U-Net Based Multi-instance Video Object Segmentation

Heguang Liu, Jingle Jiang
heguangl@stanford.edu, jxj142@stanford.edu
Stanford University

Overview

Multi-instance video object segmentation is to segment specific instances throughout a video sequence in pixel level, given only an annotated first frame.
In this paper, we implement an effective fully convolutional networks with U-Net similar structure built on top of OSVOS fine-tuned layer. We use instance isolation to transform this multi-instance segmentation problem into binary labeling problem, and use weighted cross entropy loss and dice coefficient loss as our loss function. Our best model achieves **F mean: 0.467** and **J mean: 0.424** on DAVIS dataset, which is a comparable performance with the State-of-the-Art approach. But case analysis shows **this model can achieve a smoother contour and better instance coverage, so it's better for recall focus segmentation scenario**.
We also did many experiments on other convolutional neural networks, including SegNet, Mask R-CNN, and provide insightful comparison and discussion.

Training Data and Setup

- Training Data
  DAVIS Dataset DAVIS(Densely Annotated VIdeo Segmentation). There are total 120 video sequence (8279 images), in which train: 60, val:30, test: 30.
- Evaluation Metrics
  Region Similarity The intersection-over union between the mask and ground-truth.
  Contour Accuracy The Harmonic mean of contour's precision and recall

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

$$F = \frac{P_c * R_c}{P_c + R_c}$$

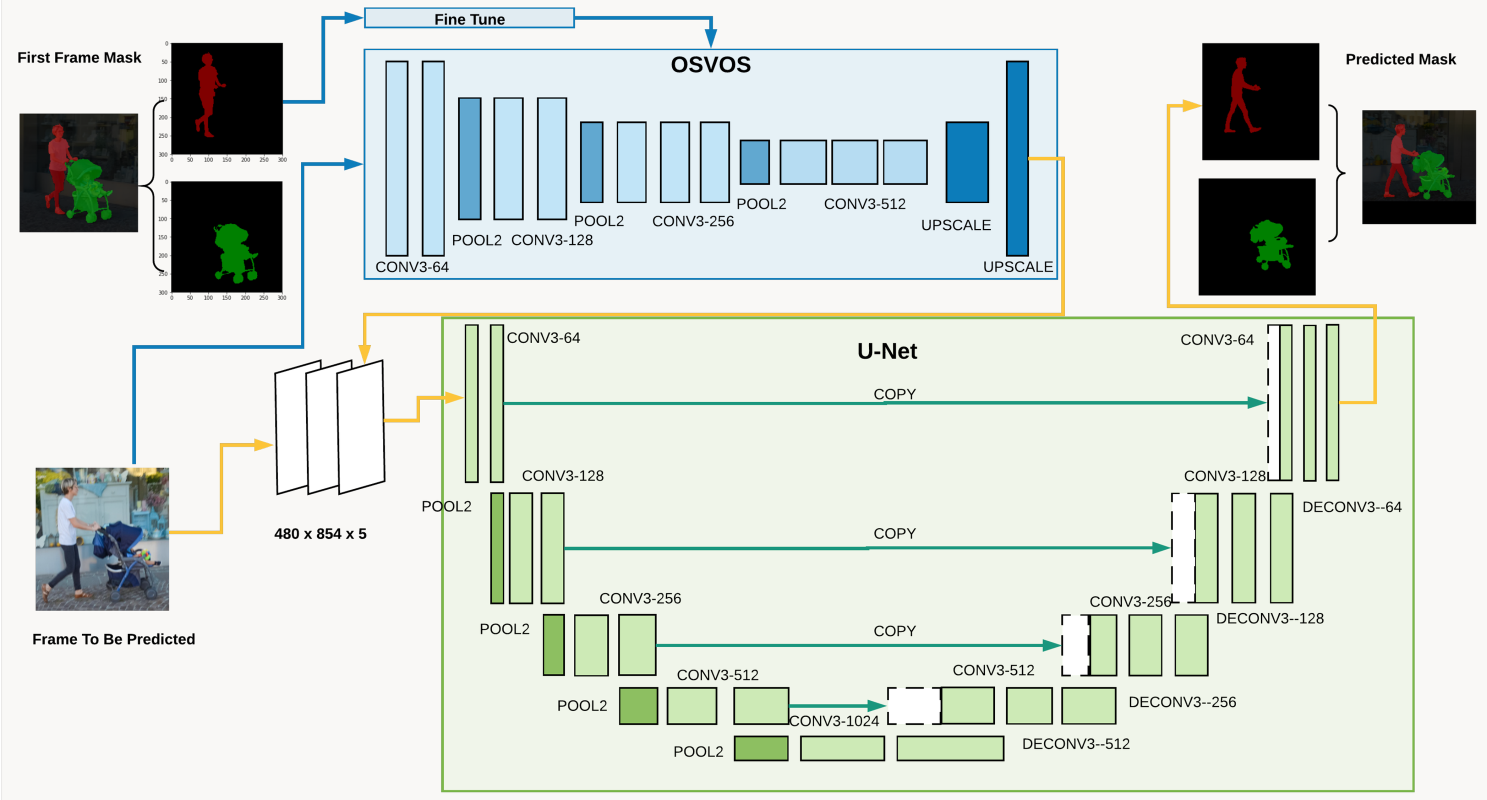
- Training Setup
  Implementation The model was implemented using tensorflow 1.8 framework and Python 3.6. We've written 3000+ lines of code.
  GPU Our model was trained on 5 N1-HighMem-8 instances on Google Cloud Compute Platform with NVIDIA Tesla P100 GPU attached.
- Parameter Tuning

- The best U-Net Model has convolutional filter number of 64, 128, 256, 512, contains 31M trainable parameters, with learning rate 4e-5 and batch size 8.
- The periodic up and downs in training loss is because we were not able to do shuffling on training dataset, due to GPU memory limitation.

U-Net Filter	J	F	Params	Lr	Batch
16,32,64	0.314	0.163	700K	4e-5	20
32,64,128	0.345	0.325	1M	4e-5	20
64,128,256,512	0.419	0.430	31M	2e-5	8
64,128,256,512	0.424	0.467	31M	4e-5	8

(a) Train CE loss, lr =4e-5
(b) Train CE loss, lr =2e-5
(c) Test CE loss, lr = 4e-5
(d) Test CE loss, lr = 2e-5

Architecture



- Instance Isolation
- OSVOS
- U-Net based Fully Convolutional Networks
  Contracting path: a series of convolutional layer and max pooling layer to capture enough context. Expanding Path: up-sampling layer to increase the output resolution and crop and merge the high resolution feature from the contracting path with these up-sampled output.
- Loss Function
  - Weighted Cross Entropy Loss
  - Dice Coefficient Loss

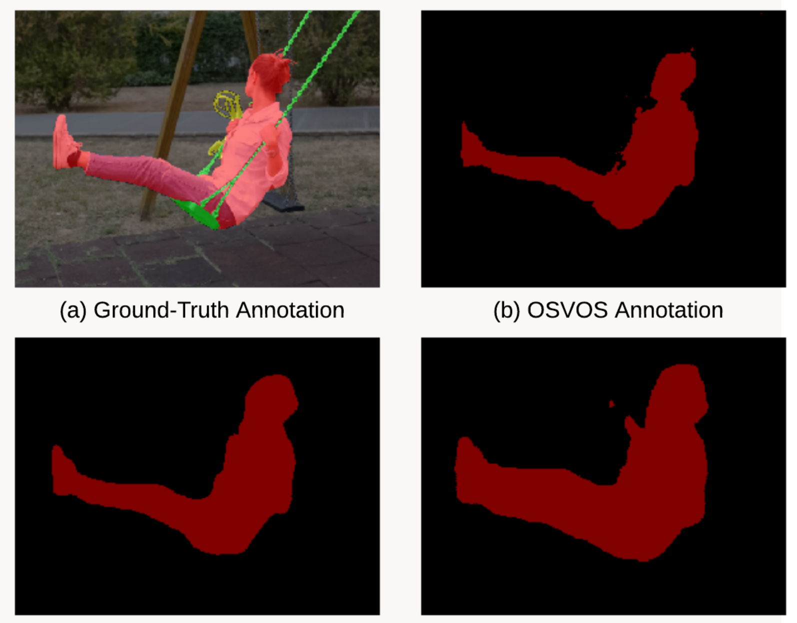
$$L = -\sum_x \omega(x)\, p(x) \log q(x)$$

$$L = 1 - \frac{2|X \cap Y|}{|X| + |Y|}$$

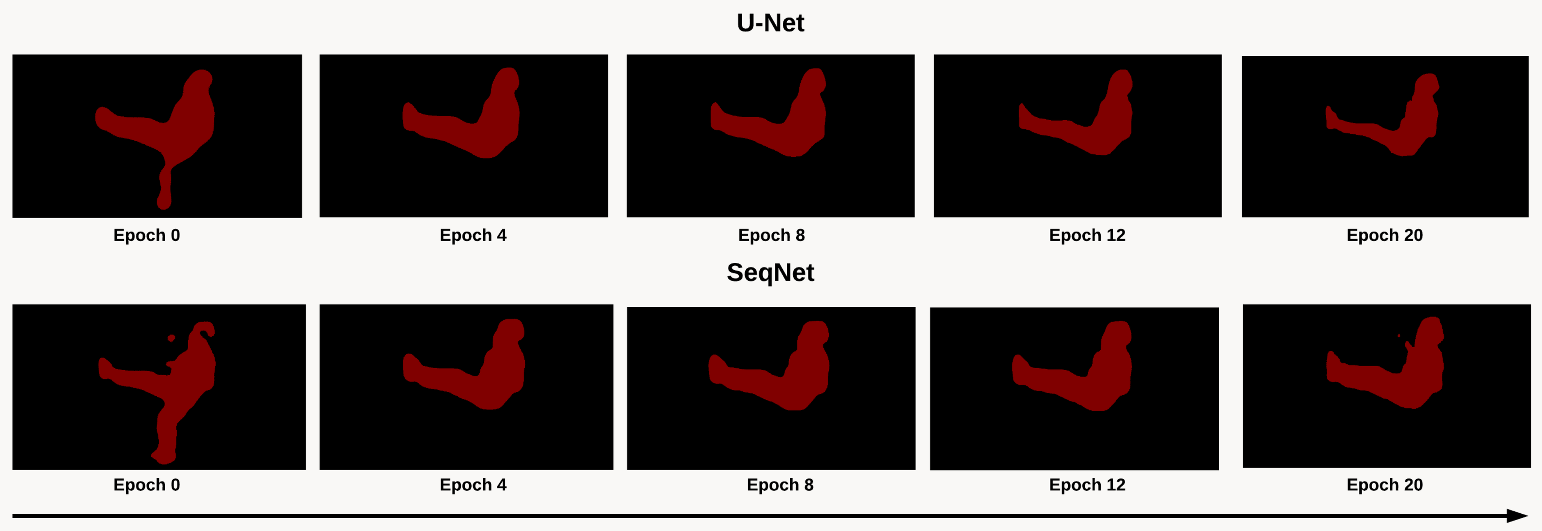
Models Comparison

- U-Net has a J-mean comparable with the State-of-the-Art, and a slightly worse F-mean
- U-Net produce a more complete instance with a better coverage and smoother contour in exchange of contour accuracy
- SegNet has a much coarse contour with lower precision

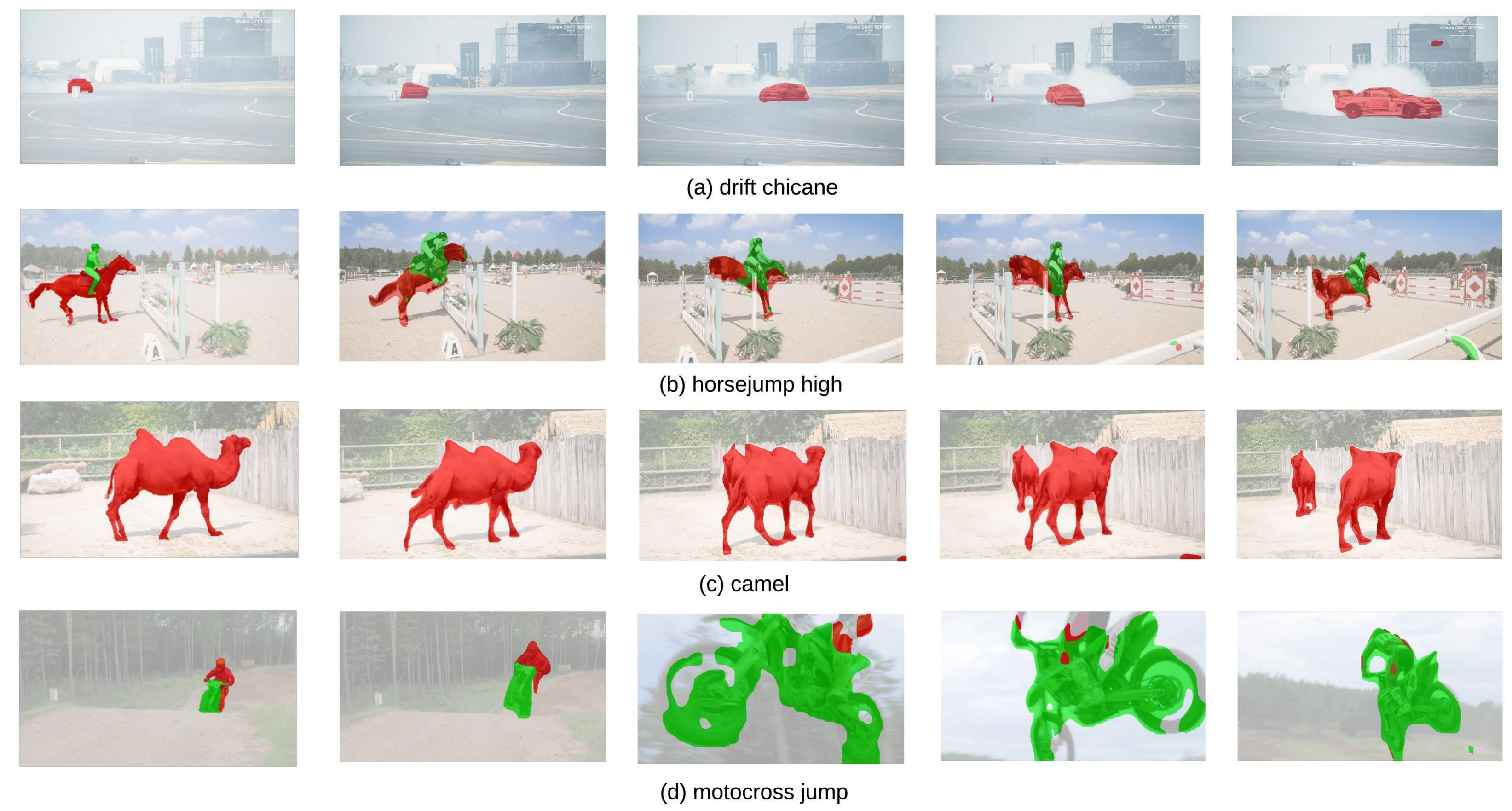
Model	J	F	Dice Loss
OSVOS	0.499	0.592	-
SegNet	0.347	0.214	0.407
U-Net	0.424	0.467	0.289



(a) Ground-Truth Annotation
(b) OSVOS Annotation
(c) U-Net Annotation
(d) SegNet Annotation



Results and Discussion



(a) drift chicane
(b) horsejump high
(c) camel
(d) motocross jump

- The model can handle intensive motion and sharp appearance change gracefully.
- The model can handle multi-instances with similar motion very well, even with overlapping.
- The model doesn't handle multiple object collusion very well.
- The model lost track when object goes beyond image boundary and goes back.

Other Failed Experiments

- Mask R-CNN
  Doesn't handle unseen instance
  Doesn't incorporate with the first frame
- Unweighted Cross Entropy as Loss Function
  Produce all background image
- Direct Feed Multi-instance Image
  Can't project to different layers

Conclusion and Future work

We implement and compare a number of fully convolutional networks to tackle the multi-instances video object segmentation problem. Among SegNet, U-Net, Mask R-CNN, U-Net based architecture achieves the best result with **F mean: 0.467** and **J mean: 0.424**. This result is comparable to the current State-of-the-Art approach on DAVIS Dataset.
From the case study, we noticed this model doesn't perform well on 2 cases: 1). Multi-instance occlusion 2). Instance lost tracking after it goes out of image boundary.
In the future, we propose experiment the following two approaches to solve these problems:

- Recurrent Neural network to better tracking each object by its temporary continuity to handle occlusion.
- Adaptive object re-identification to prevent target lost.