FusionSeg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos

Suyog Dutt Jain* Bo Xiong* Kristen Grauman
University of Texas at Austin
suyog@cs.utexas.edu, bxiong@cs.utexas.edu, grauman@cs.utexas.edu
http://vision.cs.utexas.edu/projects/fusionseg/

Abstract

We propose an end-to-end learning framework for segmenting generic objects in videos. Our method learns to combine appearance and motion information to produce pixel level segmentation masks for all prominent objects. We formulate the task as a structured prediction problem and design a two-stream fully convolutional neural network which fuses together motion and appearance in a unified framework. Since large-scale video datasets with pixel level segmentations are lacking, we show how to bootstrap weakly annotated videos together with existing image recognition datasets for training. Through experiments on three challenging video segmentation benchmarks, our method substantially improves the state-of-the-art results for segmenting generic (unseen) objects. Code and pre-trained models are available on the project website.

1. Introduction

In video object segmentation, the task is to separate out foreground objects from the background across all frames. This entails computing dense pixel level masks for foreground objects, regardless of the object's category—i.e., learned object-specific models must *not* be assumed. A resulting foreground object segment is a spatio-temporal tube delineating object boundaries in both space and time. This fundamental problem has a variety of applications, including high level vision tasks such as activity and object recognition, as well as graphics areas such as post production video editing and rotoscoping.

In recent years, video object segmentation has received significant attention, with great progress on fully automatic algorithms [1, 2, 3, 4, 5, 6, 7, 8, 9], propagation methods [10, 11, 12, 13, 14, 15], and interactive methods [16, 17, 18, 19]. We are interested in the fully automated setup, where the system processes the video directly without any human involvement. Forgoing manual annotations could scale up the processing of video data, yet it remains a very challenging problem. Automatic algorithms not only need to produce accurate space-time boundaries for any generic object but also need to handle challenges like occlusions, shape changes, and camera motion.

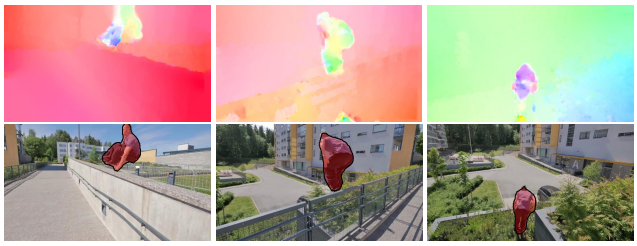

Figure 1: We show color-coded optical flow images (first row) and video segmentation results (second row) produced by our joint model. Our proposed end-to-end trainable model simultaneously draws on the respective strengths of generic object appearance and motion in a unified framework.

While appearance alone drives segmentation in images, videos provide a rich and complementary source of information in form of object motion. It is natural to expect that both appearance and motion should play a key role in successfully segmenting objects in videos. However, existing methods fall short of bringing these complementary sources of information together in a unified manner.

In particular, today motion is employed for video segmentation in two main ways. On the one hand, the propagation or interactive techniques *strongly rely on appearance* information stemming from human-drawn outlines on frames in the video. Here motion is primarily used to either propagate information or enforce temporal consistency in the resulting segmentation [13, 14, 15, 20]. On the other hand, fully automatic methods *strongly rely on motion* to seed the segmentation process by locating possible moving objects. Once a moving object is detected, appearance is primarily used to track it across frames [4, 6, 8, 9]. Such methods can fail if the object(s) are static or when there is significant camera motion. In either paradigm, results suffer because the two essential cues are treated only in a sequen-

*Both authors contributed equally to this work

tial or disconnected way.

We propose an end-to-end trainable model that draws on the respective strengths of generic (non-category-specific) object appearance and motion in a unified framework. Specifically, we develop a novel two-stream fully convolutional deep segmentation network where individual streams encode generic appearance and motion cues derived from a video frame and its corresponding optical flow. These individual cues are fused in the network to produce a final object versus background pixel-level binary segmentation for each video frame. The proposed network segments both static and moving objects in new videos without any human involvement.

Declaring that motion should assist in video segmentation is non-controversial, and indeed we are certainly not the first to inject motion into video segmentation, as noted above. However, thus far the sum is not much greater than its parts. We contend that this is because *the signal from motion is adequately complex such that rich learned models are necessary to exploit it*. For example, a single object may display multiple motions simultaneously, background and camera motion can intermingle, and even small-magnitude motions should be informative.

To learn the rich signals, sufficient training data is needed. However, no large-scale video datasets with pixel-level segmentations exist. Our second contribution is to address this practical issue. We propose a solution that leverages readily available *image* segmentation annotations together with *weakly annotated video* data to train our model.

Our results show the reward of learning from both signals in a unified framework: a true synergy, often with substantially stronger results than what we can obtain from either one alone—even if they are treated with an equally sophisticated deep network. We significantly advance the state-of-the-art for fully automatic video object segmentation on multiple challenging datasets. In some cases, the proposed method even outperforms existing methods that require manual intervention on the target video. In summary our key contributions are:

- the first end-to-end trainable framework for producing pixel level foreground object segmentation in videos.
- state-of-the-art on multiple datasets, improving over many reported results in the literature and strongly outperforming simpler applications of optical flow, and
- a means to train a deep pixel-level video segmentation model with access to only weakly labeled videos and strongly labeled images, with no explicit assumptions about the categories present in either.

2. Related Work

Automatic methods Fully automatic or unsupervised video segmentation methods assume no human input on the video. They can be grouped into two broad categories. First we have the supervoxel methods [1, 2, 3] which oversegment the video into space-time blobs with cohesive appearance and motion. Their goal is to generate mid-level video regions useful for downstream processing, whereas ours is to produce space-time tubes which accurately delineate object boundaries. Second we have the fully automatic methods that generate thousands of "object-like" space-time segments [21, 22, 23, 24, 25]. While useful in accelerating object detection, it is not straightforward to automatically select the most accurate one when a single hypothesis is desired. Methods that do produce a single hypothesis [4, 5, 6, 8, 9, 26, 27, 28] strongly rely on motion to identify the objects, either by seeding appearance models with moving regions or directly reasoning about occlusion boundaries using optical flow. This limits their capability to segment static objects in video. In comparison, our method is fully automatic, produces a single hypothesis, and can segment both static and moving objects.

Human-guided methods Semi-supervised label propagation methods accept human input on a subset of frames, then propagate it to the remaining frames [10, 11, 29, 12, 13, 14, 15, 20, 30, 31]. In a similar vein, interactive video segmentation methods leverage a human in the loop to provide guidance or correct errors, e.g., [16, 18, 19, 32]. Since the human pinpoints the object of interest, these methods typically focus more on learning object appearance from the manual annotations. Motion is primarily used to propagate information or enforce temporal smoothness. In the proposed method, both motion and appearance play an equally important role, and we show their synergistic combination results in a much better segmentation quality. Moreover, our method is fully automatic and uses no human involvement to segment a novel video.

Category-specific semantic segmentation State-of-the-art semantic segmentation techniques for *images* rely on fully convolutional deep learning architectures that are end-to-end trainable [33, 34, 35, 36]. These deep learning based methods for segmenting images have seen rapid advances in recent years. Unfortunately, video segmentation has not seen such rapid progress. We hypothesize that the lack of large-scale human segmented video segmentation benchmarks is a key bottleneck. Recent video benchmarks like Cityscapes [37] are valuable, but 1) it addresses category-specific segmentation, and 2) thus far methods competing on it process each frame independently, treating it like multiple image segmentation tasks. In contrast, we aim to segment generic objects in video, whether or not they appear in training data. Furthermore, our idea to leverage weakly labeled video for training opens a path towards training deep segmentation models that fuse spatial and temporal cues.

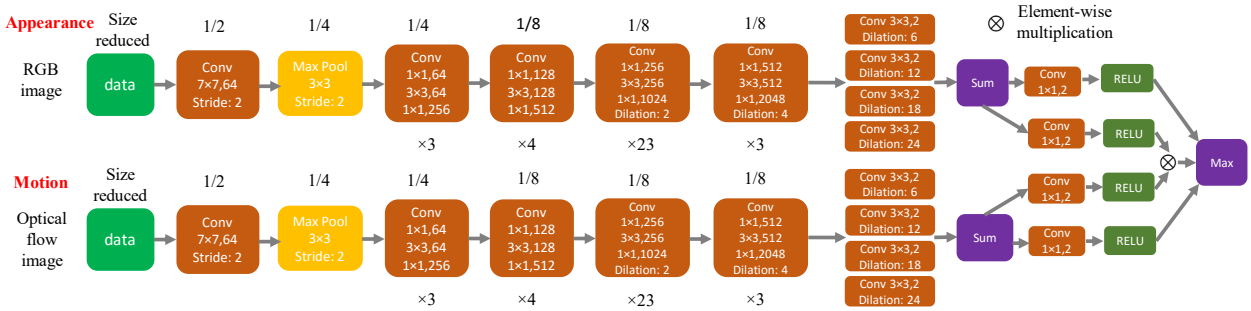

Figure 2: Network structure for our model. Each convolutional layer except the first 7× 7 convolutional layer and our fusion blocks is a residual block [38], adapted from ResNet-101. We show reduction in resolution at top of each box and the number of stacked convolutional layers in the bottom of each box.

Deep learning with motion Deep learning for combining motion and appearance in videos has proven to be useful in several other computer vision tasks such as video classification [39, 40], action recognition [41, 42], object tracking [43, 44, 45] and even computation of optical flow [46]. While we take inspiration from these works, we are the first to present a deep framework for segmenting objects in videos in a fully automatic manner.

3. Approach

Our goal is to segment generic objects in video, independent of the object categories they belong to, and without any manual intervention. We pose the problem as a dense labeling task: given a sequence of video frames $[I_1, I_2, ..., I_N]$, we want to infer either "object" or "background" for each pixel in each frame, to output a sequence of binary maps $[S_1, S_2, ..., S_N]$. We propose a solution based on a convolutional neural network.

First we segment generic objects based on appearance only from individual frames (Sec. 3.1). Then we use the appearance model to generate initial pixel-level annotations in training videos, and bootstrap strong annotations to train a model from motion (Sec. 3.2). Finally, we fuse the two streams to perform video segmentation (Sec. 3.3).

3.1. Appearance Stream

Building on our "pixel objectness" method [47], we train a deep fully convolutional network to learn a model of *generic foreground appearance*. The main idea is to pre-train for object classification, then re-purpose the network to produce binary object segmentations by fine-tuning with relatively few pixel-labeled foreground masks. Pixel objectness uses the VGG architecture [48] and transforms its fully connected layers into convolutional layers. The resulting network possesses a strong notion of objectness, making it possible to identify foreground regions of more than 3,000 object categories despite seeing ground truth masks for only 20 during training.

We take this basic idea and upgrade its implementation for our work. In particular, we adapt the image classification model ResNet-101 [38, 49] by replacing the last two groups of convolution layers with dilated convolution layers to increase feature resolution. This results in only an 8× reduction in the output resolution instead of a 32× reduction in the output resolution in the original ResNet model. In order to improve the model's ability to handle both large and small objects, we replace the classification layer of ResNet-101 with four parallel dilated convolutional layers with different sampling rates to explicitly account for object scale. Then we fuse the prediction from all four parallel layers by summing all the outputs. The loss is the sum of cross-entropy terms over each pixel position in the output layer, where ground truth masks consist of only two labels—object foreground or background. We train the model using the Caffe implementation of [49]. The network takes a video frame of arbitrary size and produces an objectness map of the same size. See Fig. 2 (top stream).

3.2. Motion Stream

Our complete video segmentation architecture consists of a two-stream network in which parallel streams for appearance and motion process the RGB and optical flow images, respectively, then join in a fusion layer (see Fig. 2).

The direct parallel to the appearance stream discussed above would entail training the motion stream to map optical flow maps to video frame foreground maps. However, an important practical catch to that solution is training data availability. While ground truth foreground image segmentations are at least modestly available, datasets for video object segmentation masks are small-scale in deep learning terms, and primarily support evaluation. For example, Segtrack-v2 [7], a commonly used benchmark dataset for video segmentation, contains only 14 videos with 1066 labeled frames. DAVIS [50] contains only 50 sequences with 3455 labeled frames. None contain enough labeled frames to train a deep neural network. Semantic video segmentation datasets like CamVid [51] or Cityscapes [37] are somewhat larger, yet limited in object diversity due to a focus on street scenes and vehicles. A good training source for our task would have ample frames with human-drawn segmentations on a wide variety of foreground objects, and would

show a good mix of static and moving objects. No such large-scale dataset exists and creating one is non-trivial.

We propose a solution that leverages readily available *image* segmentation annotations together with *weakly annotated video* data to train our model. In brief, we temporarily decouple the two streams of our model, and allow the appearance stream to hypothesize likely foreground regions in frames of a large video dataset annotated only by bounding boxes. Since appearance alone need not produce perfect segmentations, we devise a series of filtering stages to generate high quality estimates of the true foreground. These instances bootstrap pre-training of the optical flow stream, then the two streams are joined to learn the best combination from minimal human labeled training videos.

More specifically, given a video dataset with bounding boxes labeled for each object,[1] we ignore the category labels and map the boxes alone to each frame. Then, we apply the appearance stream, thus far trained only from images labeled by their foreground masks, to compute a binary segmentation for each frame.

Next we deconflict the box and segmentation in each training frame. First, we refine the binary segmentation by setting all the pixels outside the bounding box(es) as background. Second, for each bounding box, we check if the the smallest rectangle that encloses all the foreground pixels overlaps with the bounding box by at least 75%. Otherwise we discard the segmentation. Third, we discard regions where the box contains more than 95% pixels labeled as foreground, based on the prior that good segmentations are rarely a rectangle, and thus probably the true foreground spills out beyond the box. Finally, we eliminate segments where object and background lack distinct optical flow, so our motion model can learn from the desired cues. Specifically, we compute the frame's optical flow using [52] and convert it to an RGB flow image [53]. If the 2-norm between a) the average value within the bounding box and b) the average value in a box whose height and width are twice the original size exceeds 30, the frame and filtered segmentation are added to the training set. See Fig. 3 for visual illustration of these steps.

To recap, bootstrapping from the preliminary appearance model, followed by bounding box pruning, bounding box tests, and the optical flow test, we can generate accurate per-pixel foreground masks for thousands of diverse moving objects—for which no such datasets exist to date. Note that by eliminating training samples with these filters, we aim to reduce label noise for training. However, at test time our system will be evaluated on standard benchmarks for which each frame is manually annotated (see Sec. 4).

With this data, we now turn to training the motion stream. Analogous to our strong generic appearance model,

[1]We rely on ImageNet Video data, which contains 3862 videos and 30 diverse objects. See Sec. 4.

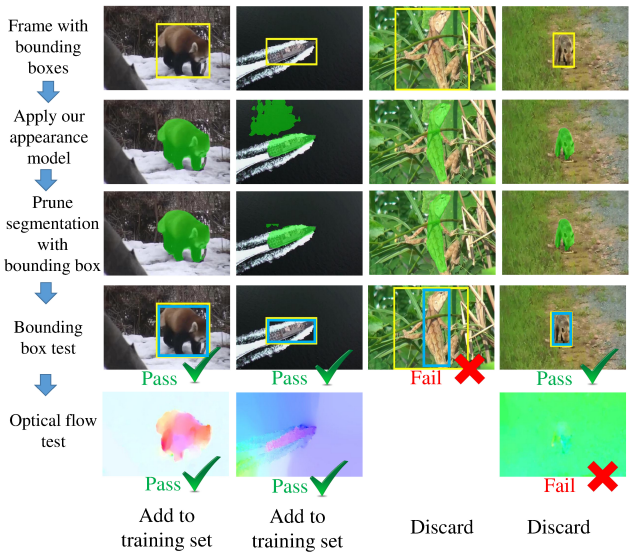

Figure 3: Procedure to generate (pseudo)-ground truth segmentations. We first apply the appearance model to obtain initial segmentations (second row, with object segment in green) and then prune by setting pixels outside bounding boxes as background (third row). Then we apply the bounding box test (fourth row, yellow bounding box is ground truth and blue bounding box is the smallest bounding box enclosing the foreground segment) and optical flow test (fifth row) to determine whether we add the segmentation to the motion stream's training set or discard it. Best viewed in color.

we also want to train a strong generic motion model that can segment foreground objects purely based on motion. We use exactly the same network architecture as the appearance model (see Fig. 2). Our motion model takes only optical flow as the input and is trained with automatically generated pixel level ground truth segmentations. In particular, we convert the raw optical flow to a 3-channel (RGB) color-coded optical flow image [53]. We use this color-coded optical flow image as the input to the motion network. We again initialize our network with pre-trained weights from ImageNet classification [54]. Representing optical flow using RGB flow images allows us to leverage the strong pre-trained initializations as well as maintain symmetry in the appearance and motion arms of the network.

An alternative solution might forgo handing the system optical flow, and instead input two raw consecutive RGB frames. However, doing so would likely demand more training instances in order to discover the necessary cues. Another alternative would directly train the joint model that combines both motion and appearance, whereas we first "pre-train" each stream to make it discover convolutional features that rely on appearance or motion alone, followed by a fusion layer (below). Our design choices are rooted in avoiding bias in training our model. Since the (pseudo) ground truth comes from the initial appearance network, training jointly from the onset is liable to bias the network to exploit appearance at the expense of motion. By feeding the motion model with only optical flow, we ensure our motion stream learns to segment objects from motion.

3.3. Fusion Model

The final processing in our pipeline joins the outputs of the appearance and motion streams, and aims to leverage a whole that is greater than the sum of its parts. We now describe how to train the joint model using both streams.

An object segmentation prediction is reliable if 1) either appearance or motion model alone predicts the object segmentation with very strong confidence or 2) their combination together predicts the segmentation with high confidence. This motivates the structure of our joint model.

We implement the idea by creating three independent parallel branches: 1) We apply a 1×1 convolution layer followed by a RELU to the output of the appearance model 2) We apply a 1×1 convolution layer followed by a RELU to the output of the motion model 3) We replicate the structure of first and second branches and apply element-wise multiplication on their outputs. The element-wise multiplication ensures the third branch outputs confident predictions of object segmentation if and only if both appearance model and motion model have strong predictions. We finally apply a layer that takes the element-wise maximum to obtain the final prediction. See Fig. 2.

As discussed above, we do not fuse the two streams in an early stage because we want them both to have strong independent predictions. Another advantage of our approach is we only introduce six additional parameters in each 1×1 convolution layer, for a total of 24 trainable parameters. We can then train the fusion model with very limited annotated video data, without overfitting. In the absence of large volumes of video segmentation training data, precluding a complete end-to-end training, our strategy of decoupling the individual streams and training works very well in practice.

4. Results

Datasets and metrics: We evaluate our method on three challenging video object segmentation datasets: DAVIS [50], YouTube-Objects [55, 14, 56] and Segtrack-v2 [7]. To measure accuracy we use the standard Jaccard score, which computes the intersection over union overlap (IoU) between the predicted and ground truth object segmentations. The three datasets are:

- **DAVIS [50]:** the latest and most challenging video object segmentation benchmark consisting of 50 high quality video sequences of diverse object categories with $3,455$ densely annotated, pixel-accurate frames. The videos are unconstrained in nature and contain challenges such as occlusions, motion blur, and appearance changes. Only the prominent moving objects are annotated in the ground-truth.
- **YouTube-Objects [55, 14, 56]:** consists of 126 challenging web videos from 10 object categories with more than 20,000 frames and is commonly used for evaluating video object segmentation. We use the subset defined in [56] and the ground truth provided by [14] for evaluation.
- **SegTrack-v2 [7]:** one of the most common benchmarks for video object segmentation consisting of 14 videos with a total of $1,066$ frames with pixel-level annotations. For videos with multiple objects with individual ground-truth segmentations, we treat them as a single foreground for evaluation.

Baselines: We compare with several state-of-the-art methods for each dataset as reported in the literature. Here we group them together based on whether they can operate in a fully automatic fashion (automatic) or require a human in the loop (semi-supervised) to do the segmentation:

- **Automatic methods:** Automatic video segmentation methods do not require any human involvement to segment new videos. Depending on the dataset, we compare with the following state of the art methods: FST [8], KEY [4], NLC [9] and COSEG [26]. All use some form of unsupervised motion or objectness cues to identify foreground objects followed by post-processing to obtain space-time object segmentations.
- **Semi-supervised methods:** Semi-supervised methods bring a human in the loop. They have some knowledge about the object of interest which is exploited to obtain the segmentation (e.g., a manually annotated first frame). We compare with the following state-of-the-art methods: HVS [1], HBT [57], FCP [20], IVID [19], HOP [14], and BVS [30]. The methods require different amounts of human annotation to operate, e.g. HOP, BVS, and FCP make use of manual complete object segmentation in the first frame to seed the method; HBT requests a bounding box around the object of interest in the first frame; HVS, IVID require a human to constantly guide the algorithm whenever it fails.

Note that our method requires human annotated data only during training. At test time it operates in a fully automatic fashion. Thus, given a new video, we require equal effort as the automatic methods, and less effort than the semi-supervised methods.

Apart from these comparisons, we also examine some natural baselines and variants of our method:

- **Flow-thresholding (Flow-Th):** To examine the effectiveness of motion alone in segmenting objects, we adaptively threshold the optical flow in each frame using the flow magnitude. Specifically, we compute the mean and standard deviation from the L2 norm of flow magnitude and use "mean+unit std." as the threshold.
- **Flow-saliency (Flow-Sal):** Optical flow magnitudes can have large variances, hence we also try a variant

DAVIS: Densely Annotated Video Segmentation dataset (50 videos)											
Methods	Flow-Th	Flow-Sal	FST [8]	KEY [4]	NLC [9]	HVS [1]	FCP [20]	BVS [30]	Ours-A	Ours-M	Ours-Joint
Human in loop?	No	No	No	No	No	Yes	Yes	Yes	No	No	No
Avg. IoU	42.95	30.22	57.5	56.9	64.1	59.6	63.1	**66.5**	64.69	60.18	**71.51**

Table 1: Video object segmentation results on DAVIS dataset. We show the average accuracy over all 50 videos. Our method outperforms several state-of-the art methods, including the ones which actually require human annotations during segmentation. The best performing methods grouped by whether they require human-in-the-loop or not during segmentation are highlighted in bold. Metric: Jaccard score, higher is better. Please see supp. for per video results.

which normalizes the flow by applying a saliency detection method [58] to the flow image itself. We use average thresholding to obtain the segmentation.

- **Appearance model (Ours-A):** To quantify the role of appearance in segmenting objects, we obtain segmentations using only the appearance stream of our model.
- **Motion model (Ours-M):** To quantify the role of motion, we obtain segmentations using only the motion stream of our model.
- **Joint model (Ours-Joint):** Our complete joint model that learns to combine both motion and appearance together to obtain the final object segmentation.

Implementation details: To train the appearance stream, we rely on the PASCAL VOC 2012 segmentation dataset [59] and use a total of 10,582 training images with binary object vs. background masks (see [47] for more details). As weak bounding box video annotations, we use the ImageNet-Video dataset [54]. This dataset comes with a total of 3,862 training videos from 30 object categories with 866,870 labeled object bounding boxes from over a million frames. Post refinement using our ground truth generation procedure (see Sec. 3.2), we are left with 84,929 frames with good pixel segmentations[2] which are then used to train our motion model. For training the joint model we use a held-out set for each dataset. We train each stream for a total of 20,000 iterations, use "poly" learning rate policy (power = 0.9) with momentum (0.9) and weight decay (0.0005). No post-processing is applied on the segmentations obtained from our networks.

Quality of training data: To ascertain that the quality of training data we automatically generate for training our motion stream is good, we first compare it with a small amount of human annotated ground truth. We randomly select 100 frames that passed both the bounding box and optical flow tests, and collect human-drawn segmentations on Amazon MTurk. We first present crowd workers a frame with a bounding box labeled for each object, and then ask them to draw the detailed segmentation for all objects within the bounding boxes. Each frame is labeled by three crowd workers and the final segmentation is obtained by majority vote on each pixel. The results indicate that our strategy to gather pseudo-ground truth is effective. On the 100 labeled frames, Jaccard overlap with the human-drawn ground truth is 77.8 (and 70.2 before pruning with bounding boxes).

[2]Available for download on our project website.

Quantitative evaluation: We now present the quantitative comparisons of our method with several state-of-the-art methods and baselines, for each of the three datasets in turn.

DAVIS dataset: Table 1 shows the results, with some of the best performing methods taken from the benchmark results [50]. Our method outperforms all existing methods on this dataset and significantly advances state-of-the-art. Our method is significantly better than simple flow baselines. This supports our claim that even though motion contains a strong signal about foreground objects in videos, it is not straightforward to simply threshold optical flow and obtain those segmentations. A data-driven approach that learns to identify motion patterns indicative of objects as opposed to backgrounds or camera motion is required.

The appearance and motion variants of our method themselves result in a very good performance. The performance of the motion variant is particularly impressive, knowing that it has no information about object's appearance and purely relies on the flow signal. When combined together, the joint model results in a significant improvement, with an absolute gain of up to 11% over individual streams.

Our method is also significantly better than fully automatic methods, which typically rely on motion alone to identify foreground objects. This illustrates the benefits of a unified combination of both motion and appearance. Most surprisingly, our method significantly outperforms even the state-of-the-art semi-supervised techniques, which require substantial human annotation on every video they process. The main motivation behind bringing a human in the loop is to achieve higher accuracies than fully automated methods, yet in this case, our proposed fully automatic method outperforms the best human-in-the-loop algorithms by a significant margin. For example, the BVS [30] method—which is the current best performing semi-supervised method and requires the first frame of the video to be manually segmented—achieves an overlap score of 66.5%. Our method significantly outperforms it with an overlap score of 71.51%, yet uses no human involvement.

YouTube-Objects dataset: In Table 2 we see a similarly

YouTube-Objects dataset (126 videos)										
Methods	Flow-Th	Flow-Sal	FST [8]	COSEG [26]	HBT [57]	HOP [14]	IVID [19]	Ours-A	Ours-M	Ours-Joint
Human in loop?	No	No	No	No	Yes	Yes	Yes	No	No	No
airplane (6)	18.27	33.32	70.9	69.3	73.6	86.27	**89**	**83.38**	59.38	81.74
bird (6)	31.63	33.74	70.6	**76**	56.1	81.04	**81.6**	60.89	64.06	63.84
boat (15)	4.35	22.59	42.5	53.5	57.8	68.59	**74.2**	**72.62**	40.21	72.38
car (7)	21.93	48.63	65.2	70.4	33.9	69.36	**70.9**	74.50	61.32	**74.92**
cat (16)	19.9	32.33	52.1	66.8	30.5	58.89	**67.7**	67.99	49.16	**68.43**
cow (20)	16.56	29.11	44.5	49	41.8	68.56	**79.1**	**69.63**	39.38	68.07
dog (27)	17.8	25.43	65.3	47.5	36.8	61.78	**70.3**	69.10	54.79	**69.48**
horse (14)	12.23	24.17	53.5	55.7	44.3	53.96	**67.8**	**62.79**	39.96	60.44
mbike (10)	12.99	17.06	44.2	39.5	48.9	60.87	**61.5**	61.92	42.95	**62.74**
train (5)	18.16	24.21	29.6	53.4	39.2	66.33	**78.2**	**62.82**	43.13	62.20
Avg. IoU	17.38	29.05	53.84	58.11	46.29	67.56	**74.03**	**68.57**	49.43	68.43

Table 2: Video object segmentation results on YouTube-Objects dataset. We show the average performance for each of the 10 categories from the dataset. The final row shows an average over all the videos. Our method outperforms several state-of-the art methods, including the ones which actually require human annotation during segmentation. The best performing methods grouped by whether they require human-in-the-loop or not during segmentation are highlighted in bold. Metric: Jaccard score, higher is better.

Segtrack-v2 dataset (14 videos)										
Methods	Flow-Th	Flow-Sal	FST [8]	KEY [4]	NLC [9]	HBT [57]	HVS [1]	Ours-A	Ours-M	Ours-Joint
Human in loop?	No	No	No	No	No	Yes	Yes	No	No	No
Avg. IoU	37.77	27.04	53.5	57.3	**80***	41.3	**50.8**	56.88	53.04	61.40

Table 3: Video object segmentation results on Segtrack-v2. We show the average accuracy over all 14 videos. Our method outperforms several state-of-the art methods, including the ones which actually require human annotation during segmentation. The best performing methods grouped by whether they require human-in-the-loop or not during segmentation are highlighted in bold. *For NLC results are averaged over 12 videos as reported in their paper [9]. Metric: Jaccard score, higher is better. Please see supp. for per video results.

strong result on the YouTube-Objects dataset. Our method again outperforms the flow baselines and all the automatic methods by a significant margin. The publicly available code for NLC [9] runs successfully only on 9% of the YouTube dataset (1725 frames); on those, its jaccard score is 43.64%. Our proposed model outperforms it by a significant margin of 25%. Even among human-in-the-loop methods, we outperform all methods except IVID [19]. However, IVID [19] requires a human to consistently track the segmentation performance and correct whatever mistakes the algorithm makes. This can take up to minutes of annotation time for each video. Our method uses zero human involvement but still performs competitively.

It is also important to note that this dataset shares categories with the PASCAL segmentation benchmark which is used to train our appearance stream. Accordingly, we observe that the appearance stream itself results in the overall best performance. Moreover, this dataset has a mix of static and moving objects which explains the relatively weaker performance of our motion model alone. Overall the joint model works similarly well as appearance alone, however our ablation study (see Table 4) where we rank test frames by their amount of motion, shows that our joint-model is stronger for moving objects. In short, our joint model outperforms our appearance model on moving objects, while our appearance model is sufficient for the most static frames. Whereas existing methods tend to suffer in one extreme or the other, our method handles both well.

Methods	Top 10% moving	Top 10% static
Ours-A	71.58	61.79
Ours-Joint	72.34	59.86

Table 4: Ablation study for YouTube-Objects dataset: Performance of our appearance and joint models on frames with most (left) and least (right) motion.

Segtrack-v2 dataset: In Table 3, our method outperforms all semi-supervised and automatic methods except NLC [9] on Segtrack. While our approach significantly outperforms NLC [9] on the DAVIS dataset, NLC is exceptionally strong on this dataset. Our relatively weaker performance could be due to the low quality and resolution of the Segtrack-v2 videos, making it hard for our network based model to process them. Nonetheless, our joint model still provides a significant boost over both our appearance and motion models, showing it again realizes the synergy of motion and appearance in a serious way.

Qualitative evaluation: Fig. 4 shows qualitative results. The top half shows visual comparisons between different components of our method including the appearance, motion, and joint models. We also show the optical flow image that was used as an input to the motion stream. These images help reveal the complexity of learned motion signals. In the bear example, the flow is most salient only on the bear's head, still our motion stream alone is able to segment the bear completely. The boat, car, and sail examples show that even when the flow is noisy—including strong

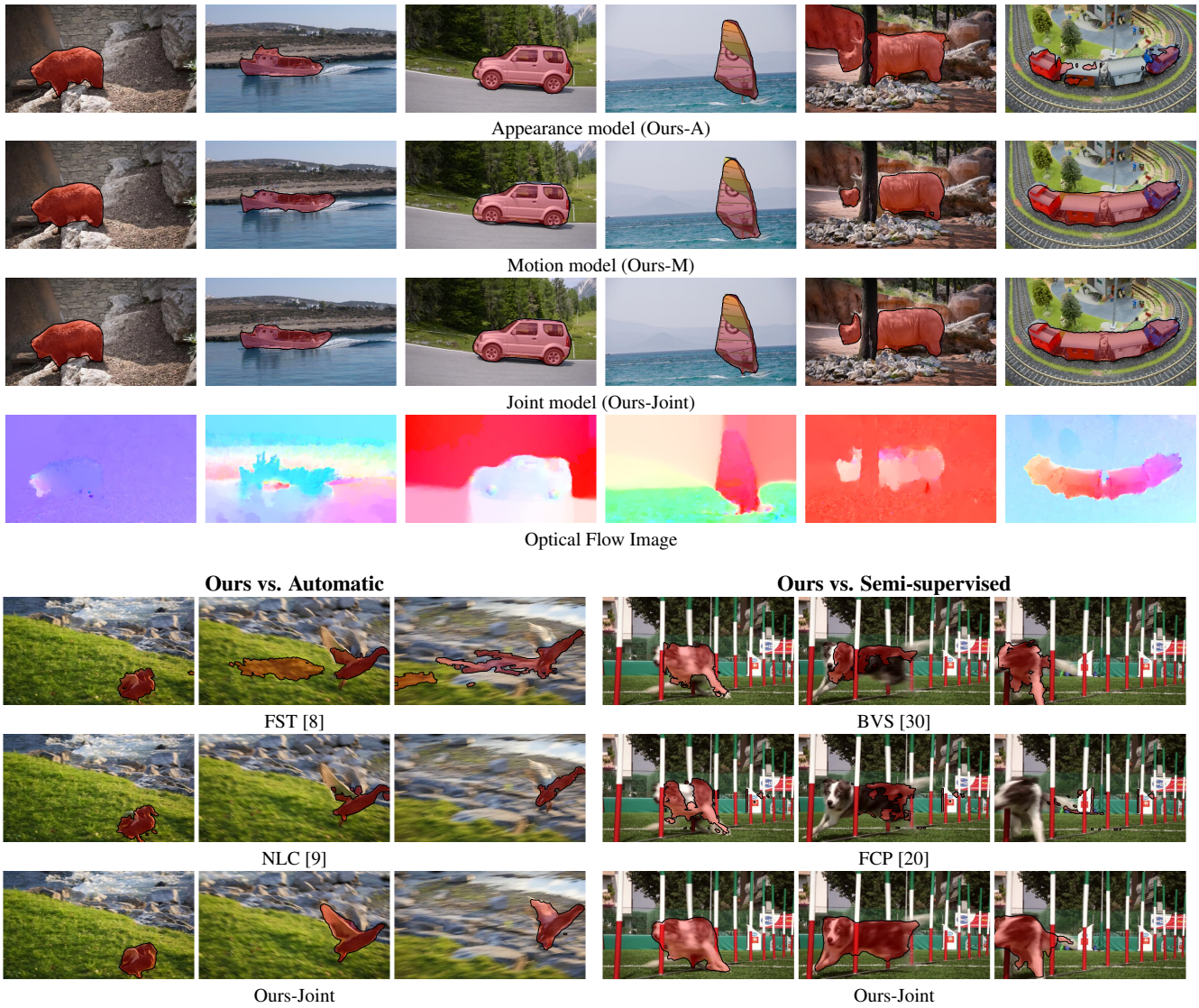

Figure 4: Qualitative results: The top half shows examples from our appearance, motion, and joint models along with the flow image which was used as an input to the motion network. The bottom rows show visual comparisons of our method with automatic and semi-supervised baselines (best viewed on pdf and see text for the discussion). Videos of our segmentation results are available on the project website.

flow on the background—our motion model is able to learn about object shapes and successfully suppresses the background. The rhino and train examples show cases where the appearance model fails but when combined with the motion stream, the joint model produces accurate segmentations.

The bottom half of Fig. 4 shows visual comparisons between our method and state-of-the-art automatic [8, 9] and semi-supervised [20, 30] methods. The automatic methods have a very weak notion about object's appearance; hence they completely miss parts of objects [9] or cannot disambiguate the objects from background [8]. Semi-supervised methods [20, 30], which rely heavily on the initial human-segmented frame to learn about object's appearance, start to fail as time elapses and the object's appearance changes considerably. In contrast, our method successfully learns to combine generic cues about object motion and appearance, segmenting much more accurately across all frames even in very challenging videos.

5. Conclusions

We presented a new approach for learning to segment generic objects in video that 1) achieves deeper synergy between motion and appearance and 2) addresses practical challenges in training a deep network for video segmentation. Results show sizeable improvements over many existing methods—in some cases, even those requiring human intervention. In future work we plan to explore extensions that could permit individuation of multiple touching foreground objects, as well as ways to incorporate

human intervention intelligently into our framework.

Video examples, code & pre-trained models available at:
`http://vision.cs.utexas.edu/projects/fusionseg/`

Acknowledgements: This research is supported in part by ONR YIP N00014-12-1-0754.

6. Appendix

Per-video results for DAVIS and Segtrack-v2: Table 5 shows the per video results for the 50 videos from the DAVIS dataset. Table 1 in the main paper summarizes these results over all 50 videos. We compare with several semi-supervised and fully automatic baselines. Our method outperforms the per-video best fully automatic and semi-supervised baseline in 25 out of 50 videos.

Table 6 shows the per video results for the 14 videos from the Segtrack-v2 dataset. Table 3 in the main paper summarizes these results over all 14 videos. Our method outperforms the per-video best fully automatic method in 5 out of 14 cases. Our method also outperforms the semi-supervised HVS [1] method in 8 out of 14 cases.

References

[1] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchical graph based video segmentation," in *CVPR*, 2010. 1, 2, 5, 6, 7, 9, 10, 11

[2] C. Xu, C. Xiong, and J. J. Corso, "Streaming Hierarchical Video Segmentation," in *ECCV*, 2012. 1, 2

[3] F. Galasso, R. Cipolla, and B. Schiele, "Video segmentation with superpixels," in *ACCV*, 2012. 1, 2

[4] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *ICCV*, 2011. 1, 2, 5, 6, 7, 10, 11

[5] T. Ma and L. Latecki, "Maximum weight cliques with mutex constraints for video object segmentation," in *CVPR*, 2012. 1, 2

[6] D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," in *CVPR*, 2013. 1, 2

[7] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg, "Video Segmentation by Tracking Many Figure-Ground Segments," in *ICCV*, 2013. 1, 3, 5

[8] A. Papazoglou and V. Ferrari, "Fast object segmentation in unconstrained video," in *ICCV*, 2013. 1, 2, 5, 6, 7, 8, 10, 11

[9] A. Faktor and M. Irani, "Video segmentation by non-local consensus voting," in *BMVC*, 2014. 1, 2, 5, 6, 7, 8, 10, 11

[10] X. Ren and J. Malik, "Tracking as repeated figure/ground segmentation," in *CVPR*, 2007. 1, 2

[11] D. Tsai, M. Flagg, and J. Rehg, "Motion coherent tracking with multi-label mrf optimization," in *BMVC*, 2010. 1, 2

[12] A. Fathi, M. Balcan, X. Ren, and J. Rehg, "Combining self training and active learning for video segmentation," in *BMVC*, 2011. 1, 2

[13] S. Vijayanarasimhan and K. Grauman, "Active frame selection for label propagation in videos," in *ECCV*, 2012. 1, 2

[14] S. D. Jain and K. Grauman, "Supervoxel-consistent foreground propagation in video," in *ECCV*, 2014. 1, 2, 5, 7

[15] L. Wen, D. Du, Z. Lei, S. Z. Li, and M.-H. Yang, "Jots: Joint online tracking and segmentation," in *CVPR*, June 2015. 1, 2

[16] J. Wang, P. Bhat, A. Colburn, M. Agrawala, and M. F. Cohen, "Interactive video cutout," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 585–594, 2005. 1, 2

[17] Y. Li, J. Sun, and H.-Y. Shum, "Video object cut and paste," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 595–600, 2005. 1

[18] X. Bai, J. Wang, D. Simons, and G. Sapiro, "Video snapcut: Robust video object cutout using localized classifiers," in *SIGGRAPH*, 2009. 1, 2

[19] N. Shankar Nagaraja, F. R. Schmidt, and T. Brox, "Video segmentation with just a few strokes," in *ICCV*, 2015. 1, 2, 5, 7

[20] F. Perazzi, O. Wang, M. Gross, and A. Sorkine-Hornung, "Fully connected object proposals for video segmentation," in *ICCV*, December 2015. 1, 2, 5, 6, 8, 10

[21] Z. Wu, F. Li, R. Sukthankar, and J. M. Rehg, "Robust video segment proposals with painless occlusion handling," in *CVPR*, June 2015. 2

[22] K. Fragkiadaki, P. Arbelaez, P. Felsen, and J. Malik, "Learning to segment moving objects in videos," in *CVPR*, June 2015. 2

[23] G. Yu and J. Yuan, "Fast action proposals for human action detection and search," in *CVPR*, June 2015. 2

[24] D. Oneata, J. Revaud, J. Verbeek, and C. Schmid, "Spatio-temporal object detection proposals," in *ECCV*, Sep 2014. 2

[25] F. Xiao and Y. J. Lee, "Track and segment: An iterative unsupervised approach for video object proposals," in *CVPR*, 2016. 2

[26] Y.-H. Tsai, G. Zhong, and M.-H. Yang, "Semantic co-segmentation in videos," in *ECCV*, 2016. 2, 5, 7

[27] P. Sundberg, T. Brox, M. Maire, P. Arbelaez, and J. Malik, "Occlusion boundary detection and figure/ground assignment from optical flow," in *CVPR*, 2011. 2

[28] D. Hoiem, M. Hebert, and A. Stein, "Learning to find object boundaries using motion cues," *ICCV*, 2007. 2

[29] V. Badrinarayanan, F. Galasso, and R. Cipolla, "Label propagation in video sequences," in *CVPR*, 2010. 2

[30] N. Märki, F. Perazzi, O. Wang, and A. Sorkine-Hornung, "Bilateral space video segmentation," in *CVPR*, 2016. 2, 5, 6, 8, 10

[31] Y.-H. Tsai, M.-H. Yang, and M. J. Black, "Video segmentation via object flow," in *CVPR*, 2016. 2

DAVIS: Densely Annotated Video Segmentation dataset (50 videos)									
Methods	FST [8]	KEY [4]	NLC [9]	HVS [1]	FCP [20]	BVS [30]	Ours-A	Ours-M	Ours-Joint
Human in loop?	No	No	No	Yes	Yes	Yes	No	No	No
Bear	89.8	89.1	90.7	93.8	90.6	95.5	91.52	86.30	90.66
Blackswan	73.2	84.2	87.5	91.6	90.8	94.3	89.54	61.71	81.10
Bmx-Bumps	24.1	30.9	63.5	42.8	30	43.4	38.77	26.42	32.97
Bmx-Trees	18	19.3	21.2	17.9	24.8	38.2	34.67	37.08	43.54
Boat	36.1	6.5	0.7	78.2	61.3	64.4	63.80	59.53	66.35
Breakdance	46.7	54.9	67.3	55	56.7	50	14.22	61.80	51.10
Breakdance-Flare	61.6	55.9	80.4	49.9	72.3	72.7	54.87	62.09	76.21
Bus	82.5	78.5	62.9	80.9	83.2	86.3	80.38	77.70	82.70
Camel	56.2	57.9	76.8	87.6	73.4	66.9	76.39	74.19	83.56
Car-Roundabout	80.8	64	50.9	77.7	71.7	85.1	74.84	84.75	90.15
Car-Shadow	69.8	58.9	64.5	69.9	72.3	57.8	88.38	81.03	89.61
Car-Turn	85.1	80.6	83.3	81	72.4	84.4	90.67	83.92	90.23
Cows	79.1	33.7	88.3	77.9	81.2	89.5	87.96	82.22	86.82
Dance-Jump	59.8	74.8	71.8	68	52.2	74.5	10.32	64.22	61.16
Dance-Twirl	45.3	38	34.7	31.8	47.1	49.2	46.23	55.39	70.42
Dog	70.8	69.2	80.9	72.2	77.4	72.3	90.41	81.90	88.92
Dog-Agility	28	13.2	65.2	45.7	45.3	34.5	68.94	67.88	73.36
Drift-Chicane	66.7	18.8	32.4	33.1	45.7	3.3	46.13	44.14	59.86
Drift-Straight	68.3	19.4	47.3	29.5	66.8	40.2	67.24	69.08	81.06
Drift-Turn	53.3	25.5	15.4	27.6	60.6	29.9	85.09	72.09	86.30
Elephant	82.4	67.5	51.8	74.2	65.5	85	86.18	77.51	84.35
Flamingo	81.7	69.2	53.9	81.1	71.7	88.1	44.46	63.80	75.67
Goat	55.4	70.5	1	58	67.7	66.1	84.11	74.99	83.09
Hike	88.9	89.5	91.8	87.7	87.4	75.5	82.54	58.30	76.90
Hockey	46.7	51.5	81	69.8	64.7	82.9	66.03	44.89	70.05
Horsejump-High	57.8	37	83.4	76.5	67.6	80.1	71.09	54.10	64.93
Horsejump-Low	52.6	63	65.1	55.1	60.7	60.1	70.23	55.20	71.20
Kite-Surf	27.2	58.5	45.3	40.5	57.7	42.5	47.71	18.54	38.98
Kite-Walk	64.9	19.7	81.3	76.5	68.2	87	52.65	39.35	49.00
Libby	50.7	61.1	63.5	55.3	31.6	77.6	67.70	35.34	58.48
Lucia	64.4	84.7	87.6	77.6	80.1	90.1	79.93	49.18	77.31
Mallard-Fly	60.1	58.5	61.7	43.6	54.1	60.6	74.62	42.64	68.46
Mallard-Water	8.7	78.5	76.1	70.4	68.7	90.7	83.34	25.31	79.43
Motocross-Bumps	61.7	68.9	61.4	53.4	30.6	40.1	83.78	56.56	77.15
Motocross-Jump	60.2	28.8	25.1	9.9	51.1	34.1	80.43	59.02	77.50
Motorbike	55.9	57.2	71.4	68.7	71.3	56.3	28.67	45.71	41.15
Paragliding	72.5	86.1	88	90.7	86.6	87.5	17.68	60.76	47.42
Paragliding-Launch	50.6	55.9	62.8	53.7	57.1	64	58.88	50.34	57.00
Parkour	45.8	41	90.1	24	32.2	75.6	79.39	58.51	75.81
Rhino	77.6	67.5	68.2	81.2	79.4	78.2	77.56	83.03	87.52
Rollerblade	31.8	51	81.4	46.1	45	58.8	63.27	57.73	69.01
Scooter-Black	52.2	50.2	16.2	62.4	50.4	33.7	36.07	62.18	68.47
Scooter-Gray	32.5	36.3	58.7	43.3	48.3	50.8	73.22	61.69	73.40
Soapbox	41	75.7	63.4	68.4	44.9	78.9	49.70	53.24	62.57
Soccerball	84.3	87.9	82.9	6.5	82	84.4	29.27	73.56	79.72
Stroller	58	75.9	84.9	66.2	59.7	76.7	63.91	54.40	66.55
Surf	47.5	89.3	77.5	75.9	84.3	49.2	88.78	73.00	88.41
Swing	43.1	71	85.1	10.4	64.8	78.4	73.75	59.41	74.05
Tennis	38.8	76.2	87.1	57.6	62.3	73.7	76.88	47.19	70.75
Train	83.1	45	72.9	84.6	84.1	87.2	42.50	80.33	75.56
Avg. IoU	57.5	56.9	64.1	59.6	63.1	**66.5**	64.69	60.18	**71.51**

Table 5: Video object segmentation results on DAVIS dataset. We show the results for all 50 videos. Table 1 in the main paper summarizes these results over all 50 videos. Our method outperforms several state-of-the art methods, including the ones which actually require human annotation during segmentation. The best performing methods grouped by whether they require human-in-the-loop or not during segmentation are highlighted in bold. Metric: Jaccard score, higher is better.

[32] B. L. Price, B. S. Morse, and S. Cohen, "Livecut: Learning-based interactive video segmentation by evaluation of multiple propagated cues," in *ICCV*, 2009. 2

[33] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *ICCV*, 2015. 2

[34] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr, "Conditional random fields as recurrent neural networks," in *ICCV*, 2015. 2

[35] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *CVPR*, Nov. 2015. 2

Segtrack-v2 dataset (14 videos)							
Methods	FST [8]	KEY [4]	NLC [9]	HVS [1]	Ours-A	Ours-M	Ours-Joint
Human in loop?	No	No	No	Yes	No	No	No
birdfall2	17.50	49.00	74.00	57.40	6.94	55.50	38.01
bird of paradise	81.83	92.20	-	86.80	49.82	62.46	69.91
bmx	67.00	63.00	79.00	35.85	59.53	55.12	59.08
cheetah	28.00	28.10	69.00	21.60	71.15	36.00	59.59
drift	60.50	46.90	86.00	41.20	82.18	80.03	87.64
frog	54.13	0.00	83.00	67.10	54.86	52.88	57.03
girl	54.90	87.70	91.00	31.90	81.07	43.57	66.73
hummingbird	52.00	60.15	75.00	19.45	61.50	60.86	65.19
monkey	65.00	79.00	71.00	61.90	86.42	58.95	80.46
monkeydog	61.70	39.60	78.00	43.55	39.08	24.36	32.80
parachute	76.32	96.30	94.00	69.10	24.86	59.43	51.58
penguin	18.31	9.27	-	74.45	66.20	45.09	71.25
soldier	39.77	66.60	83.00	66.50	83.70	48.37	69.82
worm	72.79	84.40	81.00	34.70	29.13	59.94	50.63
Avg. IoU	53.5	57.3	**80***	**50.8**	56.88	53.04	61.40

Table 6: Video object segmentation results on Segtrack-v2. We show the results for all 14 videos. Table 3 in the main paper summarizes these results over all 14 videos. Our method outperforms several state-of-the art methods, including the ones which actually require human annotation during segmentation. For NLC results are averaged over 12 videos as reported in their paper [9]. The best performing methods grouped by whether they require human-in-the-loop or not during segmentation are highlighted in bold. Metric: Jaccard score, higher is better.

[36] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," in *ICLR*, 2015. [Online]. Available: http://arxiv.org/abs/1412.7062 2

[37] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016. 2, 3

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 3

[39] J. Y.-H. Ng, M. J. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification." in *CVPR*, 2015. 3

[40] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *CVPR*, 2014. 3

[41] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NIPS*, 2014. 3

[42] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013. 3

[43] H. Li, Y. Li, and F. Porikli, "Deeptrack: Learning discriminative feature representations by convolutional neural networks for visual tracking," in *BMVC*, 2014. 3

[44] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *ICCV*, December 2015. 3

[45] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *ICCV*, 2015. 3

[46] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *ICCV*, December 2015. 3

[47] S. Jain, B. Xiong, and K. Grauman, "Pixel objectness," *arXiv preprint arXiv:1701.05349*, 2017. 3, 6

[48] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. 3

[49] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *arXiv preprint arXiv:1606.00915*, 2016. 3

[50] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *CVPR*, 2016. 3, 5, 6

[51] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, 2009. 3

[52] C. Liu, "Beyond pixels: exploring new representations and applications for motion analysis," Ph.D. dissertation, Citeseer, 2009. 4

[53] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *International Journal of Computer Vision*, vol. 92, no. 1, pp. 1–31, 2011. 4

[54] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015. 4, 6

[55] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari, "Learning object class detectors from weakly annotated video," in *CVPR*, 2012. 5

[56] K. Tang, R. Sukthankar, J. Yagnik, and L. Fei-Fei, "Discriminative segment annotation in weakly labeled video," in *CVPR*, 2013. 5

[57] M. Godec, P. M. Roth, and H. Bischof, "Hough-based tracking of non-rigid objects," in *ICCV*, 2011. 5, 7

[58] B. Jiang, L. Zhang, H. Lu, C. Yang, and M.-H. Yang, "Saliency detection via absorbing markov chain," in *ICCV*, 2013, pp. 1665–1672. 6

[59] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010. 6