Applying SlowFast Networks to Video Object Segmentation

Chantal Pellegrini
Technical University of Munich
chantal.pellegrini@tum.de

Ege Özsoy
Technical University of Munich
ege.oezsoy@tum.de

Abstract

1. Introduction

VOS is a computationally expansive task, especially requiring a lot of GPU memory to look into many frames at once. SlowFast architecture, which was successfully applied for video action classification, splits the workload between two branches, allowing it to use less memory. Our main goal is to apply SlowFast Networks to Video Object Segmentation and to understand if this concept is beneficial for VOS; both in the unsupervised and semi-supervised settings. To this end, we build a SlowFast inspired VOS architecture, and evaluate different configurations on it.

2. Related Work

The main work of consideration is the SlowFast Network [2], which introduces the idea of using two pathways to analyze a video. This architecture has one fast and one slow pathway, allowing it to concentrate on different aspects of the video while keeping the performance high. The slow pathway is computationally much more expensive than the fast pathway and works with less fps. In the original task of video action classification they show this two pathway approach is especially beneficial if there is a high speed action such as clapping or dancing involved.

OSVOS [1] focuses on the task of one-shot video segmentation, where in test time, the network is fine-tuned on the first frame of the video, which has a manually annotated mask. This fine-tuning allows the network to adapt to the object in that scene.

Mask R-CNN [3] is one of the best known architectures for image segmentation. They extend the popular faster R-CNN architecture with a masking layer at the end.

3. Approach

In the following we present the details of our approach.

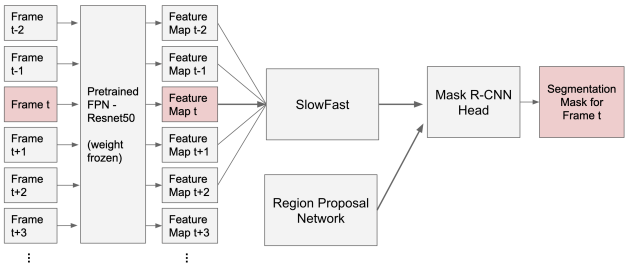

Figure 1: Architecture Overview.

3.1. Dataset

We used both of the DAVIS datasets [4, 5]. These datasets include videos with segmentation ground truths for every frame. We trained on DAVIS17, as it contains more video sequences, but evaluated on DAVIS16, which is a subset of DAVIS17, since there only exists one annotation per frame.

3.2. Network Design

Our architecture mainly uses MaskRCNN, which we extended with a SlowFast module between the backbone and the head. An overview is shown in Fig. 1. The backbone is a ResNet50 Feature Pyramid Network, which was pretrained on Coco and fine-tuned on DAVIS17 [5].

The computed feature maps of several frames are fed into the SlowFast module, which can be seen in Fig. 2. It consists of two pathways, both with three 3D convolutions, followed by batch norm and for the first two layers a ReLu activation. After the first two layers the outputs of the fast pathway are fused into the slow pathway using another combination of Convolution, Batchnorm and ReLu.

The final outputs of the slow and the fast pathway are then concatenated and used as input for the Mask R-CNN head. Additionally the head also receives region proposals computed by a RPN, which works with the original image features computed by the backbone.

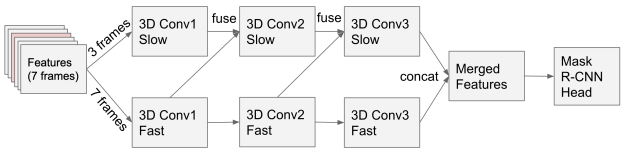

Figure 2: Overview of SlowFast Layers.

3.3. Training

For the unsupervised case we are training for 20 epochs on the training data of DAVIS17 [5]. We are using SGD with momentum as optimizer and our learning rate is set to 0.001.

The semi-supervised training starts with a parent model trained on the task of unsupervised VOS and finetunes this model on specific sequences, resulting in one model per sequence. We are using different augmentations, including Random Horizontal Flip, Rotation of up to 30 degrees, and Scaling of the image. We experimented with different scaling strengths and learning rates, that are described in the experiment section.

4. Experiments

As our network has can be applied in many different configurations, both in terms of pathway size but also in terms of freezing certain layers etc, we conducted several experiments. The qualitative results can be found here TODO

4.1. Hyperparameter Tuning

As our resources were limited, we relied on previous works for some decisions such as which optimizer to use. Nonetheless, for the unsupervised case, we experimented with 3 different learning rates and also freezing the backbone or not. In the end we settled on 0.001 as lr and freezing the backbone, as training it didn't provide much improvement but came at a big cost of speed. For the semi-supervised case we actually ran in total 36 different experiments for every configuration, which tested different amount of data augmentations, learning rate and freezing different parts of the network. For every pathway configuration except for pathway size 1-1, the best performing configuration was only freezing the SlowFast part of the network, a random scale of 0.25 and learning rate of 0.001. For 1-1, everything is the same except random scale which was 0.4 for the best configuration. We used the best performing configurations in all the following experiments.

4.2. Pathway Configurations

The main goal of the pathway configurations are two fold, first two show that using more temporal context provides a benefit, and second to show the benefit of SlowFast. For both of these, we created a baseline configuration, 1-1, which doesn't use any temporal context. We then created two additional architectures 3-3 and 7-7 which progressively use more temporal context, but also work get slower with the increased capacity. It is important to note that these 3 configurations are not using a SlowFast inspired architecture, as both of the pathways have the same size. In addition to these 3, we tested 1-7 and 3-7 configurations, both utilizing the concept of SlowFast.

The first three experiments are to show the benefit of seeing more temporal context. The last two are to see the benefit of SlowFast.

Configuration	J & F Mean	Param. Count	Eval. Time
1-1	0.645	45,421,851	477 sec
3-3	0.679	46,398,747	544 sec
7-7	0.673	48,407,835	853 sec
1-7	0.655	45,618,459	528 sec
3-7	0.676	46,570,779	584 sec

Table 1: Unsupervised VOS results on DAVIS16 validation set. The Evaluation Time refers to computation of masks for all validation sequences.

Configuration	J & F Mean
1-1	0.747
3-3	0.747
7-7	0.755
1-7	0.741
3-7	0.758

Table 2: Semi-supervised VOS results on DAVIS16 validation set.

4.3. Results

Table 1 and 2 show the results for the unsupervised, respectively semi-supervised experiments. As evaluation metric we use J&F Mean, like described in the DAVIS16[4] paper.

5. Evaluation

Our first observation is that in our case temporal context improves the unsupervised segmentation results, but only up to three seen frames, as can be seen by comparing the results of 1-1, 3-3 and 7-7. As SlowFast networks are mainly designed to see a lot of temporal context in the fast pathway and large temporal context does not seem to be beneficial, the potential benefit is small. When we compare 1-1 with 1-7 and 3-3 with 3-7 we see that an improvement through a larger fast pathway size is only visible for a slow pathway size of one.

In the semi-supervised experiments we only very little temporal benefit between 3-3 and 7-7 and no between 1-1 and 3-3. We also see a small improvement between 3-3 and

3-7 but a small regression from 1-1 to 1-7. Overall the small differences indicate neither temporal context nor SlowFast have significant influence on the OSVOS results.

Performance wise one can see the parameter count increases marginally with bigger pathway sizes, but the evaluation time increases significantly. The slow pathway has significantly more influence on the time, like we see when comparing e.g. evaluation time of 3-3, 3-7 and 7-7.

Overall, mainly as the benefit of temporal context vanishes quickly, the benefit of SlowFast for our tasks appears to be limited.

6. Conclusion

EGE - Benefit of SlowFast

References

[1] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. *CoRR*, abs/1611.05198, 2016.

[2] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. *CoRR*, abs/1812.03982, 2018.

[3] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017.

[4] F. Perazzi, 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 *Computer Vision and Pattern Recognition*, 2016.

[5] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017.