Multiple-Instance Video Segmentation with Sequence-Specific Object Proposals

Amirreza Shaban*, Alrik Firl+, Ahmad Humayun*, Jialin Yuan+,Xinyao Wang+,
Peng Lei+, Nikhil Dhanda*, Byron Boots*, James M. Rehg*, Fuxin Li+

* Georgia Institute of Technology
{amirreza,ahmadh,nnn3,rehg}@gatech.edu, bboots@cc.gatech.edu
+ Oregon State University
{firla,yuanjial,wangxiny,leip}@oregonstate.edu, lif@engr.orst.edu

Abstract

We present a novel approach to video segmentation which won the 4th place in DAVIS challenge 2017. The method has two main components: in the first part we extract video object proposals from each frame. We develop a new algorithm based on one-shot video segmentation (OSVOS) algorithm to generate sequence-specific proposals that match to the human-annotated proposals in the first frame. This set is populated by the proposals from fully convolutional instance-aware image segmentation algorithm (FCIS). Then, we use the segment proposal tracking (SPT) algorithm to track object proposals in time and generate the spatio-temporal video object proposals. This approach learns video segments by bootstrapping them from temporally consistent object proposals, which can start from any frame. We extend this approach with a semi-Markov motion model to provide appearance motion multi-target inference, backtracking a segment started from frame T to the 1st frame, and a "re-tracking" capability that learns a better object appearance model after inference has been done. With a dense CRF refinement method, this model achieved 61.5% *overall accuracy in DAVIS challenge 2017.*

1. Introduction

Our GaTech-Oregon State team reaches 61.5% overall mean accuracy in the DAVIS2017 challenge. Our pipeline consists of 3 parts: 1) Proposal Generation 2) SPT Tracking of the Proposals 3) Spatial Refinement. Separating proposal extraction from tracking allows us to use different algorithms to generate set of proposals that each has a high recall on part of the DAVIS dataset. For each sequence we first extract segment proposals using two approaches, a novel approach extending OSVOS [1] with LucidDream [3] augmentation method to generate proposals that match the human-annotated first frame segment in the video, and running the FCIS [6] instance segmentation algorithm to generate proposals for known semantic classes. These proposals are treated as segment proposals in each image, and then a novel enhanced version of the multi-segment tracking and object discovery algorithm SPT [5, 10] is used for tracking the objects and selecting which ones to match to which object annotation in the 1st frame. This version of SPT finds objects that start from any frame and lasts for any duration (a minimal 7 frames is required in the final submission), learns a long-term appearance model of them based on Color-SIFT, and also handles partial and complete occlusions and finds objects that re-enter the scene. After SPT, all the found object tracks are backtracked to the 1st frame and matched with the ground truth annotation in the 1st frame. After the matching, SPT is used again to learn a long-term appearance model of the consolidated tracks that match the ground truth, which improved performance. Finally, a fully-connected CRF for spatial refinement is performed with unaries coming from SPT. There are a number of novelties in the approach:

- A novel approach extending OSVOS and LucidDream in generating object proposals that match a ground truth object.
- Incorporating a semi-Markov pixel-level motion model in SPT-Occlusion.
- Backtracking all the SPT segment tracks to the 1st frame and re-tracking them on the whole sequence.
- Fully-connected CRF on SPT unaries.

We review each part of the pipeline in the next sections.

2. Object Proposal Generation

We used a modified version of OSVOS [1] to generate sequence specific proposals and FCIS [6] to generate pro-

posals from seen semantic classes. We evaluate the importance of each algorithm by reporting per-frame Jaccard (IoU) measure on the validation set. Since there are many proposals for each object, we choose the proposal that maximize per-frame Jaccard region similarity measure [8].

2.1. Sequence-Specific Proposals

We use one-shot video segmentation method [1] to generate sequence specific proposals. Here we discuss all the modifications we do to the original algorithm to generate proposals for our algorithm. Note that, we use the publicly available code[1] for OSVOS algorithm. Our method has three parts: offline training in which we train the parent network, the online training/testing which generates one proposal per frame per instance, and the last combinatorial step which generates multiple hypothesis by exploring the combinatorial space of connected components.

Offline Training: The parent model is trained on the DAVIS2017 training dataset with the same parameter set as suggested in the paper. For a specific sequence, let I_t and M_{it} be the image and the corresponding binary mask for instance i in time t. At each iteration we pick a random instance from the dataset and do a gradient decent update. We use a modified version of balanced loss function in our model:

$$L = \alpha \sum \log p^{+} + \beta \sum \log \left(1 - p^{-}\right) \qquad (1)$$

where $\alpha = |M_{it}|_{-}/|M_{it}|$ and $\beta = \min(|M_{it}|_{+}/|M_{it}|, 0.1)$. Setting the minimum value of 0.1 is specially crucial in the online-training step for getting acceptable performance on the small objects (see Figure 1).

Online training/testing To generate proposals for each instance j in the test set, we fine-tune the parent network on the first frame image/mask (I_0, M_{j0}) and test the network on the rest of the frames which gives us one proposal per frame per object. To increase the generalization performance of the network we augment the training set (I_0, M_{j0}) using the method in [3]. Even with augmenting the data, OSVOS proposals classifies parts of different similar instances as foreground (Figure 2 first column). In the worst case scenarios, the predictions drifts gradually from one instance to another instance in the scene. More importantly, this smooth drifting can tricks our segment tracking algorithm which allows smooth temporal changes in the proposals' appearances. To alleviate this problem we use a simple and effective combinatorial grouping algorithm to generate many proposals from each prediction. This method increases the Jaccard performance by 3%.

Combinatorial Grouping Given the binary prediction mask $\hat{M}_{jt}$, we first find the spatially connected components in the mask with the area > 20 pixels. Then we generate all the possible combinations of the remaining CCs (sorted from the one with fewer number of CCs and highest area). Then, we reject any combination in which the distance between two CCs is more than 50 pixels. See Figure 2 for the qualitative performance of this algorithm. The overall accuracy for the generated proposals is around 67% on the validation set.

2.2. Semantic Proposals

Recently, instance-aware image segmentation algorithms [6, 2, 9] have showed exceptional performance in segmenting objects from known semantic classes. To cover these semantic classes better, we enriched the set of proposals by the proposals generated from FCIS [6] algorithm. We use the publicly available FCIS implementation [2]. Since our proposals are class agnostic, we only use the mask predictions and discard the semantic labels. To maximize the proposal recall we accept proposals with the confidence prediction higher than 0.1. We realize that decreasing the confidence threshold from 0.5 to 0.1 helps to generalize to new objects and increases the IoU by 5%. FCIS proposal alone shows 72% performance. Combined with sequence-specific proposals we got 78% performance. We also tried using SharpMask [9] and COB [7] proposals but got inferior performance compared to FCIS method.

3. SPT Tracking of the Proposals

Our tracking algorithm is unique in that it does not specifically learn from the ground truth annotation in the first frame. Instead, it is still mostly a general unsupervised video segmentation algorithm that is only **tested** on the ground truth annotation in the first frame and use that to help reduce the number of tracks to be refined. Hence, it does not depend on precise annotation in the first frame, nor does it depend on the ground truth object must be present in the first frame, etc. Such flexibility could turn out to be a benefit in real-life scenarios.

We used a version of segment proposal tracking (SPT) algorithm[5] similar to the one in [10], with the following modifications:

- Removed the backtracking in [10] to every 5-th frame.
- Added a backtracking towards the 1-st frame after track consolidation.
- Selection of tracks that correspond to the ground truth objects.
- Refined inference with a pixel-level semi-Markov motion model during inference.

[1] http://www.vision.ee.ethz.ch/~cvlsegmentation/osvos/

[2] https://github.com/msracver/FCIS

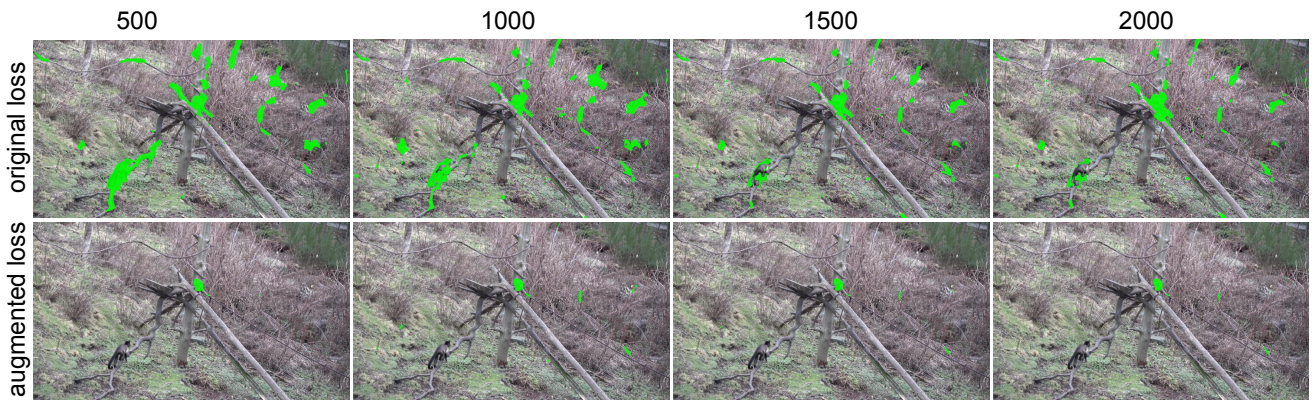

Figure 1. **Effect of the Loss Function** While the original loss which is used in OSVOS algorithm does not converge after 2000 iterations for the tiny object in the monkeys-trees sequence (**first row**), the augmented loss in Equation 1 converges after 500 iterations (**second row**). Columns show the prediction after different number of iterations.

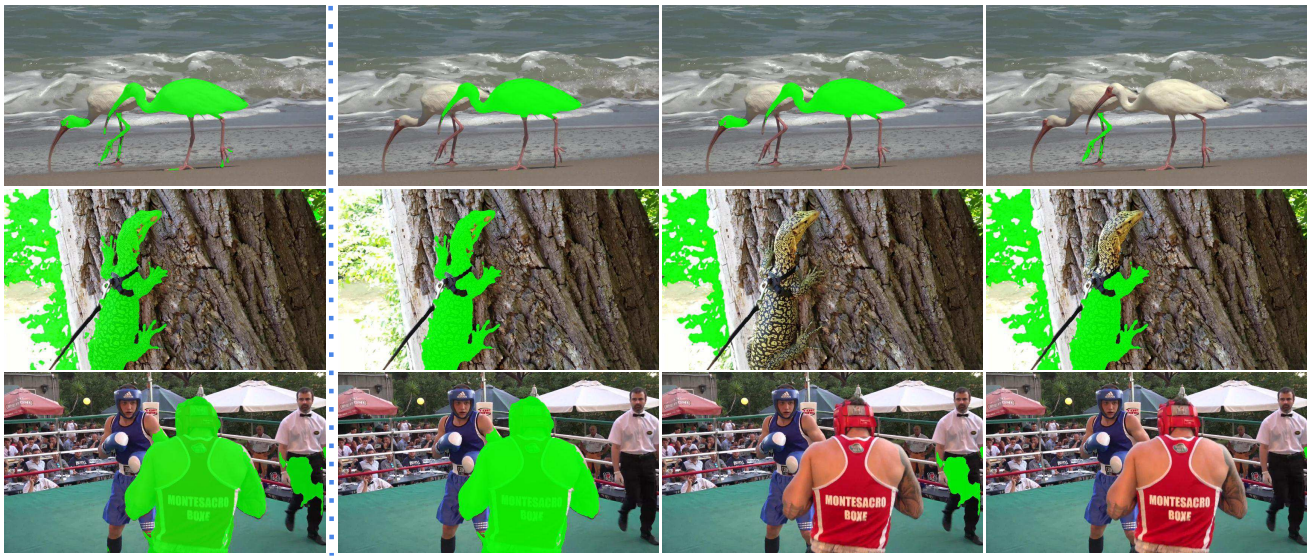

Figure 2. **Combinatorial Grouping** First column shows OSVOS prediction. Second column shows the best proposal generated from OSVOS prediction. Other columns show samples from generated proposals. The grouping algorithm increases recall by taking into account the prior on the continuity of the parts, the maximum distance between each two parts, and the area of each part.

- Added re-tracking after multi-object inference.

These are discussed in detail in the following paragraphs.

Remove backtracking to every 5-th frame SPT depends on recursive least squares on many proposals to compute regression models based on appearance. In SPT, regressions against multiple targets from the same set of proposals are sped-up with least square objects (LSOs), which encapsulate the sample covariance matrix as well as the products from inputs and the targets. Regression can be directly performed from LSOs without any additional information [5, 10]. In [10], it was proposed to merge several LSOs together in order to improve computational speed. During the challenge, it was discovered that due to the proposal weighting mechanisms in the code of SPT, such merging would make regression scores (predicted overlaps) learnt from different proposals incomparable, hence we removed it and instead directly ran each LSO to the end of the sequence, as in [5].

Backtracking towards the 1-st frame In the DAVIS challenge, the ground truth was presented in the 1-st frame. However in many cases, the 1-st frame is not very easy to track and SPT picks up segments from latter frames which did not include the 1-st frame. SPT would result in many different tracks starting and ending at different frames, which are consolidated by only retaining tracks that exceed a certain length (7 frames in the DAVIS challenge).

After consolidation, for all the tracks that start later than the 1-st frame, we ran the SPT algorithm in reverse order to backtrack to the 1-st frame. The LSOs in those cases are initialized to be the final LSOs in the original forward-tracking SPT, then updated by including all the proposals in all the intermediate frames as training examples in the same manner as the original SPT [5]. The only difference is that no new tracks are created or removed during this process.

Selection of tracks that correspond to the ground truth objects For each tracked segment track, SPT will generate an appearance model that predicts overlaps from each object to this track. We use the predicted overlap on the ground truth object in the first frame as the score of the track w.r.t. the ground truth. Hence, there might be multiple tracks that correspond to the same ground truth object. This is resolved in the spatial refinement step. We threshold to only retain tracks that have predicted overlaps at least 70% of the maximal predicted overlap of each ground truth objects (e.g., if 3 tracks predicted the same ground truth object to have overlap of 0.7, 0.63 and 0.44, the third one is not selected). This threshold is fairly arbitrary and may not be required.

Refined Inference with a pixel-level semi-Markov motion model It is not possible to incorporate a strong motion model in SPT due to the algorithm simultaneously tracking thousands of objects. However, once we consolidated to a few tracks that correspond to each ground truth object, a stronger motion model can be used to adjust the score of each segment. In this work, we utilize a semi-Markov motion model with constant velocity. The motion model is defined as follows:

$$\begin{aligned} M_k(p_i) &= \frac{\sum_{j=1}^{10} w_j S_{k-j,j}(p_i)}{\sum_{p_i}\sum_{j=1}^{10} w_j S_{k-j,j}(p_i)} \\ S_{k-j,j} &= G_\sigma * T_{\mathbf{v}_{k-j}}(S_{k-j,j-1}) \\ S_{k-j,0} &= S_{k-j} \\ v_k &= 0.7 * v_{k-1} + 0.3(c(S_k) - c(S_{k-1})) \end{aligned} \quad (2)$$

where S_k is the segment of the track at frame k and $S_{k,j}$ is the estimated location of S_k after j frames of motion. $T_{\mathbf{v}_k}$ denote a translation operator with $\mathbf{v}_k$ being the amount of translation (velocity), G_σ is a Gaussian blur with parameter σ. In each frame, $S_{k,j}$ is updated by applying the velocity vector $\mathbf{v}_k$ to the segment mask first, and then applying a Gaussian blur with parameter σ. The velocity vector is a 2-dimensional vector computed by the difference between the centroid of segments S_k and S_{k-1}, denoted as $c(S_k)$ and $c(S_{k-1})$. The velocity is updated with a momentum factor of 0.7, which is not applied in the first 5 frames.

This motion model takes into account the motion of the object in the past 10 frames, with further ago frames being blurred more and having smaller weights. This is to account for the fact that the tracking may be noisy and in some frames the results may be completely wrong. Past frames are always moved with a linear motion model v_{k-j} which is not updated. $M_k(p_i)$ is normalized to a distribution.

After computing $M_k(p_i)$, the motion score of each proposal in frame k is computed as the average log-likelihood of all the pixels in the segment S_k:

$$m(S_k) = \frac{\sum_{p_i \in S_k} log(M_k(p_i))}{|S_k|} \quad (3)$$

where $|S_k|$ denotes the number of pixels in S_k.

Then, $m(S_k)$ is considered in additon to $o_w(S_k)$, the predicted overlap of S_k from the appearance model w, and the segment proposal that maximizes the final score $o_w(S_k) + \alpha_m m(S_k)$ is selected as the segment representing the track with appearance model w. This is used instead of the original SPT model where the segment that maximizes $o_w(S_k)$ is selected as representing the track.

Re-tracking Because the motion model changes the segments that are selected in tracking, the appearance model would no longer be as accurate as before. Therefore, it makes sense to re-train the appearance model with the improved motion scores in mind. This is implemented in the system. During the re-tracking step, the SPT tracker utilizes the motion model defined in the previous subsection and re-train segment track appearance models $\mathbf{w}$ from scratch. In this round, the chosen segments for the track and their corresponding appearance scores are stored. At each frame, the tracker determines which segment proposal to be chosen as the ground truth based on the predicted overlap plus the motion score. Then, the score of the highest-scoring proposal is compared against the stored scores from last round, plus a new motion score computed by the current segment track. If a segment proposal has a higher score than the stored segment from last round, then such a proposal is chosen as the segment for the track, otherwise, the stored segment is chosen as the segment for the track. Such procedure is ran until the end of the sequence. In certain sequences, this procedure significantly improves the appearance model and led to an improved tracking performance.

4. Spatial Refinement

As we mentioned earlier, the SPT tracking algorithm provides pixel-level confidence map for each instance in the scene. We use this values as the unary potential in a fully connected CRF [4] to enhance the predicted segments. We use the same binary potentials and optimization method as described in the paper.

5. Experiments

Quantitative results on DAVIS2017 dataset are available in the leaderboard[3] under name "Haamo". We include some qualitative results on the challenging sequences in Figure 3.

References

[1] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. One-shot video object segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[2] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017.

[3] A. Khoreva, R. Benenson, E. Ilg, T. Brox, and B. Schiele. Lucid data dreaming for object tracking. In *arXiv preprint arXiv: 1703.09554*, 2017.

[4] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[5] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg. Video segmentation by tracking many figure-ground segments. 2013.

[6] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutional instance-aware semantic segmentation. *arXiv preprint arXiv:1611.07709*, 2016.

[7] K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. V. Gool. Convolutional oriented boundaries: From image segmentation to high-level tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017.

[8] 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.

[9] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *European Conference on Computer Vision*, pages 75–91. Springer, 2016.

[10] Z. Wu, F. Li, R. Sukthankar, and J. M. Rehg. Robust video segment proposals with painless occlusion handling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4194–4203, 2015.

[3] `http://davischallenge.org/challenge2017/leaderboard.html`

Figure 3. **Qualitative Results** Qualitative results from challenging sequences in test-challenge set. Images are sample every $\approx$ 20 frames.