UKRAINIAN CATHOLIC UNIVERSITY

BACHELOR THESIS

# Global Motion Understanding in Large-Scale Video Object Segmentation

*Author:*
Volodymyr FEDYNYAK

*Supervisor:*
Roman RIAZANTSEV

*A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science*

*in the*

Department of Computer Sciences and Information Technologies
Faculty of Applied Sciences


Lviv 2023

# Declaration of Authorship

I, Volodymyr FEDYNYAK, declare that this thesis titled, "Global Motion Understanding in Large-Scale Video Object Segmentation" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

*"No matter how difficult it was for us, but now we definitely won't be ashamed ."*

Valerii Zaluzhnyi

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Bachelor of Science

**Global Motion Understanding in Large-Scale Video Object Segmentation**

by Volodymyr FEDYNYAK

# *Abstract*

In this thesis, we show that transferring knowledge from other domains of video understanding combined with large-scale learning can improve robustness of Video Object Segmentation (VOS) under complex circumstances. Namely, we focus on integrating scene global motion knowledge to improve large-scale semi-supervised Video Object Segmentation. Prior works on VOS mostly rely on direct comparison of semantic and contextual features to perform dense matching between current and past frames, passing over actual motion structure. On the other hand, Optical Flow Estimation task aims to approximate the scene motion field, exposing global motion patterns which are typically undiscoverable during all pairs similarity search. We present WarpFormer, an architecture for semi-supervised Video Object Segmentation that exploits existing knowledge in motion understanding to conduct smoother propagation and more accurate matching. Our framework employs a generic pretrained Optical Flow Estimation network whose prediction is used to warp both past frames and instance segmentation masks to the current frame domain. Consequently, warped segmentation masks are refined and fused together aiming to inpaint occluded regions and eliminate artifacts caused by flow field imperfects. Additionally, we employ novel large-scale MOSE 2023 dataset to train model on various complex scenarios. Our method demonstrates strong performance on DAVIS 2016/2017 validation (93.0% and 85.9%), DAVIS 2017 test-dev (80.6%) and YouTube-VOS 2019 validation (83.8%) that is competitive with alternative state-of-the-art methods while using much simpler memory mechanism and instance understanding logic.

# Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| **VOS** | **V**ideo **O**bject **S**egmentation |
| **AOT** | **A**ssociating **O**bjects with **T**ransformers |
| **ISVOS** | **I**nstance **S**egmentation matters in **V**ideo **O**bject **S**egmentation |
| **GPU** | **G**raphics **P**rocessing **U**nit |
| **MOSE** | More Co**m**plex Video **O**bject **Se**gmentation |
| **RAFT** | **R**eccurent **A**ll pairs **F**ield **T**ransform model |
| **GMA** | **G**lobal **M**otion **A**ggregation model |
| **CNN** | **C**onvolutional **N**eural **N**etwork |
| **RMNet** | **R**egional **M**emory **N**etwork |
| **RTB** | **R**efinement **T**ransformer **B**lock |
| **Swin** | **S**hifted **Win**dow transformer |
| **IoU** | **I**ntersection **o**ver **U**nion |
| **FPS** | **F**rames **P**er **S**econd |
| **-T,S,B,L** | -**T**iny, **S**mall, **B**ase, **L**arge, *e.g.*, Swin-**B** (Swin-**B**ase) |

*Dedicated to Motherland Ukraine*

# Chapter 1

# Introduction

Video Object Segmentation (VOS) is a fundamental task in Video Understanding, aiming to segment multiple objects through an entire video sequence. In this work, we address semi-supervised video object segmentation, i.e. the scenario where only the first frame annotations are given, or the annotations are given only for the frames where the corresponding object appears in the video for the first time.

The key feature of Video Object Segmentation is the complete agnosticity of the actual class information for considered objects. This allows a very broad range of possible applications, including but not limited to autonomous driving, sports and video editing.

Prior works achieved significant success in VOS, focusing on making solution highly generalizable and robust under different complex scenarios while maintaining real-time efficiency and low GPU memory footprint. AOT [40] proposed to map objects to a pre-defined set of feature vectors making possible simultaneous processing of many instances. While most works use feature memory to correctly treat occlusions and eliminate errors during propagation, XMem [7] points out the high memory consumption of such an approach and designs efficient unified multi-type memory inspired by Atkinson-Shiffrin model. DeAOT [41] notes the poor performance of existing methods when the objects drastically change in scale and appearance during the video, presenting a novel feature decoupling block to treat such cases more robustly. ISVOS [35] argues that instance understanding matters in VOS and employ an instance segmentation branch based on state-of-the-art instance segmentation architectures increasing the VOS performance for video clips with a high number of similar objects.

Existing approaches rely on dense attention-based feature matching [32] to propagate segmentation masks through the video sequence. Even though this achieves remarkably high scores on existing benchmarks, a single all-pairs correlation search is not capable of capturing global motion context and uncovering relevant motion patterns. In this work, we argue that motion understanding matters in VOS. Inspired by ISVOS proposing to reuse existing instance segmentation architectures to improve instance understanding for VOS domain, we propose to reuse existing optical flow estimation architectures to propagate instance information between video frames.

We present WarpFormer, an VOS architecture that benefits from global motion structure knowledge. We adopt a generic VOS architecture for spatial-temporal matching similar to [40] and replace short-term memory mechanism with optical flow warp, for which we employ a flow estimation network. The propagation process is tackled by optical flow warp while the spatial windowed attention is used to refine warped segmentation mask and inpaint occlusions. Finally, refined mask is fused with long-term memory matches and passed to decoder.

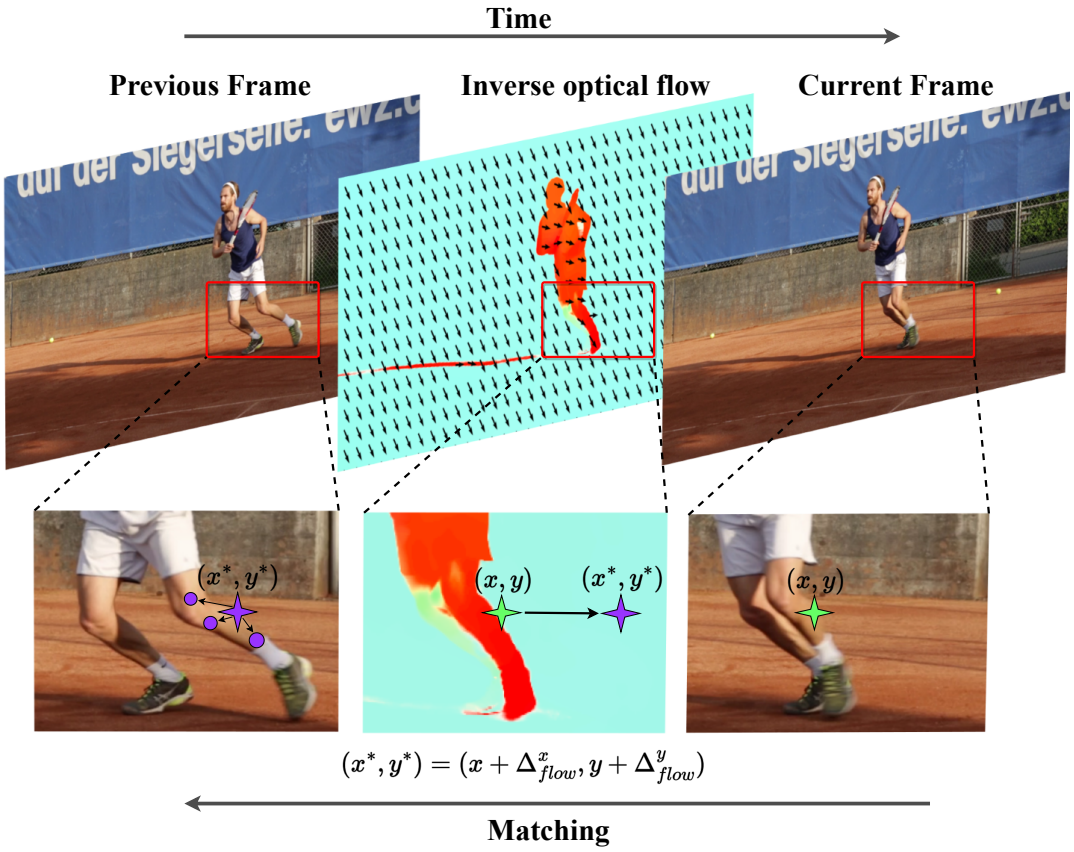



FIGURE 1.1: **Matching process with Optical Flow guidance.** Existing methods adopt windowed attention to perform short-term matching, though, employing a single attention module to perform both motion compensation and semantic search leads to performance degradation. Contrastively, in WarpFormer we decouple motion and semantic contexts to eliminate aforementioned ambiguity.

We conduct additional training of our model on large-scale MOSE 2023 [8] dataset to achieve robustness under complex VOS scenarios. We evaluate our method on DAVIS 2016 & 2017 and YouTube-VOS 2019 benchmarks. Conducted experiments demonstrate that both exploiting global motion structure and large-scale training improve evaluation scores and qualitative results.

In Chapter 2 we provide the generic description of Video Object Segmentation problem along with the central concepts and relevant challenges. Chapter 3 overviews the existing approaches on VOS and optical flow estimation. The proposed WarpFormer architecture is described in details in Chapter 4. The model training details and related hyperparameters are provided in Chapter 5. In Chapter 6 we discuss the conducted experiments and analyze the results. Finally, we conclude the contributions of presented work and outline future research directions in Chapter 7.

# Chapter 2

# Problem Formulation

## 2.1 Video Object Segmentation

This work focuses on problem of semi-supervised Video Object Segmentation. Given a video clip and the segmentation masks for some set of objects for the first frame, and optionally for some other frames, the task is to predict the segmentation masks for the entire video sequence. The reference first frame defines the set of the objects and the initial segmentation masks. As new objects may appear for the first time in the middle of the video, their segmentation masks may be given separately.

## 2.2 Class agnosticity

One of the descriminative features of VOS problem is class agnosticity, i.e. the options for reference objects are not limited with predefined set of classes and could include literally anything. On the other hand, object detection, instance segmentation and video instance segmentation problems are class-dependent making VOS independent and self-contained domain of research. Therefore, class-agnostic formulation settles the clear requirement for VOS methods to have strong generalization power and demonstrate robustness while performing on previously unseen object classes. Another central problem in VOS is developing large-scale properly labeled video datasets covering a wide variety of reference objects categories.


FIGURE 2.1: **Semi-supervised Video Object Segmentation formulation.** For the first frame segmentation masks for some object instances are given. The target is to correctly predict the segmentation masks for all the following frames. Challenging scenarios typically include: similar objects (rows 1, 3, 4), complex occlusions (rows 2, 4), drastic scale (row 1) and shape (rows 2, 3) change. Image taken from [22].

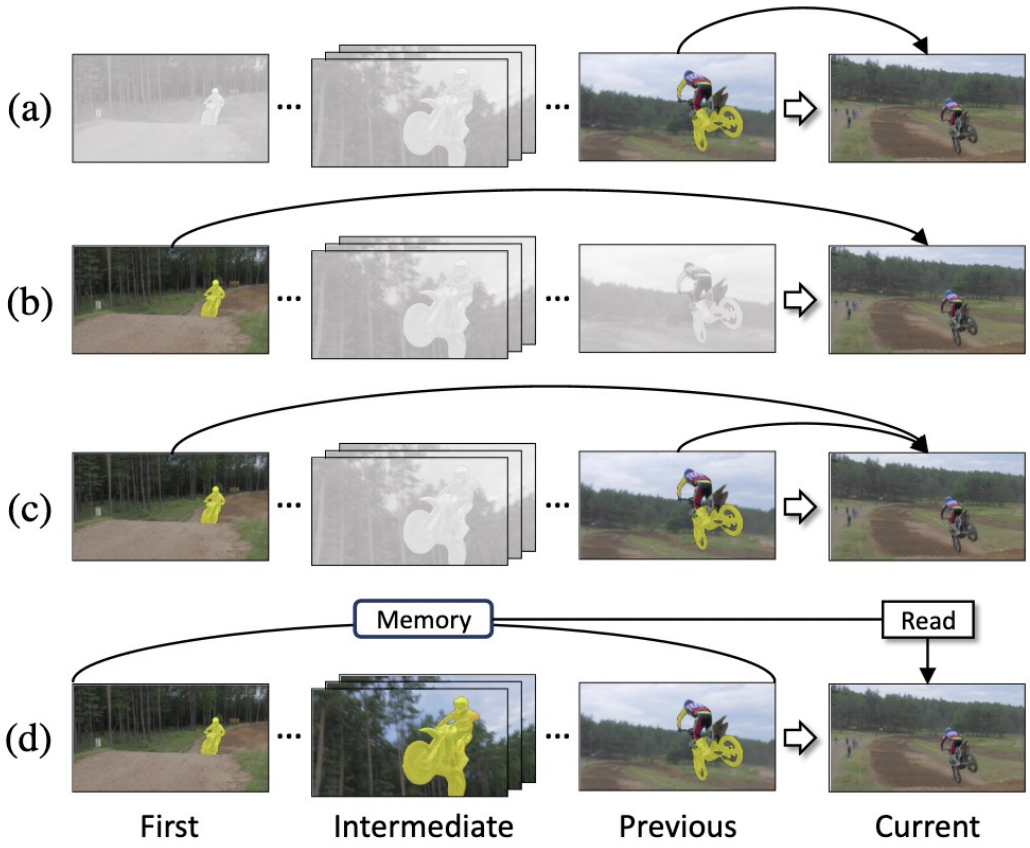



FIGURE 2.2: **Overview of propagation and matching approaches.** Short-term matching (a) adopts only the previous frame for propagation. Long-term matching keeps the first (b) and optionally some intermediate (d) frames in memory bank to perform memory readout. Recent VOS methods usually adopt both short-term and long-term matching making prediction by fusing two matching outputs (c, d). Image taken from [22].

## 2.3 Mask propagation

The two fundamental concepts in VOS approaches are mask propagation and memory matching. Mask propagation is a process of subsequent prediction of object segmentation mask starting from the reference mask given for the first frame. The order of video sequence traversing is strictly aligned with timeline. In mask propagation both the predicted masks for the previous frames and the reference masks for the first frame may be used to predict the mask for the current frame.

## 2.4 Memory matching

Memory matching stands for seeking the similarities between the current and past frames based on visual representations and predicting object masks based on found correspondences. Long-term matching focuses on memorization of objects appearance in different timestamps of video keeping visual representations for different poses, shapes and backgrounds, which is crucial for handling occlusions and demonstrating stable long-time performance. Some approaches introduce short-term matching, sometimes called short-term propagation, i.e. predicting the object mask using


FIGURE 2.3: **Memory bank readout operation.** The multiple frames are kept in memory bank covering different object poses, scales and backgrounds. For the query (i.e. current) frame visual features are matched with visual features stored in memory and the best correspondence is selected. The most important features for the query object are shown as bright regions on memory frames. Image taken from [22].

the immediate predecessor frame and limiting the search space with object neighbour locations assuming smooth motion. Short-term matching ensures temporal consistency and correct handling of dynamic scale and shape changes.

# Chapter 3

# Related Work

## 3.1 Optical Flow Estimation

Optical flow estimation plays a crucial role in our study. The primary function of optical flow is to model the movement of individual points from one image frame to another. Initial efforts in this field centered on optimization problems, maximizing visual similarity with regularization terms [11, 2, 3, 29]. The entry of deep neural networks, particularly convolutional networks, substantially propelled progress in this area.

Early deep learning models such as FlowNet [9] and FlowNet2.0 [14] set the stage for more sophisticated methods. These include SpyNet [26], PWC-Net [30], and LiteFlowNet [13], which employ coarse-to-fine and iterative estimation approaches. Despite their advancements, these models struggled to capture small, fast-moving objects during the coarse stage.

### 3.1.1 Recurrent refinement approaches

The Recurrent All-Pairs Field Transforms for Optical Flow (RAFT) model [31] introduced significant improvements, a novel architecture employing a coarse-and-fine (multi-scale search window per iteration), and a recurrent approach to optical flow estimation.

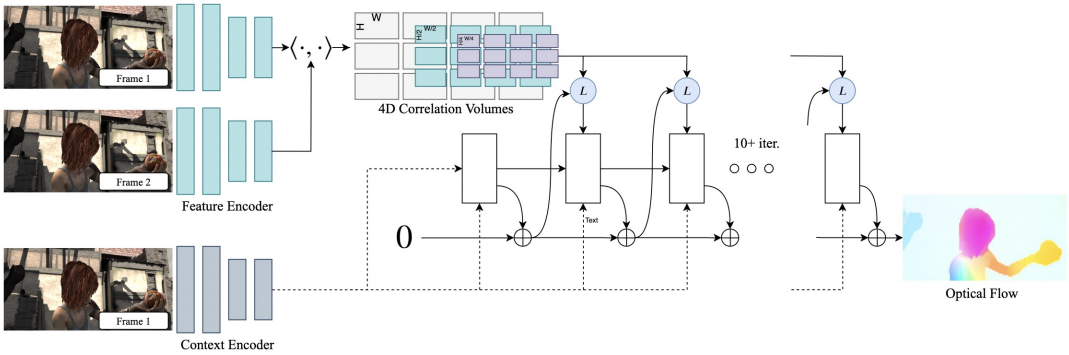



FIGURE 3.1: **RAFT [31] optical flow estimation method architecture overview.** Subsequent video frames are first separately encoded with feature and context encoders. The outputs of feature encoders are used to build 4D correlation volumes representing the feature matching. Recurrent refinement block adopts convolutional GRU to iterativelty update estimated optical flow by querying the 4D correlatation volume with current optical flow and combining the result with context encoder features. Image taken from [31].

Following the introduction of RAFT, subsequent studies like GMA [16] and DEQ-Flow [1] focused on enhancing flow accuracy or improving computational efficiency.

### 3.1.2 Transformer-based approaches

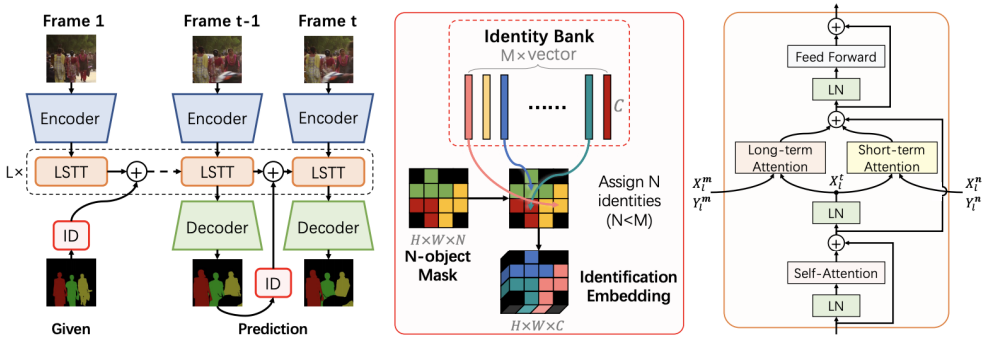
Flowformer [12], a recent state-of-the-art recurrent method, further extends the RAFT architecture. It incorporates a transformer-based strategy that aggregates cost volume in a latent space, building upon the work of Perceiver IO [15]. This was the first model to use transformers [32] for establishing long-range relationships in optical flow, and it achieved state-of-the-art performance. FlowFormer leverages the cost volume as a compact similarity representation and expands the search space globally by aggregating similarity information using a transformer architecture.

Another cutting-edge approach is GMFlow [37], which treats optical flow as a global matching problem and employs a specialized Transformer for feature enhancement, global feature matching, and flow propagation. This approach outperforms the RAFT on the Sintel benchmark while offering greater efficiency [37].

## 3.2 Video Object Segmentation

### 3.2.1 AOT

A key approach that has delivered groundbreaking results in the field of Video Object Segmentation (VOS) is AOT (Associating Objects with Transformers for VOS) [40]. This method relies on a Long Short-Term Transformer (LSTT) block which comprises of self-attention, short-term attention, and long-term attention mechanisms to distill features from input imagery. Here, long-term attention gathers data from extended memory frames, while short-term attention disseminates information from the preceding frame. The outputs from both long-term and short-term attention units are integrated in a feed-forward network, which conveys information to the decoder that subsequently yields the mask estimation for the current frame.




(a) Overview (b) Identity assignment (c) $l$-th LSTT block

FIGURE 3.2: **AOT [40] method architecture overview.** The object mask information is encoded to the embedding space with the identity assignment procedure. Each object is associated with random vector form learnable identity bank. The mask embedding is then combined with frame's visual representation and propagated through time with Long Short-Term Transformer (LSTT) block. LSTT block adopts two separate attention matching branches whose predictions are further fused and passed to a feed-forward network. Image taken from [40].

Moreover, AOT employs a synergistic architecture that integrates an attention map for the attention blocks and a 4D correlation volume, as seen in the RAFT architecture [31], to compute the equivalent spatial correlation between frames. The short-term attention in AOT and the 4D correlation volume in RAFT both determine the correlation between features from successive frames, which can be consolidated in the shared section of the joint architecture as a cohesive motion representation.

### 3.2.2 DeAOT

Building upon the hierarchical propagation concept introduced in AOT, DeAOT [41] (Decoupling Features in Hierarchical Propagation for Video Object Segmentation) presents an advanced method for semi-supervised video object segmentation. DeAOT separates the hierarchical propagation of object-agnostic and object-specific embeddings into two distinct branches to avoid the dilution of object-agnostic visual data in deeper propagation layers. To offset the increased computation stemming from dual-branch propagation, DeAOT debuts a Gated Propagation Module that is meticulously designed with single-head attention. Experimental results illustrate that DeAOT surpasses AOT in both precision and efficiency, setting new SoTA in several benchmarks, including YouTube-VOS, DAVIS 2016, and DAVIS 2017.

### 3.2.3 XMem

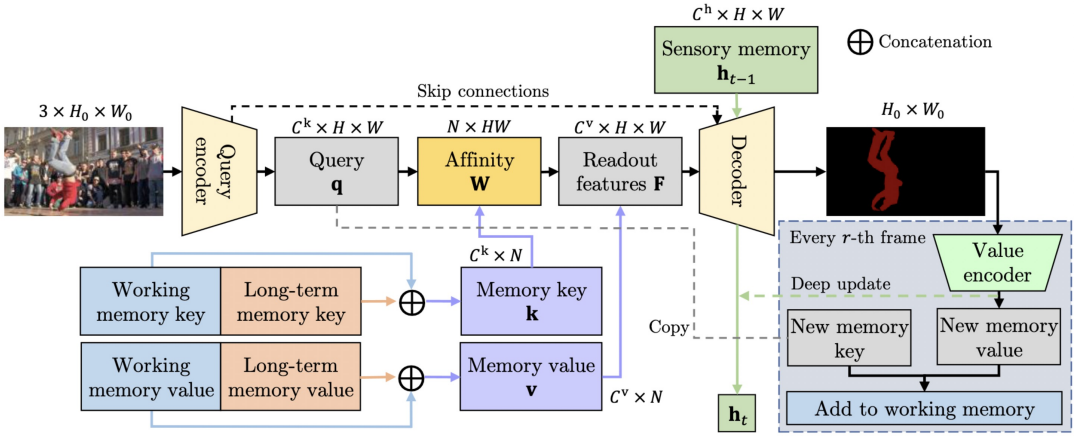



FIGURE 3.3: **XMem [7] method architecture overview.** Sensory memory is updated with every new frame and used for short-term propagation. Working memory keeps a fixed number of previous frames. The most important features from working memory are added to long-term memory via memory consolidation procedure. Image taken from [7].

XMem [7] is a Video Object Segmentation (VOS) architecture that's specifically designed for long videos. Its key innovation lies in the application of the Atkinson-Shiffrin memory model to develop an architecture with multiple independent, yet deeply interconnected feature memory stores. The system incorporates a rapidly updated sensory memory, a high-resolution working memory, and a compact, sustainable long-term memory. A memory potentiation algorithm is employed to regularly consolidate actively used working memory elements into the long-term memory, helping to avoid memory overload and minimizing performance degradation for long-term prediction. Alongside a novel memory reading mechanism, XMem is

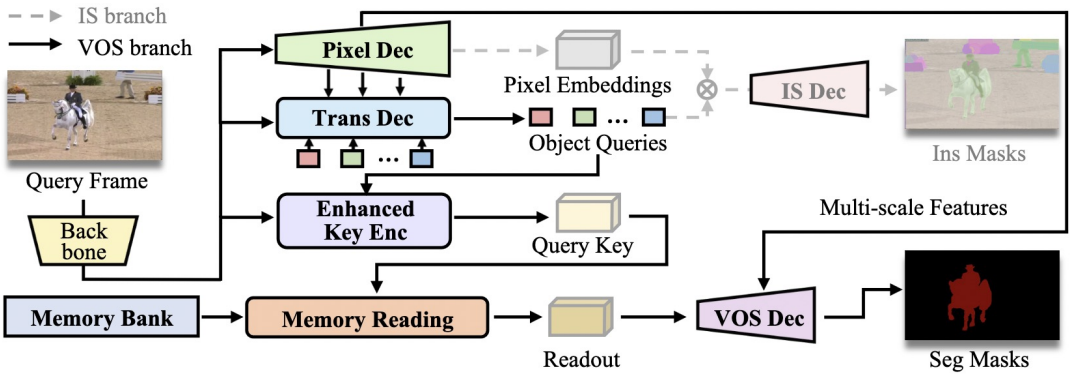



FIGURE 3.4: **ISVOS [35] method architecture overview.** Instance Segmentation branch predicts instance segmentation masks given the set of learnable queries. Transformer decoder's output queries are further refined with Query Enhancement block and considered as features for VOS temporal-spatial matching. Image taken from [35].

able to significantly outperform state-of-the-art techniques on long-video datasets, while maintaining competitive performance on short-video datasets.

### 3.2.4 ISVOS

The paper ISVOS [35] further highlights the importance of instance understanding in VOS. While recent memory-based methods have achieved impressive results in VOS through dense matching between current and past frames, these methods often falter when confronted with large appearance variations or viewpoint changes caused by object and camera movements. To mitigate these issues, the authors propose a two-branch network for VOS, which incorporates a query-based instance segmentation (IS) branch to delve into the instance details of the current frame. This approach allows the integration of instance-specific information into the query key, facilitating instance-augmented matching. These works collectively underscore the importance of instance understanding in VOS and propose solutions that effectively integrate this concept into existing VOS methods.

### 3.2.5 Segmnet Anything

The Segment Anything [18] provides a crucial advancement in the field of image segmentation and, by extension, Video Object Segmentation (VOS). The authors developed an efficient model, the Segment Anything Model (SAM), that was trained on the largest segmentation dataset to date, consisting of over 1 billion masks on 11 million licensed, privacy-respecting images. The model was designed to be promptable, enabling it to perform zero-shot transfer to new image distributions and tasks with impressive performance, often matching or exceeding prior fully supervised results. In the context of VOS, the SAM's feature encoder can be employed to enhance instance understanding, thus addressing a critical challenge in VOS – the ability to discern between instances of objects with similar appearances but different temporal and spatial contexts.

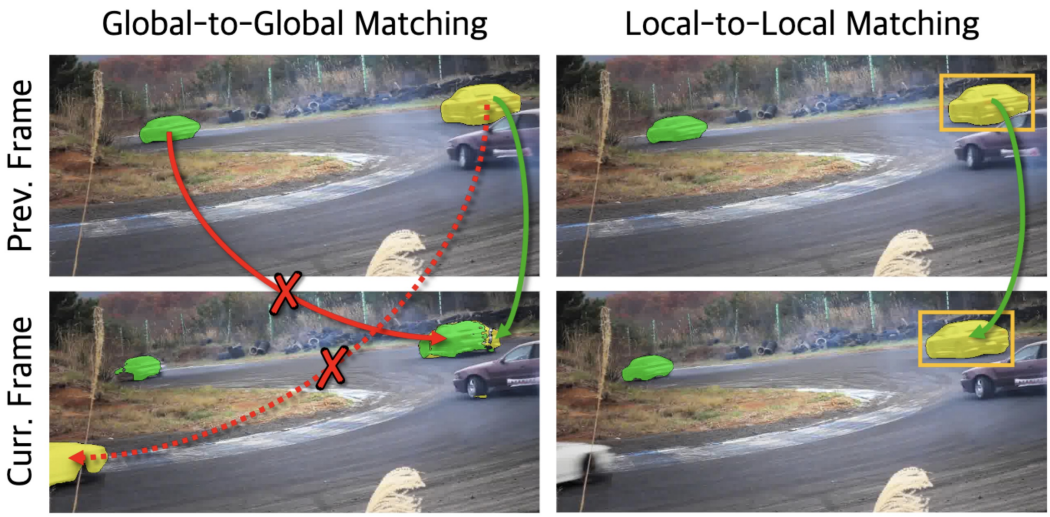



FIGURE 3.5: **RMNet local-to-local matching.** In the case of presence of several similar looking objects global-to-global matching fails to correctly distinguish them while local-to-local matching ensures temporal consistency. Image taken from [36].

## 3.3 Optical Flow-based Video Segmentaiton

Optical flow-based Video Object Segmentation has progressed substantially over time. One of the early works in this domain, MaskTrack [17], combined object segmentation and tracking by employing optical flow for object mask propagation and refining the results using a convolutional neural network (CNN). Building on this foundation, OSVOS [4] further enhanced segmentation performance. More advanced methods like PDB [42] and RVOS [33] emerged, employing multi-stage frameworks and recurrent neural networks, respectively, while still leveraging optical flow.

The Efficient Regional Memory Network (RMNet) [36] has emerged as a compelling method for video object segmentation, showcasing its efficacy in addressing this challenging task. With a focus on efficiency, RMNet leverages a regional memory mechanism to capture and retain relevant information across frames. This approach enables the network to effectively handle long-term dependencies and complex object interactions, leading to accurate and robust segmentation results. RMNet stands out in the landscape of video object segmentation methods by striking a balance between accuracy and computational efficiency, making it suitable for real-time or resource-constrained applications. By incorporating regional memory, RMNet demonstrates its ability to leverage temporal cues and spatial context, empowering it to excel in various scenarios and significantly advance the field of video object segmentation.

We build on these foundational works in our proposed method, utilizing optical flow for short-term frames and attention mechanism for long-term frames to enhance the segmentation process.

# Chapter 4

# Method

## 4.1 Background

Video object segmentation is a challenging task that often involves tracking multiple objects of interest in a single video. Previous approaches to this problem have focused on matching and propagating a single object, requiring independent matching and propagation of each object in multi-object scenarios [34]. This can result in increased GPU usage and inference time, hindering the efficiency of the overall pipeline.

To address this challenge, AOT proposed an identification mechanism for embedding masks of any number into the same high-dimensional space, enabling multi-object scenario training and inference as efficient as single-object ones [40]. This mechanism involves creating a predefined set of $M$ trainable vectors, known as the identity bank, and picking a vector from this bank for each pixel corresponding to a specific class. During training, the vector corresponding to each class is randomly selected to ensure uniform training of the identity bank. To add object-specific information to the feature maps in our architecture, which are at the $\frac{1}{16}$ spatial size of the input video, we adopt a patch-wise identity bank strategy similar to AOT [40]. This involves dividing the input mask into non-overlapping $16 \times 16$ patches, matching each pixel in the patch with the corresponding vector from the identity bank, and obtaining the final result for the identity bank by summing the values for the pixels inside the patch. This operation also encodes some geometry inside the patch and can be implemented as a single $16 \times 16$ convolution.

## 4.2 Warp Refinement Transformer

The straightforward approach for VOS uses optical flow to propagate masks from the previous frame to the current frame. However, occlusions and optical flow imperfections can lead to errors in mask propagation, degrading the quality of the propagated mask with each frame. Additionally, this approach cannot handle newly appeared parts of an object. Our proposed method, WarpFormer, aims to refine the estimated mask using semantic information, which is easier to interpret after motion was decoupled. The overall architecture of WarpFormer is shown in Figure 4.1.

To achieve this, some previous frame $I_k$ and mask $M_k$ is used as a reference point. Our method calculates optical flow using a given Flow Estimator:

$$f_{k \to t} = \text{FlowEstimator}(I_k, I_t)$$

To estimate the current frame mask, the following equation is used:

$$M_k^{warp} = \text{Warp}(M_k, f_{k \to t})$$

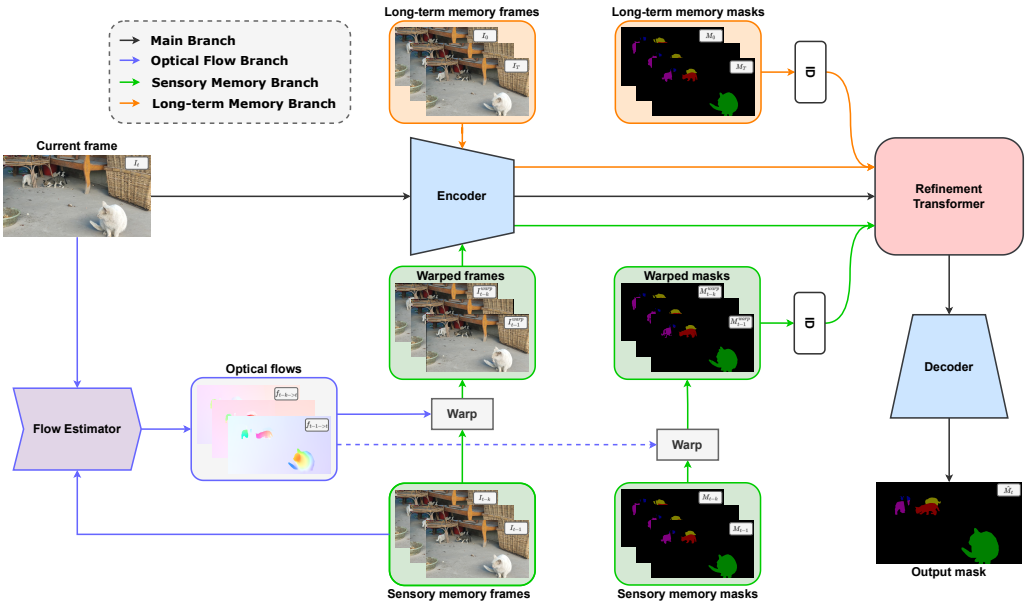



FIGURE 4.1: **The overview of the WarpFormer architecture**. Best viewed in color. Optical flow branch takes the current and previous frames to compute motion fields. Sensory memory branch warps both previous images and predicted masks to the current frame domain. Both long-term and short-term images and masks are transformed to the common embedding space with feature encoder and identity assignment mechanism. Subsequently, latent memory representations are passed to Refinement Transformer which performs long-term and short-term matching. Finally, matching results are fused together and passed to the decoder which reconstructs the original spatial resolution and outputs the predicted segmentation mask.

The method then warps the previous frame $I_k$ using the same optical flow $f_{k \to t}$ to obtain $I_k^{warp}$. Next, the features $X_t$ and $X_k$ are extracted from the current frame $I_t$ and $I_k^{warp}$ using a Feature Encoder and embedding $Y_k$ of our mask $M_k^{warp}$ is formed from an identity bank. Similarly, we create features $X_m$ and identification embedding $Y_m$ from the long-term memory frames $I_m$ with masks $M_m$. The resulting information is fed into our Refinement Transformer Block, which outputs the refined mask $\widehat{M}_t$. Finally, the decoder upsamples the refined mask estimation to the spatial dimensions of the current frame.

## 4.3 Refinemenet Transformer Block

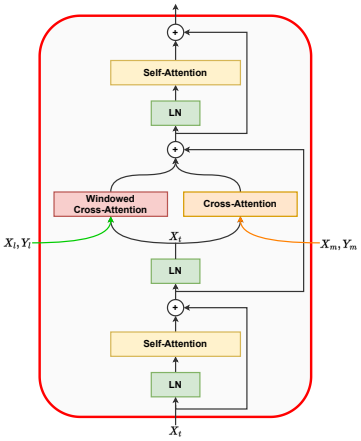
Many recent cutting-edge VOS methods have utilized the attention mechanism and have demonstrated promising results. To define the attention mechanism formally, we consider queries ($Q$), keys ($K$), and values ($V$). The attention operation can then be defined as follows:

$$\text{Attn}(Q, K, V) = \text{Corr}(Q, K)V = \text{softmax}\left(\frac{QK^T}{\sqrt{C}}\right)V,$$

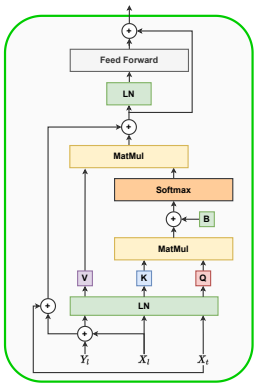
where $C$ is the number of channels.

In our method, we incorporate the identification embedding into the attention operation for mask refinement as follows:

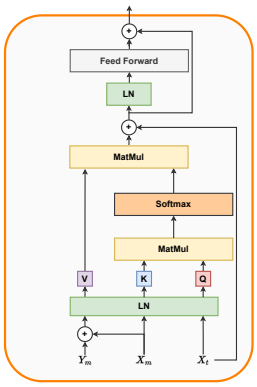
FIGURE 4.2: **The overview of the WarpFormer modules**. Best viewed in color. Firstly,the current image features are passed to self-attention block. Subsequently, features are used for short-term and long-term matching implemented with windowed cross-attention and global cross-attention respectively. Finally, the matching outputs are added and processed with another self-attention. Every attention module is equipped with layer normalization and a skip connection.




(A) Refinement Transformer Block




(B) Windowed Cross-Attention




(C) Cross-Attention

$$\text{AttnID}(Q, K, V, ID) = \text{Attn}(Q, K, V + ID)$$

Following the common transformer blocks, our Refinement Transformer Block (RTB) first employs a self-attention layer on the features of the images to learn the association between the targets within our frames (Figure 4.2a). Our RTB, similarly to the AOT[40], is then divided into two branches: long-term and short-term.

The long-term branch (Figure 4.2c) is responsible for aggregating information from long-term (reference) memory frames. It utilizes simple cross-attention, defined as:

$$\text{CrossAttn}(X_t, X_m, Y_m) = \text{AttnID}(X_t W_k, X_m W_k, X_m W_v, Y_m),$$

where $X_m$ and $Y_m$ are the features and masks embeddings of the long-term memory frames. Besides, $W_k$ and $W_v$ are trainable projections for matching and refinement, respectively.

The short-term (sensory memory) branch (Figure 4.2b) propagates information from the previous frames by taking a look at only some neighboring patches to apply matching. Since image changes between consecutive frames are smooth and continuous, this approach is only more powerful as we convert our previous frames to the current frame domain after warp. The short-term branch utilizes windowed cross-attention:

$$\text{WindowedCrossAttn}(X_t, X_l, Y_l | p) = \text{CrossAttn}(X_t^p, X_l^{N(p)}, Y_l^{N(p)}),$$

where $X_l$ and $Y_l$ are the features and masks embeddings of warped previous frames, $X_t^p$ - feature of $X_t$ at location $p$ and $N(p)$ is a $\lambda \times \lambda$ spatial neighborhood centered at location $p$, where $\lambda$ is window size. We implement windowed cross-attention by including a relative position bias $B$:

$$\text{WindowedCrossAttn}(Q, K, V) = \text{softmax}\Big(\frac{QK^T}{\sqrt{C}} + B\Big)V$$

Finally, the outputs of the long-term and short-term branches are combined together in one more self-attention layer.

# Chapter 5

# Implementation Details

## 5.1 Network details

To study performance capabilities and contributions impact we introduce two variants of network architecture. Namely, **WarpFormer-S** (Small) is an efficient implementation of the proposed method, which adopts MobileNet-V2 [27] as encoder backbone, only a single reference frame is exploited for long-term memory. Alternatively, **WarpFormer-L** (Large) is a large-scale implementation, for which we adopt cutting edge transformer-based encoder Swin-B [20]; following [40], we append every $2^{nd}$ frame to long-term memory bank for training and every $5^{th}$ frame for evaluation. For both architecture variants we use FPN decoder with Group Normalization [19]. We employ Global Motion Aggregation (GMA) [16] as an optical flow estimating network for both WarpFormer-S and WarpFormer-L.

Following [40], we set the number of identification vectors $M$ to 10 in order to align it with the maximum object number in most of benchmarks. For encoders and patch-wise identity bank, their final resolution is $\frac{1}{16}$ as of an input image and mask. For self-attention and cross-attention blocks in Warp Refinement Transformer we use traditional multi-head architecture [32] with Feed-Forward layer and Layer Normalization. The embedding dimension is set to 256, the number of heads is 8 and the hidden dimension of Feed-Forward layers is 1024. For windowed cross-attention used to refine warped sensory memory, we employ original implementation [20] with relative position bias and additionally equip learned relative positional embedding [28]. The window size is set to 15. We also apply fixed sine spatial positional embedding to the self-attention following [5].

## 5.2 Training details

We train both architecture variants in two stages. On the first stage, the model is trained for 40K optimization steps, while the second stage takes 60K steps. During the entire training process, we employ a mixture of DAVIS 2017 [25, 4] train and YouTube-VOS 2019 [38, 39] train datasets in $5:1$ proportion. Additionally, we study adopting MOSE 2023 [8] as additional training data, in which case we apply DAVIS, YouTube-VOS and MOSE mixture with proportion $5:k:p$ where $k+p=1$. Initial value of $k_{start}=0.5$ linearly decays during the training to a final value $k_{end}=0.25$. More detailed description of datasets is presented in Section 6.1. For both stages we use curriculum sampling strategy [23]. Notably, ground truth memory masks are used for temporal-spatial matching during the first stage, while second stage only implies an utilization of the first reference mask providing better supervision for inference setup. Identity banks are frozen after the first stage following [40].

We adopt AdamW optimizer [21] with a one-cycle learning rate schedule. Initial learning rate of $lr_{start} = 3 \times 10^{-4}$ declines to a final value of $lr_{end} = 2 \times 10^{-5}$ in polynomial manner with 0.9 decay factor. We also use learning rate warm-up [10] for 3000 steps. In order to prevent overfitting, we set the learning rate for the encoder to 0.1 of the overall learning rate. Following [7], we use bootstrapped cross entropy and dice losses with equal weighting. For both stages, we use a batch size of 8. WarpFormer-L model training is distributed across four RTX 3090 GPUs, while for WarpFromer-S we use only two RTX 3090 GPUs. The entire training process takes around 40 hours for the large model and 35 hours for the small one.

## 5.3 Video augmentations

We employ a variety of video augmentations to prevent overfitting on the seen data. Specifically, we apply random scaling followed by object-balanced random cropping to the sampled sequence. Additionally, color jitter, random Gaussian blur and random grey-scaling are applied to RGB images.

### 5.3.1 Dynamic merge augmentation

In order to better adapt our model to a multi-object scenario, we adopt dynamic merging augmentation. To enrich generated sequence with more objects, we generate another sequence of the same length from a different video clip and overlay it on the top of the first one. In details, the merging process is as follows: for pair of corresponding frames from the first and second sequence the resulting frame at pixel $(x, y)$, denoted by $I_{merge}(x, y)$, is set to $I_1(x, y)$ if no objects from the second image are present at that pixel, and $I_2(x, y)$ otherwise.

For both training stages we employ the full set of augmentations, for the DAVIS and YouTube-VOS dynamic merge augmentation is applied with probability 0.4, for MOSE merge augmentation is not used since it already features complex multi-object scenes.

# Chapter 6

# Experiments

## 6.1 Evaluation

### 6.1.1 Metrics

In order to evaluate our models we use traditional VOS metrics as proposed in [25].

**$\mathcal{J}$ score for region similarity evaluation.** $\mathcal{J}$ score (Jaccard index) is defined as the intersection-over-union (IoU) rate of the predicted and ground-truth segmentation mask. Given a predicted mask $\widehat{M}$ and ground-truth mask $G$:

$$\mathcal{J} = \frac{|\widehat{M} \cap G|}{|\widehat{M} \cup G|}$$

**$\mathcal{F}$ score for contour accuracy evaluation.** To estimate contour matching accuracy, one finds the contour-based precision $P_c$ and recall $R_c$ between the boundaries of the predicted and ground-truth mask. Subsequently, one computes a F1-score as a simple harmonic mean:

$$\mathcal{F} = \frac{2P_cR_c}{P_c + R_c}$$

Scores are averaged on whole video clip separately for each object. **$\mathcal{J}\&\mathcal{F}$ score** is the average of $\mathcal{J}$ score and $\mathcal{F}$ score presenting a good trade-off between boundary quality and region matching.

### 6.1.2 DAVIS 2016

DAVIS 2016 [24] is a single-object VOS benchmark containing 20 video sequences. Even though single-object scenario is significantly less complex then the multi-object setup, the benchmark features various challenging scenarios including heavy occlusions, objects changing in shape, scale and appearance, fast movements and unfavorable environment settings.

### 6.1.3 DAVIS 2017

DAVIS 2017 [25] benchmark complements DAVIS 2016 with multi-object video clips. It contains 205 different objects and features a 16.1% disappearance rate [8]. Benchmark presents train, validation and test-dev splits containing 60, 30 and 30 sequences respectively. While validation split doesn't introduce a high amount of unseen during training classes, test-dev is much more challenging featuring complex circumstances in most of videos.

We evaluate our method on DAVIS 2016 & 2017 using the default 480p 24FPS videos, not benefiting from full-resolution details. Also we do not apply any test-time augmentations like multi-scale inference [6].

### 6.1.4 YouTube-VOS

YouTube-VOS [38, 39] benchmark introduces a large-scale VOS dataset covering a wide variety of in-the-wild videos. YouTube-VOS 2019 training and validation splits contain 3471, 474 video sequences respectively. Dataset features 91 object categories (7755 objects in total), 26 of which are not present in training split. The explicit annotation of unseen classes is available and the official evaluation tool additionally computes separate metrics for seen and unseen classes to benchmark the generalization power of the approaches. The disappearance rate is only 13% [8], so, in general, YouTube-VOS implies less challenging circumstances compared to DAVIS.

While evaluating our method on YouTube-VOS 2019 validation split we exploit all intermediate frames of the videos to benefit from smooth motion implying more accurate optical flow. Even though we use 24 FPS sequences during evaluation, 6FPS version is used during training and for metric computation.

### 6.1.5 MOSE 2023

MOSE 2023 [8] (Co**M**plex video **O**bject **SE**gmentation) is a novel VOS benchmark featuring extreme scenarios of the video sequence which are not handled good enough by existing VOS methods. The main features of introduced videos include: large number of crowded and similar objects, heavy occlusions by similar looking objects, extremely small-scale objects and reference masks covering only a small region of the whole object. MOSE contains 1507 training and 311 validation video clips with 36 object categories (5200 objects in total). MOSE features overall disappearance rate of 28.8% which is significantly higher compared to classic VOS benchmarks.

FIGURE 6.1: **Sample sequences from MOSE 2023 [8] dataset.** The scenes feature complex occlusions, large number of similar looking objects and poor quality of reference masks. Image taken from [8].

## 6.2 Comparison with State-of-the-art Methods

Our method doesn't adopt complex memory model used in existing methods (XMem [7]), neither it features special architecture injecting instance segmentation logic to benefit from better instance-specific understanding (ISVOS [35]). Also both our small and large models feature only a single transformer block for spatial-temporal matching while existing methods (AOT [40], DeAOT [41]) use up to three blocks. Instead, we incorporate additional training data from MOSE 2023, allowing WarpFormer to tackle scenarios with heavy occlusions, large number of overlapping similar objects or objects dramatically changing in appearance and scale.

TABLE 6.1: **The quantitative evaluation on multi-object benchmarks YouTube-VOS 2019 and DAVIS 2017.** * denotes training on MOSE 2023. Bold denotes the best or three best results.

| | YouTube-VOS 2019 Val | | | | | DAVIS 2017 Val | | | DAVIS 2017 Test | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Methods | $\mathcal{J}_s$ | $\mathcal{F}_s$ | $\mathcal{J}_u$ | $\mathcal{F}_u$ | $\mathcal{J}\&\mathcal{F}$ | $\mathcal{J}$ | $\mathcal{F}$ | $\mathcal{J}\&\mathcal{F}$ | $\mathcal{J}$ | $\mathcal{F}$ | $\mathcal{J}\&\mathcal{F}$ | FPS |
| AOT-T | 79.6 | 83.8 | 73.7 | 81.8 | 79.7 | 77.4 | 82.3 | 79.9 | 68.3 | 75.7 | 72.0 | 51.4 |
| DeAOT-T | **81.2** | **85.6** | **76.4** | **84.7** | **82.0** | **77.7** | 83.3 | 80.5 | **70.0** | **77.3** | **73.7** | **63.5** |
| **WarpFormer-S** | 79.0 | 85.1 | 73.5 | 82.8 | 80.1 | 77.6 | 84.2 | 80.9 | 66.2 | 76.1 | 71.1 | 37.0 |
| **WarpFormer-S*** | 79.0 | 85.3 | 73.1 | 82.5 | 80.1 | 77.8 | **84.3** | **81.0** | 65.9 | 76.1 | 71.0 | 37.0 |
| CFBI+ | 81.7 | 86.2 | 77.1 | 85.2 | 82.6 | 80.1 | 85.7 | 82.9 | 74.4 | 81.6 | 78.0 | 3.4 |
| RMNet | 74.0 | 82.2 | 80.2 | 79.9 | 77.4 | 81.0 | 86.0 | 83.5 | 71.9 | 78.1 | 75.0 | - |
| STCN | 81.1 | 85.4 | 78.2 | 85.9 | 82.7 | 82.2 | 88.6 | 85.4 | 73.1 | 80.0 | 76.1 | 19.5 |
| XMem | **84.3** | 88.6 | **80.3** | **88.6** | **85.5** | **82.9** | **89.5** | **86.2** | **77.4** | 84.5 | 81.0 | 20.2 |
| ISVOS | **85.2** | **89.7** | **80.7** | **88.9** | **86.1** | **83.7** | **90.5** | **87.1** | **79.3** | **86.2** | **82.8** | - |
| Swin-B AOT-L | 84.0 | 88.8 | 78.4 | 86.7 | 84.5 | 82.4 | 88.4 | 85.4 | 77.3 | **85.1** | **81.2** | 12.1 |
| Swin-B DeAOT-L | **85.3** | **90.2** | **80.4** | **88.6** | **86.1** | **83.1** | 89.2 | **86.2** | **78.9** | **86.7** | **82.8** | 15.4 |
| **WarpFormer-L** | 83.2 | 88.9 | 78.1 | 84.9 | 83.8 | 81.1 | 88.9 | 85.0 | 76.4 | 84.9 | 80.6 | **23.9** |
| **WarpFormer-L*** | 83.3 | **89.1** | 78.0 | 85.0 | 83.8 | 82.4 | **89.3** | 85.9 | 76.3 | 84.9 | 80.6 | **23.9** |

### 6.2.1 Quantitative comparison.

The comparison of WarpFormer with other state-of-the-art methods on DAVIS 2017 validation, DAVIS 2017 test-dev and Youtube-VOS 2019 validation validation may be found in Table 6.1. The quantitative comparison with relevant existing methods on DAVIS 2016 validation are listed in Table 6.2a.

Without training on MOSE 2023, our Swin-B WarpFormer-L achieves state-of-the-art performance on DAVIS 2016 single-object benchmark scoring **93.0%** $\mathcal{J}\&\mathcal{F}$. Being evaluated on multi-object benchmarks, model demonstrates highly competitive performance wrapping up with top-ranked scores *i.e.* **85.0%** and **80.6%** $\mathcal{J}\&\mathcal{F}$ on DAVIS 2017 validation and test-dev splits and **83.8%** $\mathcal{J}\&\mathcal{F}$ on Youtube-VOS 2019 validation.

Trained only on Youtube-VOS and DAVIS, our MobileNet-V2 WarpFormer-S outperforms most of its competitors on both single-object and multi-object benchmarks. Namely, it scores **88.9%**, **81.0%** and **71.0%** $\mathcal{J}\&\mathcal{F}$ on DAVIS 2016 validation and DAVIS 2017 validation & test-dev. YouTube-VOS 2019 validation score is **80.1%** $\mathcal{J}\&\mathcal{F}$. We believe that strong and balanced performance under different complex scenarios, simple architecture and lightweight encoder along with agnosticity of actual flow estimation method make WarpFormer-S ideal candidate for usage in various industrial applications.

### 6.2.2 Qualitative comparison.

The qualitative comparison of state-of-the-art approaches and our method is visualized in Fig. 6.2. Existing methods fail to reconstruct fine-grained details under the rapid motion circumstances. In contrast, our method benefits from global motion field and is much more robust to motion blur. On the other hand, adopting MOSE as additional training data gives enough supervision to successfully handle overlapping similar objects without having special architecture design, as instance segmentation branch [35] or feature decoupling module [41].

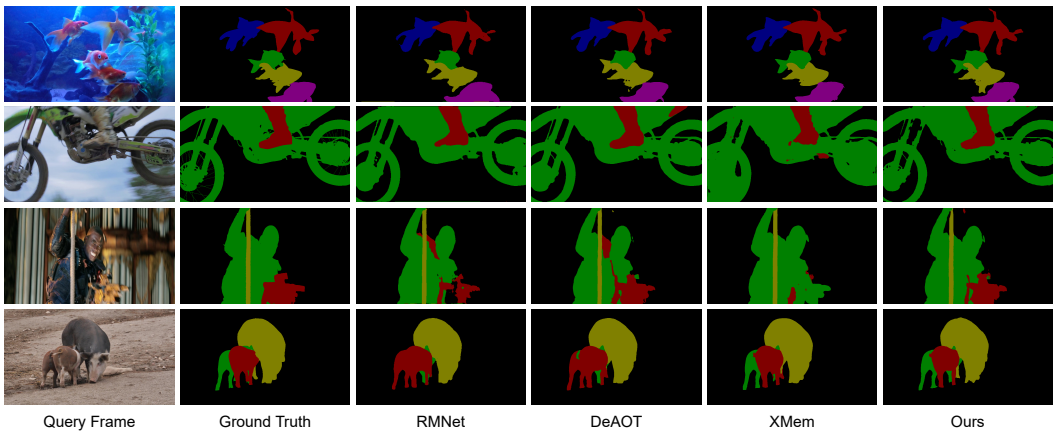



FIGURE 6.2: **Qualitative comparison between WarpFormer and several state-of-the-art VOS methods.** Best viewed in zoom. We don't include ISVOS [35] since there is no source code available. For all methods we used DAVIS2017 val sequences in 480p.

## 6.3 Ablation study

### 6.3.1 Training with MOSE 2023

Adopting MOSE 2023 as training data gives a significant boost on MOSE 2023 validation split so that both our WarpFormer-S and WarpFormer-L models achieve state-of-the-art performance among competitors, scoring **51.7%** and **60.0%** $\mathcal{J}\&\mathcal{F}$ respectively. One the other hand, performance on the classic benchmarks experience an insignificant boost, likely because they doesn't feature any similar extreme scenarios. However, they focus on circumstances with a large number of object classes and classes unseen during training, along with a wide variety of challenging environments, while MOSE 2023 lacks such flexibility. Wrapping up, even minor improvements on classic benchmarks while training with MOSE 2023 indicate the high robustness and performance capacity of the proposed method. The quantitative comparison with other methods on MOSE 2023 validation are listed in Table 6.2b.

TABLE 6.2: **Additional Quantitative comparison.** * denotes training on MOSE 2023. Bold denotes the best result.

(A) **The quantitative evaluation on DAVIS 2016.**

| Methods | $\mathcal{J}$ | $\mathcal{F}$ | $\mathcal{J}\&\mathcal{F}$ |
|---|---|---|---|
| AOT-T | 86.1 | 87.4 | 86.8 |
| DeAOT-T | **87.8** | 89.9 | **88.9** |
| **WarpFormer-S** | 87.2 | **90.5** | **88.9** |
| RMNet | 88.9 | 88.7 | 88.8 |
| STCN | 90.8 | 92.5 | 91.6 |
| XMem | 90.4 | 92.7 | 91.5 |
| ISVOS | **91.5** | 93.7 | 92.6 |
| Swin-B AOT-L | 90.7 | 93.3 | 92.0 |
| Swin-B DeAOT-L | 91.1 | 94.7 | 92.9 |
| **WarpFormer-L** | 90.7 | **95.3** | **93.0** |

(B) **The quantitative evaluation on MOSE 2023.**

| Methods | $\mathcal{J}$ | $\mathcal{F}$ | $\mathcal{J}\&\mathcal{F}$ |
|---|---|---|---|
| STCN | 46.6 | 55.0 | 50.8 |
| RDE | 44.6 | 52.9 | 48.8 |
| SWEM | 46.8 | 54.9 | 50.9 |
| **WarpFormer-S*** | **47.7** | **55.6** | **51.7** |
| XMem | 53.3 | 62.0 | 57.6 |
| Swin-B AOT-L | 53.1 | 61.3 | 57.2 |
| Swin-B DeAOT-L | 55.1 | 63.8 | 59.4 |
| **WarpFormer-L*** | **55.1** | **64.9** | **60.0** |

### 6.3.2 Optical Flow benchmark

We benchmark different optical flow estimation methods during evaluation on DAVIS 2017. As our architecture is completely agnostic to the actual implementation of the flow estimator, we test various approaches in terms of performance / resource requirements trade-off. For RAFT-based models [31, 16, 12], we also try various numbers of iterative flow updates. To demonstrate the impact of flow-warped windowed attention refinement, we also include "zero-flow", which implies identity transformation; in this case, our sensory memory processing degenerates to simple windowed attention similar to [40]. The quantitative comparison may be found in Table 6.3.

The results indicate that our model is indeed optical flow agnostic, and its performance is directly proportional to the quality of the flow. Additionally, for iterative-based optical flow approaches, we observed that a smaller number of iterations was sufficient to achieve fairly good results. This may be attributed to the model's ability to already capture the global motion trend. However, the accuracy of "zero-flow" deteriorated, as our network was trained solely for refinement, rather than direct matching.

TABLE 6.3: **Optical Flow estimator benchmark.** Subscript denotes the number of flow optimization iterations.

| Methods | DAVIS 2017 Val: $\mathcal{J}\&\mathcal{F}$ | Number of parameters | FPS |
|---|---|---|---|
| MobileNet-V2 | | | |
| Zero-Flow | 76.1 | 7.7M | 57.8 |
| RAFT-$S_4$ | 80.5 | 8.7M | 34.7 |
| $RAFT_4$ | 80.7 | 13M | 33.6 |
| $RAFT_{12}$ | 80.7 | 13M | 18.4 |
| $GMA_1$ | 80.2 | 13.6M | 37.0 |
| $GMA_4$ | 80.8 | 13.6M | 27.7 |
| $GMA_{12}$ | 81.0 | 13.6M | 12.6 |
| $GMA_{32}$ | 80.8 | 13.6M | 6.1 |
| FlowFormer | 80.7 | 23.9M | 3.9 |
| Swin-B | | | |
| Zero-Flow | 80.7 | 64.9M | 32.2 |
| $GMA_1$ | 85.0 | 70.8M | 23.9 |
| $GMA_4$ | 85.7 | 70.8M | 15.2 |
| $GMA_{12}$ | 85.9 | 70.8M | 10.0 |
| FlowFormer | 85.9 | 81.1M | 3.6 |

# Chapter 7

# Conclusion and Future Works

## 7.1 Conclusion

This work proposes to reuse existing motion understanding knowledge by adopting optical flow estimation network to support a generic VOS architecture. To integrate global motion structure we replace propagation with optical flow warping and introduce Warp Refinement Transformer block, which aims to inpaint occlusions and fuse warped segmentation mask with long-term memory information. Experimental results show that our method demonstrates strong performance and generalization capabilities. We believe that combining WarpFormer with complex memory mechanisms or specific architecture blocks for instance understanding may further boost it effectivness.

## 7.2 Future Works

The main direction of the future work is investigation on more advanced options of motion context injection. For instance, adopting deformable attention mechanism for short-term matching implies natural consistency with motion fields, *i.e.* utilization of optical flow as strong prior for learnable offsets.

Another research direction may be focused on exploring powerful yet efficient options for image encoders and mask decoders in VOS. Large-scale pretrained Segment Anything ViT-B encoder is supposed to be able of extracting rich instance-aware features while exhibiting the number of parameters similar to ImageNet1K pretrained Swin-B encoder used in this work.

# Bibliography


[1] Shaojie Bai et al. "Deep Equilibrium Optical Flow Estimation". In: *arXiv preprint arXiv:2204.08442* (2022).

[2] Michael J Black and P Anandan. "A framework for the robust estimation of optical flow". In: *Proceedings of the 4th International Conference on Computer Vision*. IEEE. 1993, pp. 231–236.

[3] André Bruhn, Joachim Weickert, and Christoph Schnörr. "Lucas/kanade meets horn/schunck: Combining local and global optic flow methods". In: *International journal of computer vision* 61.3 (2005), pp. 211–231.

[4] S. Caelles et al. "One-shot video object segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 221–230.

[5] Nicolas Carion et al. *End-to-End Object Detection with Transformers*. 2020. arXiv: 2005.12872 [cs.CV].

[6] Siddhartha Chandra and Iasonas Kokkinos. *Fast, Exact and Multi-Scale Inference for Semantic Image Segmentation with Deep Gaussian CRFs*. 2016. arXiv: 1603.08358 [cs.CV].

[7] Ho Kei Cheng and Alexander G. Schwing. "XMem: Long-Term Video Object Segmentation with an Atkinson-Shiffrin Memory Model". In: *ECCV*. 2022.

[8] Henghui Ding et al. "MOSE: A New Dataset for Video Object Segmentation in Complex Scenes". In: *arXiv preprint arXiv:2302.01872* (2023).

[9] Alexey Dosovitskiy et al. "Flownet: Learning optical flow with convolutional networks". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 2758–2766.

[10] Akhilesh Gotmare et al. *A Closer Look at Deep Learning Heuristics: Learning rate restarts, Warmup and Distillation*. 2018. arXiv: 1810.13243 [cs.LG].

[11] Berthold K Horn and Brian G Schunck. "Determining optical flow". In: *Artificial intelligence* 17.1-3 (1981), pp. 185–203.

[12] Ziqi Huang et al. "FlowFormer: A Transformer Architecture for Optical Flow". In: *arXiv preprint arXiv:2203.16194* (2022).

[13] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. "Liteflownet: A lightweight convolutional neural network for optical flow estimation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8981–8989.

[14] Eddy Ilg et al. "Flownet 2.0: Evolution of optical flow estimation with deep networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2462–2470.

[15] Andrew Jaegle et al. "Perceiver IO: A general architecture for structured inputs & outputs". In: *arXiv preprint arXiv:2107.14795* (2021).

[16] Saining Jiang et al. "Learning to estimate hidden motions with global motion aggregation". In: *arXiv preprint arXiv:2104.02409* (2021).

[17] A. Khoreva et al. "Learning Video Object Segmentation from Static Images". In: *arXiv preprint arXiv:1612.02646* (2016).

[18] Alexander Kirillov et al. "Segment Anything". In: *arXiv:2304.02643* (2023).

[19] Tsung-Yi Lin et al. *Feature Pyramid Networks for Object Detection*. 2017. arXiv: 1612.03144 [cs.CV].

[20] Ze Liu et al. "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[21] Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization*. 2019. arXiv: 1711.05101 [cs.LG].

[22] Seoung Wug Oh et al. "Video Object Segmentation Using Space-Time Memory Networks". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 9225–9234. DOI: 10.1109/ICCV.2019.00932.

[23] Seoung Wug Oh et al. *Video Object Segmentation using Space-Time Memory Networks*. 2019. arXiv: 1904.00607 [cs.CV].

[24] F. Perazzi et al. "A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation". In: *Computer Vision and Pattern Recognition*. 2016.

[25] Jordi Pont-Tuset et al. "The 2017 DAVIS Challenge on Video Object Segmentation". In: *arXiv:1704.00675* (2017).

[26] Anurag Ranjan and Michael J Black. "Optical flow estimation using a spatial pyramid network". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4161–4170.

[27] Mark Sandler et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. 2019. arXiv: 1801.04381 [cs.CV].

[28] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. *Self-Attention with Relative Position Representations*. 2018. arXiv: 1803.02155 [cs.CL].

[29] Deqing Sun, Stefan Roth, and Michael J Black. "A quantitative analysis of current practices in optical flow estimation and the principles behind them". In: *International Journal of Computer Vision* 106.2 (2014), pp. 115–137.

[30] Deqing Sun et al. "Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8934–8943.

[31] Zachary Teed and Jia Deng. "RAFT: Recurrent all-pairs field transforms for optical flow". In: *European Conference on Computer Vision*. Springer. 2020, pp. 402–419.

[32] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[33] C. Ventura et al. "RVOS: End-to-end recurrent network for video object segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 5277–5286.

[34] Paul Voigtlaender et al. *FEELVOS: Fast End-to-End Embedding Learning for Video Object Segmentation*. 2019. arXiv: 1902.09513 [cs.CV].

[35] Junke Wang et al. *Look Before You Match: Instance Understanding Matters in Video Object Segmentation*. 2022. arXiv: 2212.06826 [cs.CV].

[36] Haoxiang Xie et al. "RMNet: Equivalently Removing Residual Connection from Networks". In: *arXiv preprint arXiv:2111.00687* (2021).

[37] Hengshuang Xu et al. "GMFlow: Learning Optical Flow via Global Matching". In: *arXiv preprint arXiv:2111.13680* (2021).

[38] Ning Xu et al. *YouTube-VOS: A Large-Scale Video Object Segmentation Benchmark*. 2018. arXiv: 1809.03327 [cs.CV].

[39] Ning Xu et al. *YouTube-VOS: Sequence-to-Sequence Video Object Segmentation*. 2018. arXiv: 1809.00461 [cs.CV].

[40] Zongxin Yang, Yunchao Wei, and Yi Yang. "Associating Objects with Transformers for Video Object Segmentation". In: *arXiv preprint arXiv:2106.02638* (2021).

[41] Zongxin Yang and Yi Yang. "Associating Objects with Transformers for Video Object Segmentation". In: *arXiv preprint arXiv:2210.09782* (2022).

[42] Xuming Zhang et al. "PDB: A multi-stage approach for video object segmentation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 3142–3151.