP part of VISIGRAPP) [**Oral**], Prague, Czech Republic, February 25 - 27, 2019. DOI:10.5220/0007254302210232

3. **Motion-based Occlusion-aware Pixel Graph Network for Video Object Segmentation**, Saptakatha Adak and Sukhendu Das, In 26th International Conference on Neural Information Processing (ICONIP) [**Rank - A**] [**Oral**], Sydney, Australia, December 12-15, 2019. DOI:10.1007/978-3-030-36711-4_43
   Achievement: Best Student Paper Award.

Miscellaneous Publications

1. **Things at your Desk: A Portable Object Dataset**, Saptakatha Adak, In 3rd International Conference on Computer Vision and Image Processing (CVIP) [**Oral**], IIITDM, Jabalpur, India, September 29 - October 1, 2018.
   DOI:10.1007/978-981-32-9088-4_36
   Achievement: IAPR Best Student Paper Award.

2. **What's there in the Dark**, Sauradip Nag*, Saptakatha Adak* and Sukhendu Das, In 26th IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, September 22-25, 2019. DOI:10.1109/ICIP.2019.8803299
   Achievement: Top 10% papers.

*Both the authors have contributed equally.

GENERAL TEST COMITTEE

- **Head of the Department**
  Prof. C. Chandra Sekhar
  Professor
  Department of Computer Science and Engineering

- **Chairperson**
  Prof. N. S. Narayanaswamy
  Professor
  Department of Computer Science and Engineering

- **Guide**
  Prof. Sukhendu Das
  Professor
  Department of Computer Science and Engineering

- **Members**
  Prof. C. Siva Ram Murthy
  Professor
  Department of Computer Science and Engineering

  Dr. Sheetal Kalyani
  Associate Professor
  Department of Electrical Engineering

CURRICULUM VITAE

- **Name:** Saptakatha Adak
- **Educational Qualification:**
  Degree: Bachelor of Technology (B.Tech)
  Year: 2016
  Specialization: Computer Science and Engineering
  Institute: Kalyani Government Engineering College, Kalyani, West Bengal

  Degree: Master of Science (M.S. by Research)
  Year: 2020
  Specialization: Computer Science and Engineering
  Institute: Indian Institute of Technology Madras.