# One-Shot Video Object Segmentation Using Attention Transfer

Omit Chanda
*Huawei Canada Research Centre*
Markham, Canada
omit.chanda1@huawei.com

Yang Wang
*University of Manitoba*
Winnipeg, Canada
ywang@cs.umanitoba.ca

***Abstract*—In this paper, we consider the problem of one-shot object segmentation in videos. Given an input video where the object mask of the first frame is provided, our goal is to segment each remaining frame in the video into foreground and background. We propose an attention based knowledge transfer mechanism that transfers the object knowledge from the first frame to other frames in a video. Our model is a Siamese network with two streams. The first stream will process the first frame in a video, and the second stream will produce segmentation mask of any other frame in a video. Each stream is a convolutional neural network (CNN) that produces attention maps in certain layers. Our proposed approach is based on the observation that the attention maps in CNN contain valuable information that can boost the performance of CNN architectures. In our work, we propose a method for transferring the attention maps from the first stream to the second stream in the Siamese architecture. This will allow our model to transfer the knowledge from the first frame (with ground-truth segmentation mask) to other frames in the video. Our experimental results on two benchmark datasets demonstrate that our proposed model outperforms other state-of-the-art approaches.**

***Index Terms*—one-shot learning, object segmentation, attention transfer**

## I. Introduction

We consider the problem of one-shot video object segmentation (OSVOS). During training, we have access to a collection of videos that are fully annotated, i.e. the segmentation mask of the object of interest is provided in every frame of a training video. During testing, we are given an input test video and the segmentation mask of the object of interest in the first frame, our goal is to generate the segmentation masks of the remaining frames in the test video. Figure 1 shows an illustration of our problem setting.

One-shot video object segmentation is an important problem with many real-world applications, such as video search, video surveillance, etc. This problem is also very challenging. Standard object detection only tries to find instances of a particular object class in an image. In contrast, OSVOS has to be able to handle any object class in a video. As a result, OSVOS cannot use models trained for a particular object class and has to be able to handle large variations of objects. Second, since we only have one annotated frame in a test video, it is difficult to apply standard deep convolutional neural networks (CNNs) to learn a model specific to this test video, since CNNs require a large amount of labeled training data.

Early approaches [1], [2] for this problem usually rely on hand-crafted features. These approaches do not leverage CNNs which have been shown to be powerful in many vision tasks, e.g. image classification [3], object detection [4], semantic segmentation [5]. Recently, there have been efforts [6], [7] on adopting CNNs for OSVOS. One of the best performing approaches, OSVOS [6], uses the labeled training videos to learn an initial model (called *parent network* in [6]). For a test video, this parent network is fine-tuned based on the first frame and its ground-truth segmentation mask. Intuitively, this fine-tuning approach can be considered as a knowledge transfer mechanism. Through fine-tuning, the knowledge about the object of interest is implicitly captured by the fine-tuned model parameters. When these model parameters are used to segment the object in remaining frames, the knowledge about the object is implicitly transferred to these frames.

In this paper, we introduce an another strategy of knowledge transfer for OSVOS. Our approach is inspired by the attention transfer in [8]. Deep learning models with attention mechanism have shown great success in various computer vision task, e.g. object detection [9], image captioning [10]. The observation in [8] is that the attention maps used in CNNs actually contain valuable information that can be used to significantly improve the performance of CNN architectures. Based on this observation, the method in [8] proposes to transfer the attention maps from a powerful network (called *teacher network*) to a smaller network (called *student network*). The goal is to improve the performance of the student network.

In this work, we propose to use attention transfer in OSVOS. We consider the teacher network to be the CNN model applied on the first frame in a video, and the student network to be the CNN model applied on any other frame in the video. By transferring the attention maps from the teacher network to the student work, we can improve the performance of the student network.

The contributions of this work are three fold. First, we propose a novel Siamese architecture with two parallel streams for OSVOS. The two streams correspond to the teacher network and the student network, respectively. Second, instead of using the attention regularization in [8], we propose a much simpler attention transfer strategy by directly adding the attention maps

978-1-7281-1817-8/19/$31.00 ©2019 IEEE

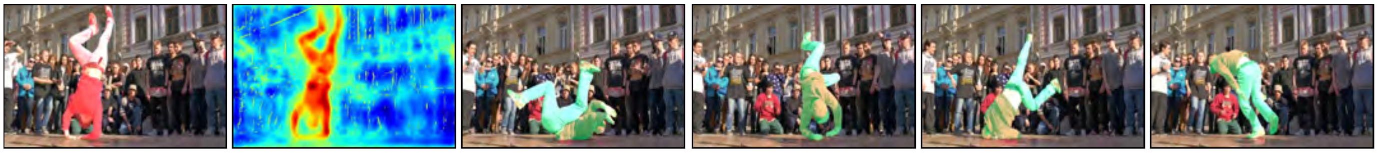

Fig. 1. An illustration of our problem formulation. Given an input video, the object mask of the first frame (red) is provided (1st column). The first frame is fed to our model and produces an attention map (2nd column). This attention map is transferred to remaining frames in order to produce the segmentation masks (green) in these remaining frames.

from the teacher network to the student network. Finally, we combine the attention transfer strategy and the fine-tuning method in [6] for OSVOS. Our experimental results show that our proposed approach outperforms all the other state-of-the-art methods.

## II. Related Works

In this section, we review different lines of previous work that are closely related to our work: one-shot learning and video object segmentation.

**One-shot Learning:** The goal of one-shot learning is to acquire knowledge based on training examples and generalize it for new classes with only one or few annotated examples. Wang and Herbert [11] introduce a learning to learn approach for predicting classifiers which is very close to the base classifier. Bertinetto et al. [12] introduce a learning network to predict the weights of their final predictor. Vinales et al. [13] develop a matching network that learns a class based on only one or few examples. Santoro et al. [14] propose an one-shot learning approach by adding a memory module in CNN.

**Video Object Segmentation:** In video object segmentation, we are given only a small number of annotations (e.g. only the first frame) in a given video. Some work [6], [15] in this area has exploited using temporal consistency between successive frames. There is also a line of work [16] on reducing computational complexity or large number of parameters in video object segmentation. Caelles et al. [6] propose an approach for semi-supervised video object segmentation where they segment the foreground object from the background based on the annotation masks of only one or few frames. They adopt a pre-trained CNN from image recognition and tune-fine their network on a labeled frame in an input video. Khoreva et al. [17] develop a method called MaskTrack that takes advantage of the prediction mask from previous frames. In MaskTrack, the predicted segmentation mask of a frame is used as an additional input to boost up the performance for next frame. Cheng et al. [18] introduce an approach to jointly predict segmentation and optical flow. Jampani et al. [7] uses bilaterial filter for video object segmentation. Jang et al. [19] introduce a trident network to incorporate optical flow propagation in video object segmentation.

## III. Our Approach

Our model is in the form of a Siamese network with two parallel streams (see Fig 2). Each stream of the Siamese network is an attention-based segmentation network (Sec. III-A) that takes an image as its input and produces the segmentation mask. The model parameters are shared by the two streams in the network. During testing, the first stream (which we call the *teacher stream*) will be used to process the first frame of an input video and produce the corresponding segmentation labels. The second stream (which we call the *student stream*) will be used to produce the segmentation mask of any other frame in the video. Both the teacher and student networks will also generate attention masks for their corresponding feature maps. During testing, the teacher stream will be fine-tuned based on the ground-truth segmentation mask of the first frame. We then transfer the attention masks from the teacher network to the student network by adding their attention masks. This attention transfer mechanism allows us to transfer the knowledge of the first frame to another frame in the video.

### A. Attention-based Segmentation Network

The backbone architecture of the Siamese network is based on the foreground FCN model in OSVOS [6]. The model in [6] consists of groups of convolution, max-pooling, and ReLU activation layers grouped into 5 stages. For a given input image, the network extracts four feature maps at different scales $X^1$, $X^2$, $X^3$ and $X^4$. These feature maps are upsampled and fused together to produce the segmentation label mask which has the same spatial dimension as the input image.

We modify the model in [6] to add an attention layer after each of the 4 feature maps. We use a technique similar to [8] to generate the attention maps. Let $X \in \mathcal{R}^{C\times H\times W}$ be a feature map in a CNN model, where $C$ is number of channels and $H \times W$ are the spatial dimensions of the feature map. We generate the corresponding attention map using a mapping function $\mathcal{F}$:

$$\mathcal{F} : \mathcal{R}^{C\times H\times W} \rightarrow \mathcal{R}^{H\times W} \tag{1}$$

The mapping function $\mathcal{F}$ takes the feature map $X$ as input which contains the data vector $X_{ij}$ for the location $(i, j)$ to produce the attention score $A_{ij}$ for each particular location. In our work, we implement $\mathcal{F}$ as follows. First, we apply a 2D convolution to map the input feature map (with dimension $C \times H \times W$) to a 2D map (with dimension $H \times W$). This is followed by applying a sigmoid function to normalize the entries of this 2D map to be between 0 and 1.

Let $A = \mathcal{F}(X)$ (where $A \in \mathcal{R}^{H\times W}$) be the attention map for a given feature map $X \in \mathcal{R}^{C\times H\times W}$. We use the attention

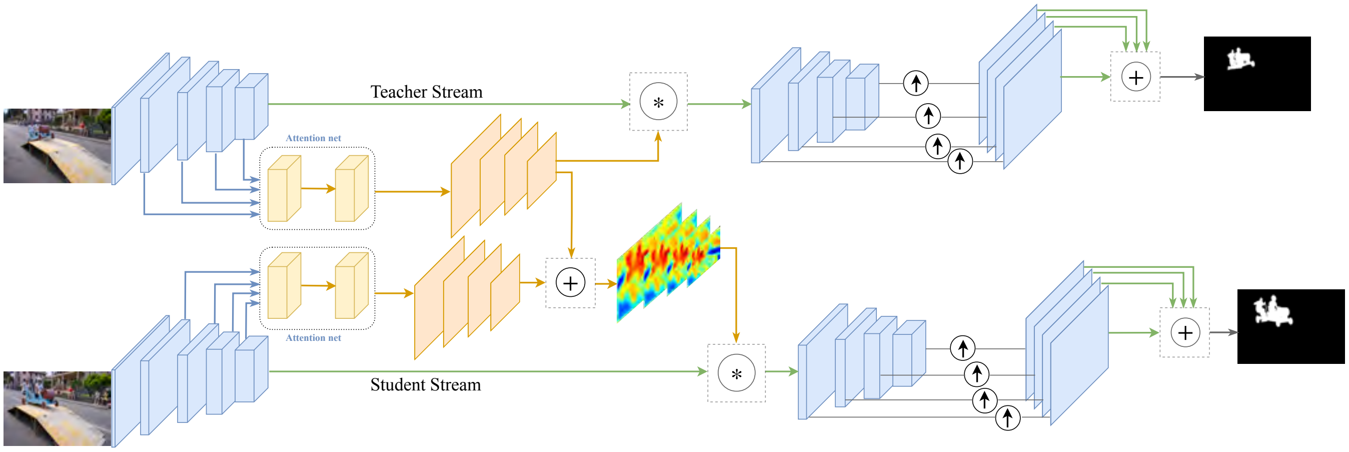

Fig. 2. Overview of our proposed framework. Our model has a Siamese architecture with two parallel streams that operates on first frame (teacher stream) and remaining frames (student stream). In the teacher stream, we have the ground-truth object segmentation mask for the first frame of a video. We produce an object specific attention map in the teacher stream. The attention map from the teacher stream will be transferred to the student stream. The student stream will take any remaining frame as its input and produce its segmentation mask using the knowledge encoded by the attention map from the teacher stream.

map $A$ to generate an attention-weighted feature map $Z \in \mathcal{R}^{C\times H\times W}$:

$$\begin{aligned} Z[c,h,w] &= A[h,w] \times X[c,h,w], \\ &\forall c \in \{1..C\}, h \in \{1..H\}, w \in \{1..W\} \end{aligned} \tag{2}$$

where $Z[c,h,w]$ ($A[h,w]$ and $X[c,h,w]$) denotes a particular entry in $Z$ ($A$ and $X$).

Since our backbone architecture (i.e. the FCN model in [6]) produces four feature maps at different scales $X^i$ ($i = 1,2,3,4$), we will have four attention-weighted feature maps $Z^i$ ($i = 1,2,3,4$). We then upsample $Z^i$ ($i = 1,2,3,4$) to have the same spatial dimension of the input image. Finally, these attention-weighed feature maps are concatenated together to produce the predicted segmentation mask.

For a frame $x$ of a video, we define the loss on this frame using the pixel wise cross entropy loss in [6] on the final classification score of the predicted mask for the binary classification of the frame $x$:

$$\begin{aligned} \mathcal{L}(W) = -\sum_j y_j \log P(y_j = 1 \mid x; W) + \\ (1 - y_j)\log(1 - P(y_j = 1 \mid x; W)) \end{aligned} \tag{3}$$

Eq.3 also can be represented as,

$$\begin{aligned} \mathcal{L}(W) = -\sum_{j\in Y_+} \log P(y_j = 1 \mid x; W) - \\ \sum_{j\in Y_-} \log P(y_j = 0 \mid x; W) \end{aligned} \tag{4}$$

In Eq.4, $W$ represents the model parameters. For each frame $x$ of a randomly picked video, $y_j$ represents the pixel wise label of $x$ where $y_j \in 0,1$. $Y_+$ and $Y_-$ represents positive and negative labeled pixels respectively. The probability $P(.)$ is obtained by applying a sigmoid function on the final classification scores at the last layer in the network. To handle the imbalance dataset, we use the modified cost function similar to [6] as follows:

$$\begin{aligned} \mathcal{L}(W) = -\beta \sum_{j\in Y_+} \log P(y_j = 1 \mid x; W) \\ -(1-\beta) \sum_{j\in Y_-} \log P(y_j = 0 \mid x; W) \end{aligned} \tag{5}$$

In Eq.5, $\beta$ represents $|Y_+|/|Y_-|$. Finally we learn model parameters by optimizing Eq.5 using stochastic gradient descent.

### B. Attention Transfer

During testing, we are given an input video and the ground-truth object segmentation mask of the first frame. The first frame will be used as the input to the teacher stream in our network. Similar to [6], we then fine-tune the parameters of the teacher stream on the first frame using its ground-truth segmentation mask. We use $A_t^i$ ($i = 1,2,3,4$) to denote the attention maps generated in the teacher stream by the fine-tuned network.

The student stream of our network will receive another frame in the video as its input. Note that since we do not have the ground-truth segmentation mask for this frame during testing, we cannot directly fine-tune the student stream. Instead, we directly replace the parameters in the student stream by copying the model parameters from the fine-tuned teacher stream.

We also like to transfer the knowledge encoded by the attention maps from the teacher stream to the student stream. In [8], this attention transfer is achieved by introducing a regularization that encourages the attention maps from two networks to look similar. In our work, we use a much simpler (yet effective) attention transfer mechanism. Let $A_t^i$ and $A_s^i$ ($i = 1, 2, 3, 4$) be the attention maps of the teacher stream and the student stream, respectively. Note that these attention maps are produced using model parameters obtained from fine-tuning the teacher stream. We directly add the attention maps $A_t^i$ to $A_s^i$ as:

$$A_{new}^i \leftarrow A_s^i + A_t^i, \quad i = 1, 2, 3, 4 \tag{6}$$

Then the updated attention maps $A_{new}^i$ in Eq. 6 will be used to generate the attention-weighted feature maps $Z_t^i$ and finally produce the final predicted segmentation mask in the student stream.

The intuition of our approach is as follows. Since the teacher stream has access to the ground-truth annotation on the first frame, it contains useful information about the object specific to the input. Intuitively we would like to transfer the knowledge from the teacher stream to the student stream. In our approach, this knowledge transfer is achieved via two ways. First, the model parameters are fine-tuned on the teacher stream and used in the student stream during testing. Second, the attention maps of the teacher stream are directly added to the student stream (Eq. 6). In the experiments, we demonstrate that both knowledge transfer mechanisms help improve the final performance of our model.

### C. Training Details

Similar to [6], we start with an ImageNet pretrained VGG model [20] as the *base network*. We initialize teacher stream and the student stream with the base network. We then perform offline training on a set of training videos. The object segmentation masks are provided for all frames for any training video. Following [6], we call the model obtained after the offline training the *parent network*. Given a test video where only the first frame is annotated, we use the parent network as the initialization and perform an online training to fine-tune the model to this particular video. The fine-tuned model is called the *test network*. We then use the test network to segment all remaining frames in the input test video.

**Offline training**: In each iteration of the offline training, we randomly pick a training video. The first frame of this video is forwarded to the teacher stream of our model. A batch of remaining frames of the video are forwarded to the student stream. The teacher and student streams predict the segmentation masks for their corresponding input frames. The attention maps from the teacher stream are transferred to the student stream as described in Sec. III-B. We use the pixel-wise cross entropy loss (Eq. 5) from both streams to learn the model parameters. After the training, we have obtain the parent network.

**Online training**: Given an input test video, we perform online training to fine-tune the parent network on this test video. Since we only have the ground-truth label of the first frame, we only use the pixel-wise cross entropy loss from the teacher frame to fine-tune the network. After the online training, we have obtained a test network specific tuned on this test video.

| Method | DAVIS | YouTube-Objects |
|---|---|---|
| OFL [24] | 68.0 | 77.6 |
| BVS [25] | 60.0 | 68.0 |
| SFL [18] | 76.1 | – |
| PLM [26] | 70.2 | – |
| VPN [7] | 75.0 | – |
| CTN [19] | 73.5 | – |
| MaskTrack [17] | 80.3 | 72.6 |
| OSVOS [6] | 79.8 | 78.3 |
| ours | **81.1** | **79.7** |

TABLE I

COMPARISON BETWEEN OUR APPROACH AND OTHER STATE-OF-THE-ART METHODS ON DAVIS VALIDATION SET AND YOUTUBE-OBJECTS DATASETS. NOTE THAT EXPERIMENTAL RESULTS FOR MASKTRACK ON DAVIS ARE REPORTED BASED ON ALL VIDEO SEQUENCES INCLUDING THE TRAINING SET, BUT IN OUR EXPERIMENT WE USE ONLY VALIDATION SET FOR THE EVALUATION. OUR METHOD ACHIEVES BEST PERFORMANCE COMPARED TO OTHER BASELINES.

We then apply the test network to segment the remaining frames in the test video. Given a test frame, we also transfer the attention maps from the teacher stream (obtained by passing the first frame) to the student stream to generate the segmentation mask on this test frame. This process is repeated until all remaining frames are processed.

In order to further improve the performance, most state-of-the-art methods also perform some post-processing, such as conditional random field (CRF) in [17], boundary snapping in [6]. As a post-processing step, we apply a DenseCRF [21] to smooth the predicted segmentation mask.

## IV. EXPERIMENTS

We evaluate our approach on two benchmark datasets: DAVIS [22] and Youtube-Objects [23]. We also perform ablation analysis to study the relative contribution of each component of our proposed method.

### A. Implementation Details

Our network is implemented based on the popular deep learning framework PyTorch. For training our network, and evaluating the performance of it, we use Nvidia Titan X with 10 core 2.3GHZ, 64GB RAM, 4TB hard drive and NVidia GTX980 GPUs with 4GB, 2048 CUDA cores. To initialize the network, we use some of the parameters from FCN [5]. We use stochastic gradient descent to update the weights of our two streams siamese network. During testing, our network receives an input image at its original size and then it produces object segmentation mask at the original resolution for each test image.

### B. Results

**DAVIS dataset:** The DAVIS dataset consists of 50 videos including 30 videos for training and the remaining 20 videos for testing. Each video has one object of interest. The pixel-level annotation of the object is provided on each frame of a video. DAVIS has two versions based on the image resolution, we use the $854 \times 480$ images for all of our experiments.

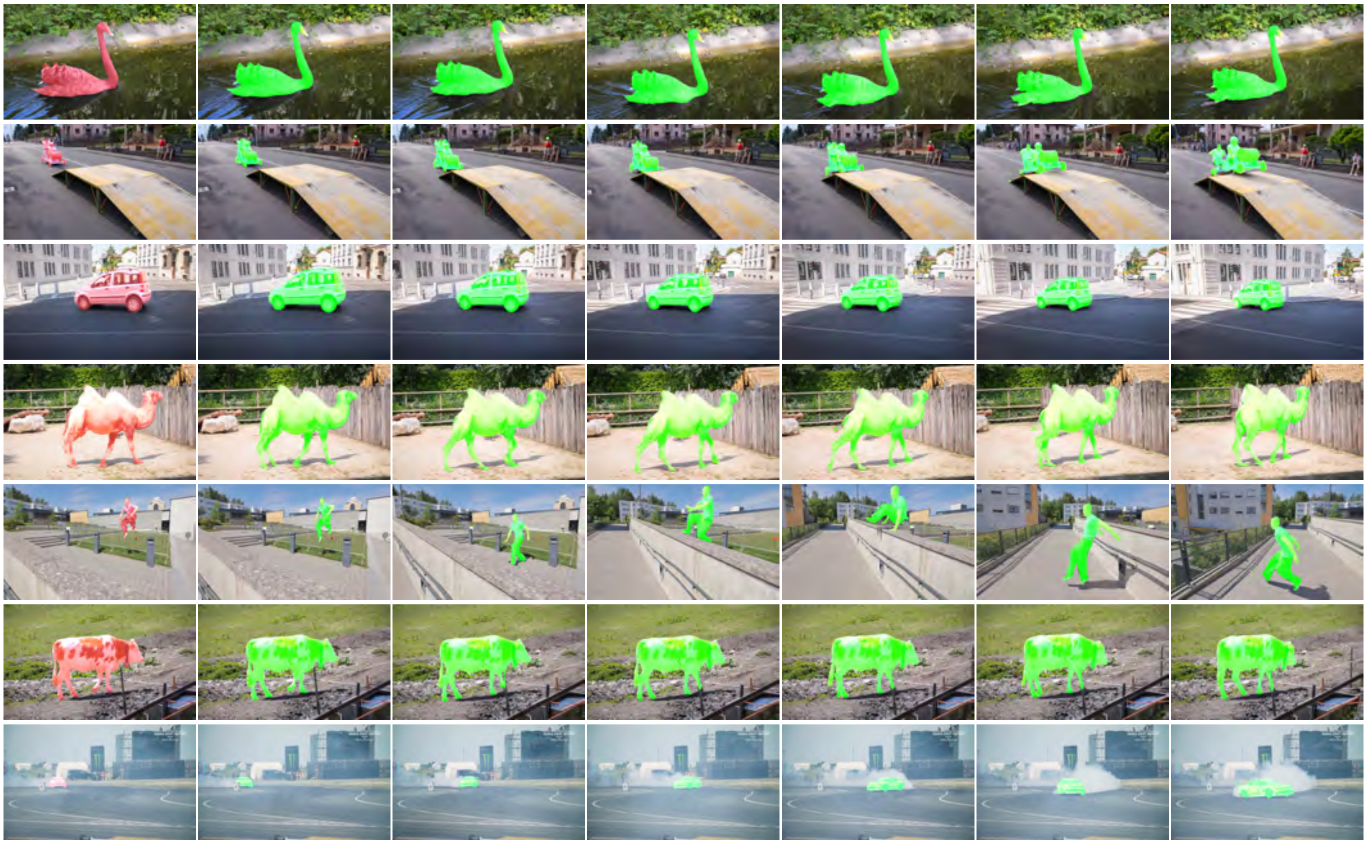

Fig. 3. Qualitative results on videos in the DAVIS validation set. The ground-truth forground object mask on the first frame is shown in Red. The predicted forground object masks in other frames are shown in Green.

| Category | LTV | HBT | FST | AFS | BVS | SCF | JFS | OFL | OSVOS | Ours |
|---|---|---|---|---|---|---|---|---|---|---|
| Aeroplane | 13.7 | 73.6 | 70.9 | 79.9 | 86.8 | 86.3 | 89.0 | **89.9** | 88.2 | 88.1 |
| Bird | 12.2 | 56.1 | 70.6 | 78.4 | 80.9 | 81.0 | 81.6 | 84.2 | 85.7 | **86.2** |
| Boat | 10.8 | 57.8 | 42.5 | 60.1 | 65.1 | 68.6 | 74.2 | 74.0 | 77.5 | **79.2** |
| Car | 23.7 | 33.9 | 65.2 | 64.4 | 68.7 | 69.4 | 70.9 | 80.9 | 79.6 | **81.4** |
| Cat | 18.6 | 30.5 | 52.1 | 50.4 | 55.9 | 58.9 | 67.7 | 68.3 | 70.8 | **73.6** |
| Cow | 16.3 | 41.8 | 44.5 | 65.7 | 69.9 | 68.6 | 79.1 | **79.8** | 77.8 | 77.0 |
| Dog | 18.0 | 36.8 | 65.3 | 54.2 | 68.5 | 61.8 | 70.3 | 76.6 | **81.3** | 81.2 |
| Horse | 11.5 | 44.3 | 53.5 | 50.8 | 58.9 | 54.0 | 67.8 | 72.6 | 72.8 | **76.3** |
| Motorbike | 10.6 | 48.9 | 44.2 | 58.3 | 60.5 | 60.9 | 61.5 | 73.7 | 73.5 | **75.4** |
| Train | 19.6 | 39.2 | 29.6 | 62.4 | 65.2 | 66.3 | 78.2 | 76.3 | 75.7 | **78.1** |
| Average | 15.5 | 46.3 | 53.8 | 62.5 | 68.0 | 67.6 | 74.0 | 77.6 | 78.3 | **79.7** |

TABLE II

EVALUATION OF PER-CATEGORY PERFORMANCE (MEAN IOU) ON THE YOUTUBE-OBJECTS DATASET. THE BEST RESULT FOR EACH CATEGORY IS HIGHLIGHTED IN BOLD FONT. OUR APPROACH ACHIEVES THE BETTER RESULTS FOR ALMOST ALL CLASSES. WE ALSO ACHIEVE THE BEST AVERAGE MIOU.

| DAVIS | Attention Transfer | Fine-tuning | CRF | mIoU |
|---|---|---|---|---|
| ✓ | | | | 52.5 |
| ✓ | ✓ | | | 53.8 |
| ✓ | | ✓ | | 77.4 |
| ✓ | ✓ | ✓ | | 79.6 |
| ✓ | ✓ | ✓ | ✓ | 81.1 |

TABLE III

EFFECT OF VARIOUS COMPONENTS OF OUR APPROACH ON THE DAVIS VALIDATION SET. IF WE ONLY USE THE PARENT NETWORK TRAINED ON THE DAVIS TRAINING DATA (1ST ROW), THE PERFORMANCE IS ONLY 52.5%. BOTH ATTENTION TRANSFER AND FINE-TUNING ON THE FIRST FRAME OF A TEST VIDEO IMPROVE THE PERFORMANCE. FINALLY, CRF POSTPROCESSING FURTHER IMPROVES THE PERFORMANCE.

We follow the setup in [6] in our experiments. First, a modified version of FCN (called *Base Network* in [6]) is initialized using pretrained weights on ImageNet. We then use the 30 training videos in DAVIS to learn our model (called *Parent Network* in [6]). During testing, we are given an input test video and the ground-truth segmentation mask on the first frame. We fine-tune the parent network on the first frame to get a model (called *Test Network* in [6]) tuned to this test video. We then perform the attention transfer described in Sec. III-B and use the test network to predict the segmentation mask for each remaining frame of the test video.

We compare our approach with several state-of-the-art methods, including OSVOS [6], OFL [24], PLM [26], CTN [19], BVS [25], VPN [7], SFL [18] and MaskTrack [17]. The results are shown in Table I (2nd column). Figure 3 shows the qualitative examples of our method on the DAVIS dataset. Figure 4 shows the visualization of the transferred and non-transferred attention maps in the student stream.

**Youtube-Objects:** Following [6], we also evaluate on the Youtube-Objects [23] dataset. This dataset contains videos of 10 object classes. Compared to DAVIS, this dataset includes frames with less variations, motion and occlusions between foreground objects of consecutive frames. We follow the same

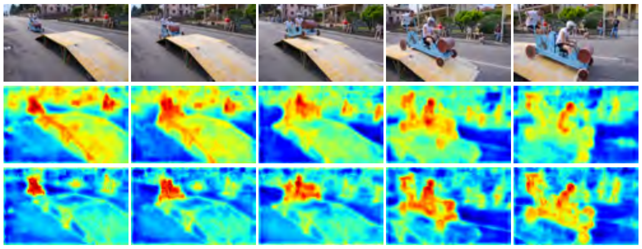

Fig. 4. Visualization of attention maps for different frame sequences in a video from the DAVIS validation set. (1st row) original images; (2nd row) attention maps with transferring the attention map from the teacher stream; (3rd row) attention maps obtained after attention transfer.

evaluation protocol of DAVIS for experimenting the method on YouTube-Objects. We compare with other state-of-the-art methods on this dataset. The results are shown in Table I (3rd column). We also show the per-class results on this dataset in Table II. Our approach significantly outperforms other state-of-the-art methods.

### C. Ablation Analysis

We perform ablation studies on the DAVIS dataset to further explore the relative contribution of each component in our approach by leaving out one or more components. Table III shows the results of this ablation analysis. If we only train the model on the DAVIS training data (i.e. only using the parent network), the performance (1st row in Table III) is only 52.5%. If the model trained on DAVIS is fine-tuned on only the first frame, it can achieve 77.4% [6]. Using fine-tuning on the first frame and the attention transfer, we are able to boost the performance to 79.6%. Using all components gives the best overall performance.

## V. Conclusion

We have presented a new approach for one-shot video object segmentation. Our model uses the Siamese network architecture with two streams. The first stream (teacher stream) is used to process the first frame in the video and its ground-truth segmentation mask. The second stream (student stream) is used to predict the segmentation mask of another frame in the video. By transferring the attention maps from the teacher stream to the student stream, our model effectively transfer the knowledge captured from the teacher stream to help the student stream to accurately segment the object. Our experimental results demonstrate that our approach outperforms other state-of-the-art methods.

## References

[1] Alon Faktor and Michal Irani, "Video segmentation by non-local consensus voting," in *British Machine Vision Conference*, 2014.

[2] Anestis Papazoglou and Vittorio Ferrari, "Fast object segmentation in unconstrained video," in *IEEE International Conference on Computer Vision*, 2013.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012.

[4] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015.

[5] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[6] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool, "One-shot video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[7] Varun Jampani, Raghudeep Gadde, and Peter V. Gehler, "Video propagation networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[8] Sergey Zagoruyko and Nikos Komadakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," in *International Conference on Learning Representations*, 2017.

[9] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu, "Recurrent models of visual attention," in *Advances in Neural Information Processing Systems*, 2014.

[10] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*, 2015.

[11] Yu-Xiong Wang and Martial Hebert, "Learning to learn: Model regression networks for easy small sample learning," in *European Conference on Computer Vision*, 2016.

[12] Luca Bertinetto, Joao F. Henriques, Jack Valmadre, Philip H. S. Torr, and Andrea Vedaldi, "Learning feed-forward one-shot learners," in *Advances in Neural Information Processing Systems*, 2016.

[13] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, et al., "Matching networks for one shot learning," in *Advances in Neural Information Processing Systems*, 2016.

[14] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap, "One-shot learning with memory-augmented neural networks," in *Advances in Neural Information Processing Systems*, 2016.

[15] Paul Voigtlaender and Bastian Leibe, "Online adaptation of convolutional neural networks for video object segmentation," in *British Machine Vision Conference*, 2017.

[16] S Avinash Ramakanth and R Venkatesh Babu, "Seamseg: Video object segmentation using patch seams," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[17] Anna Khoreva, Federico Perazzi, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung, "Learning video object segmentation from static images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[18] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang, "Segflow: Joint learning for video object segmentation and optical flow," in *IEEE International Conference on Computer Vision*, 2017.

[19] Won-Dong Jang and Chang-Su Kim, "Online video object segmentation via convolutional trident network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[20] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[21] Philipp Krähenbühl and Vladlen Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in Neural Information Processing Systems*, 2011.

[22] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[23] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari, "Learning object class detectors from weakly annotated videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[24] Yi-Hsuan Tsai, Ming-Hsuan Yang, and Michael J Black, "Video segmentation via object flow," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[25] Nicolas Märki, Federico Perazzi, Oliver Wang, and Alexander Sorkine-Hornung, "Bilateral space video segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[26] Jae Shin Yoon, Francois Rameau, Junsik Kim, Seokju Lee, Seunghak Shin, and In So Kweon, "Pixel-level matching for video object segmentation using convolutional neural networks," in *IEEE International Conference on Computer Vision*, 2017.