Supplementary material

Unsupervised space-time network for temporally-consistent segmentation of multiple motions

Etienne Meunier
Inria, Rennes, France
etienne.meunier@inria.fr

Patrick Bouthemy
Inria, Rennes, France
patrick.bouthemy@inria.fr

1. Linking predictions and segments selection

We detail the procedure for linking predictions in a subsequence and for selecting optimal segments for evaluation. Subsequences are composed of $T+1$ consecutive flow fields. Let us define:

$$\breve{m}_t(i) = \arg\max_{k=1,..,K} m_k(i,t), \tag{1}$$

where $m_k(i,t)$ is the probability of site i to belong to segment k at time t. Let $\breve{m}_t$ be the segmentation map at time t encompassing the (up to) K segments predicted by the network, $\breve{m}_t = \{\breve{m}_t(i), i \in \mathcal{I}\}$. In other words, $\breve{m}_t$ is the label array representing the set of segments extracted at time t, segments being (non necessary connected) layers. We will also use the term mask to designate $\breve{m}_t$, when no confusion can occur. The prediction of the network is given for a triplet of input flow fields $(f_{t-1}; f_t; f_{t+1})$ as a triplet of masks $(\breve{m}_{t-1}; \breve{m}_t; \breve{m}_{t+1})$ for $\tau = 1$. All those triplet predictions are produced in parallel and the triplet output are independent as illustrated in Fig.1.

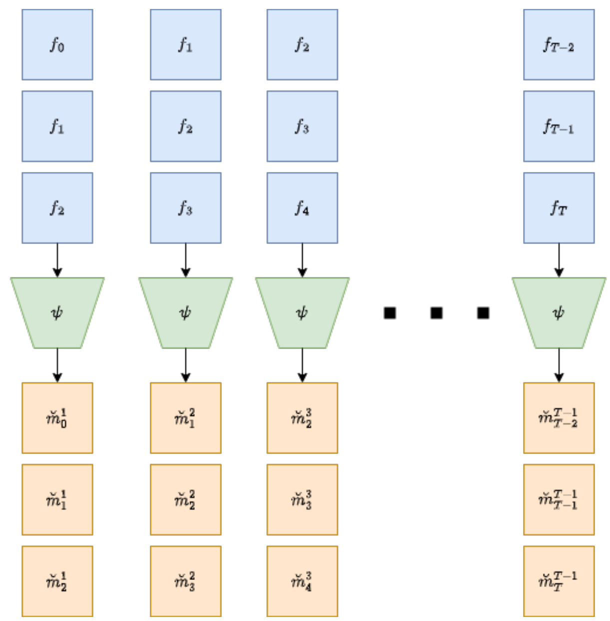

Figure 1. Output of the network by triplet (for $\tau = 1$) within a subsequence covering the time interval $[0, T]$. The lower index t of each mask $\breve{m}_t^{t'}$ represents the time instant of the corresponding flow field f_t, and the upper one t' corresponds to the time instant when it is produced, that is, the one of the reference (central) flow field of the triplet. Segmentation masks with the same lower index correspond to different segmentation instances of the same flow field.

1.1. Linking predictions

The masks in the same triplet are sharing common labels but not necessary across triplets. By label, we mean mask number. We have to link the labels by finding correspondences between triplets. Since we have three versions of the same mask $\breve{m}_t$, it is straightforward to achieve it.

First, we need to introduce an additional notation $\breve{m}_t^{t'}$ as defined below. The segmentation mask $\breve{m}_t^{t'}$ ($t' \in \{t-1, t, t+1\}$) of width W and height H, consisting of K non-overlapping classes ($\breve{m}_t^{t'} \in \{1, .., K\}^{W \times H}$) and corresponding to flow f_t, is the one predicted by the network when it takes a triplet centered around t' as input. We have to find the best label association between instances $\breve{m}_t^{t'}$ of the same mask. To do that, we compute a label reassignment table that will be applied to the two masks $\breve{m}_t^t$ and $\breve{m}_{t-1}^t$. The reassigned label l^* for each label $l \in \{1, .., K\}$ is given by:

$$l^* = \arg\max_{k \in \{1,..,K\}} J(k_t^{t-1}, l_t^t) + J(k_{t-1}^{t-1}, l_{t-1}^t), \tag{2}$$

where J is the IoU score between two segments. The binary array $k_t^{t'}$ is defined as follows:

$$k_t^{t'}(i) = \begin{cases} 1 & \text{if } \breve{m}_t^{t'}(i) = k \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

$l_t^{t'}$ is defined in a similar way. We proceed by pairs, since two consecutive triplets share two masks as illustrated in Fig.2 (the pairs of arrows). Starting from $t = 0$, we propagate the labels to the whole subsequence using the criterion of eq.(2). After the label reassignment propagation, we rename each $\breve{m}_t^{t'}$ as $\bar{m}_t^{t'}$.

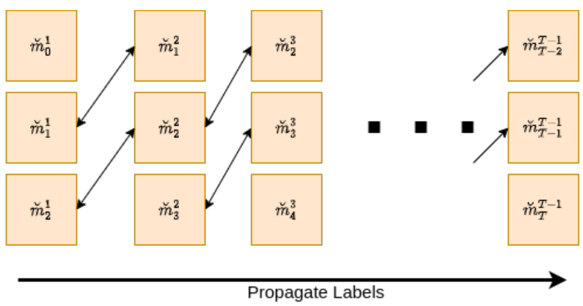

Figure 2. Propagation of the labels (i.e., masks numbers) over the subsequence.

1.2. Select optimal labels

We select an optimal set of segments for evaluation, based on the ground truth, for the entire subsequence. The goal is to show that, since we have coherent labels within the subsequence, we can select the optimal segments at the subsequence level. First, we unroll our sequence by only keeping the central prediction for each step from $t = 1$ to $t = T - 1$, and just retrieving the single instance produced for the first ($t = 0$) and last ($t = T$) time steps as depicted in Fig.3.

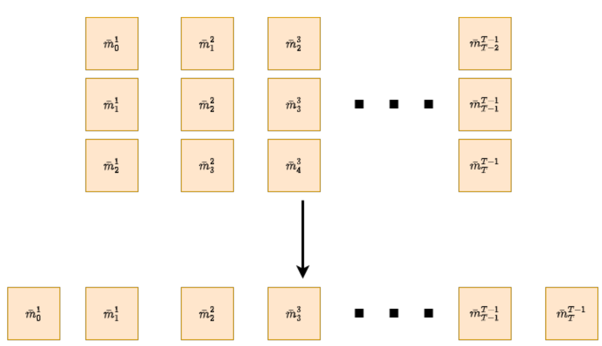

Figure 3. Formation of the mask series $\{\bar{m}_t, t = 0, .., T\}$ over the subsequence for evaluation , once the label propogation step is achieved.

Then, for the entire subsequence, we select the subset $\mathcal{S}^*$ of labels (that is, of segments) to constitute the predicted foreground, using the binary ground-truth g_t. The selected label subset $\mathcal{S}^*$ is given by:

$$\mathcal{S}^* = \arg \max_{\mathcal{S} \subset \mathcal{P}(\{1,..,K\})} \sum_{t=0}^{T} J(\bigcup_{l \in \mathcal{S}} \bar{l}_t, g_t), \tag{4}$$

where $\mathcal{P}(\{1, .., K\})$ is the partition of the labels in the subsequence, and the binary masks $\bar{l}_t$ correspond to the label mask $\bar{m}_t$. Once we have selected the subset $\mathcal{S}^*$ of labels, we can use it to build our binary segmentation $\{s_t, t = 0, .., T\}$ on the whole subsequence for evaluation, as illustrated in Fig.5.

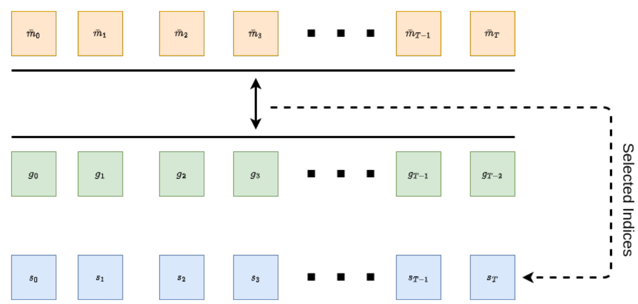

Figure 4. Comparison of the selected segments with the ground truth for evaluation.

2. Impact of subsequence length

In the temporal segment linkage and the segment selection process described above, we take into account a subsequence of length $T + 1$. For the results reported in the main paper, we used a subsequence length of 10 frames. However, we can vary the subsequence length to evaluate the robustness of our method to this parameter. All the evaluations regarding the impact of the subsequence length, are produced from the same trained network and initial segmentation. They are plotted in Fig.5. Longer subsequences are more challenging since they require a stronger temporal consistency. Also, they can be more impacted by occlusions or flow estimation errors, which can break label propagation.

We can see that our method is robust to the choice of the subsequence length, and that it is performing well on long subsequences as well.

3. Additional experiments

3.1. Training without data augmentation

To extend our ablation study, we have also carried out the evaluation of the case when our network is trained without any data augmentation. Results are collected in Table 1. Clearly, as expected, the data augmentation improves performance.

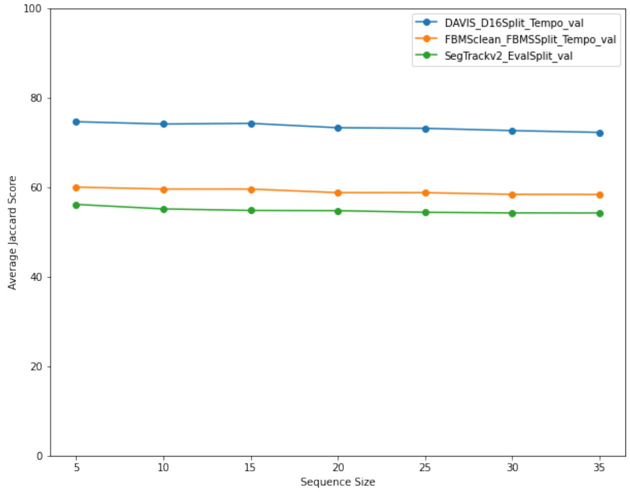

Figure 5. Evaluation of our temporal segment linkage and segment selection process for different subsequence lengths on the three datasets.

Data Augmentation	Davis Val	SegTrackV2	FBMS
With	73.2	55.0	59.4
Without	70.1	52.3	50.4

Table 1. Impact of the data augmentation. Scores (Jaccard index) obtained on the three datasets.

3.2. Impact of the temporal interval τ

Regarding the τ parameter (i.e., the temporal interval between the flows of the input triplet), we trained our network with the full configuration and randomly sampled τ values at training time. Let us remind that the flow $f_{t-\tau}$ (respectively, $f_{t+\tau}$) is computed between image frames at time instants $t-\tau$ and $t-\tau+1$ (respectively, $t+\tau$ and $t+\tau+1$). We uniformly sample τ among a set of values during training. At inference, we still use only $\tau = 1$ for the sake of efficiency. Thus, this experiment could also be perceived as a type of data augmentation.

τ while training	Davis Val	SegTrackV2	FBMS
{1}	73.2	55.0	59.4
{1,2,3,4,5,9,10,12}	73.0	54.3	57.9

Table 2. Impact of the use, at training time, of different values for the time interval τ in the input flow triplet. Scores (Jaccard index) obtained on the three datasets.

As we can observe in Table 2, it has no sensible impact on results (even a slight performance decrease). In future work, we plan to investigate the combination of different τ values, including negative values, at test time.

3.3. Results on Davis2017-motion

In addition to the datasets (DAVIS2016, FBMS59, SegTrackV2), we have evaluated our method on the DAVIS2017-motion dataset. We added this experiment in the supplementary material since DAVIS2017-motion was released on Nov.12, 2022, after the initial submission of our paper. DAVIS2017 [1] is an extension of DAVIS2016 dataset that includes additional videos with multi-object contents, resulting in multiple-segment annotations for the ground truth. It contains a total of 90 videos, split into 60 for training and 30 for validation. DAVIS2017-motion is a curated version of the DAVIS2017 dataset performed by the authors of [2] for a fair evaluation of motion segmentation based on flow information only, where connected objects sharing common motion are merged in the ground truth of the validation test.

Method / Scores	$\mathcal{J}\&\mathcal{F}\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$
Ours	42.0	38.8	45.2
MoSeg	35.8	38.4	33.2
OCLR	55.1	54.5	55.7

Table 3. Comparative evaluation on the DAVIS2017-motion validation set. The Jaccard index $\mathcal{J}$ expresses the correct overlap (intersection over union) between the extracted segments and the ground truth, while $\mathcal{F}$ focuses on segment boundary accuracy (the higher the better). $\mathcal{J}\&\mathcal{F}$ is the mean of the two. Evaluation is performed on the video as a whole, and reported scores are the average of the individual video scores.

We evaluated our method on the validation set using the official DAVIS-2017 evaluation algorithm that involves a Hungarian matching process. Results are collected in Table 3. On this dataset, our method has a better $\mathcal{J}\&\mathcal{F}$ score than MoSeg [3] (42.0 *vs* 35.8), while OCLR flow-only [2] outperforms both (55.1), but OCLR is trained using synthetic data whose generation involves human annotation.

3.4. Differentiation of motion patterns

Network output may be limited to three segments with $K = 4$, not due to the model itself but to the train set. Indeed, DAVIS2016 train set comprises too few videos with several moving objects. We have noticed in other applica-

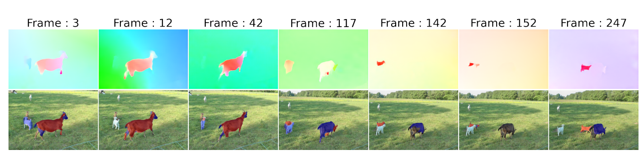

Figure 6. Different instants of "goats1" video of FBMS59. First row: input optical flow (HSV color code). Second row: motion segments predicted by the network (before applying the global temporal linkage), and superimposed on the video frame (except background mask); when static, goats are merged with backgound.

tions that the network, trained on a dataset involving many moving objects, is producing a number of segments equal to the specified K. Besides, we are not dealing with instance segmentation, but with motion segmentation into layers. Accordingly, objects with the same motion are prone to belong to the same mask (layer), but the decomposition into connected segments could be an easy postprocessing. On the other hand, our method manages to separate objects with different motions, e.g., two cars or two animals in Fig.3 of the main paper. In addition, results (with $K = 4$) on the "goats1" video are reported in Fig.6.

4. Repeatability

In order to evaluate the reliability of our method, we repeated five times the training of our model with five different initialisations and performed the abovementioned evaluation pipeline.

Experiment	DAVIS Val	SegTrackV2	FBMS	Davis 2017
1	73.2	55.0	59.4	42.0
2	73.9	57.6	59.0	39.0
3	72.6	55.1	59.5	40.8
4	73.9	56.5	60.5	38.8
5	72.7	55.2	59.1	41.0
Average	73.3	55.9	59.5	40.3

Table 4. Results ($\mathcal{J}\&\mathcal{F}$ for Davis2017-motion and Jaccard index $\mathcal{J}$ for the three others) of five experiments, involving different initialisations of the network, on the four datasets. Reported results in the main paper correspond to experiment 1.

We can observe that the results collected in Table 4 are globally stable, whereas the process described above (segment linkage and segment selection) could generate variability.

5. Latent motion representation

We give a few highlights on the latent motion representation issued from the trained network as mentioned in the main paper (Section 3.4). We carried out a preliminary experiment directly based on the normalized latent vectors of all the sites of a subsequence. The latent vectors are of dimension 32. We applied a PCA procedure to all the latent vectors over the whole subsequence of length T and taking into account the triplets at each time instant. These latent vectors are stacked as an array of dimensions $(3 \times H \times W \times T, 32)$, where H and W are respectively the height and the width of each frame of the subsequence.

Then, we compute the softmax of the projections onto the three first components of the PCA output. Interestingly, after thresholding the softmax values (threshold value of 0.7), we observe that the resulting map is likely to provide a binary segment close to the ground truth of the primary moving object, as illustrated in Fig.7. It shows that our latent motion representation is not only informative in its own, but more importantly, is coherent over the subsequence since the PCA is computed once over the subsequence. In future work, we will investigate further this possibility to provide a binary segmentation directly oriented to the VOS evaluation, when our network is trained for multiple motion segmentation with K masks.

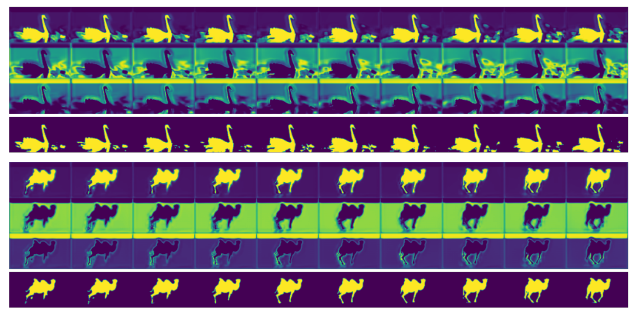

Figure 7. Illustration of the principal component analysis of the latent motion representation. Two examples from DAVIS2016: blackswan and camel. For each example, the first three rows are the projection of the latent vectors on the three first principal components. The fourth row is the binary segmentation obtained by thresholding the projection on the first component.

6. Detailed results per videos of the datasets

Hereafter, we report detailed results through tables collecting the evaluation scores obtained by our method for every video of the four datasets, DAVIS2016, SegTrackV2, FBMS59, and DAVIS2017-motion. Let us recall that the official evaluation algorithm is not the same for DAVIS2016 and DAVIS2017. The evaluation is done in one go on the whole video for DAVIS2017, while it is achieved frame by frame of the video for DAVIS2016.

6.1. DAVIS2016

Video	$\mathcal{J}$ (M)	$\mathcal{J}$ (O)	$\mathcal{J}$ (D)	$\mathcal{F}$ (M)	$\mathcal{F}$ (O)	$\mathcal{F}$ (D)
blackswan	0.584	0.833	-0.11	0.594	0.875	-0.072
bmx-trees	0.597	0.756	0.198	0.798	0.949	0.095
breakdance	0.738	0.976	0.004	0.738	0.988	-0.007
camel	0.871	1	0.12	0.859	1	0.128
car-roundabout	0.918	1	-0.026	0.828	1	-0.068
car-shadow	0.897	1	0.019	0.846	1	-0.011
cows	0.873	1	0.031	0.804	1	0.022
dance-twirl	0.821	1	-0.071	0.853	1	-0.022
dog	0.812	1	-0.044	0.709	0.931	-0.024
drift-chicane	0.664	0.8	-0.221	0.754	0.84	-0.069
drift-straight	0.861	1	0.046	0.786	0.938	0.245
goat	0.298	0	0.125	0.299	0.023	0.037
horsejump-high	0.821	1	0.097	0.87	1	0.044
kite-surf	0.424	0.354	0.249	0.41	0.208	0.087
libby	0.734	0.979	0.117	0.846	1	-0.002
motocross-jump	0.61	0.553	0.182	0.377	0.474	0.198
paragliding-launch	0.624	0.667	0.313	0.314	0.167	0.375
parkour	0.735	0.959	0.069	0.777	1	0.15
scooter-black	0.861	1	-0.023	0.739	1	0.106
soapbox	0.889	1	0.03	0.859	1	0.021
Average	0.732	0.844	0.055	0.703	0.82	0.062

Table 5. Results given for every video of DAVIS2016 dataset. Reported scores per video are the average Jaccard score over frames in the video. The very last row is the average over videos scores. $\mathcal{J}$ is the Jaccard index and $\mathcal{F}$ is the Countour Accuracy. The Mean (M) is the average of the scores, the Recall (O) is the fraction of frames per video with a score higher than 0.5, and the Decay (D) is the degradation of the score over time in the video.

6.2. SegTrackV2

Video	Jacc ($\mathcal{J}$)
Bird of paradise	51.5
birdfall	38.1
bmx	76.9
cheetah	44.1
drift	33.0
frog	78.2
girl	59.8
hummingbird	68.7
monkey	53.9
monkeydog	16.6
parachute	92.9
penguin	39.4
soldier	64.0
worm	39.3
Frames. Avg	55.0

Table 6. Results given for every video of SegTrackV2 dataset. Each reported score is the average Jaccard score over annotated frames in the video. The very last row is the average over all the frames and over all the videos.

6.3. FBMS59

Video	Jacc ($\mathcal{J}$)
camel01	27.8
cars1	88.1
cars10	54.1
cars4	83.6
cars5	83.7
cats01	71.1
cats03	79.9
cats06	38.9
dogs01	73.1
dogs02	66.1
farm01	79.1
giraffes01	36.2
goats01	45.5
horses02	65.3
horses04	72.4
horses05	43.9
lion01	50.7
marple12	61.3
marple2	64.3
marple4	77.8
marple6	51.0
marple7	58.0
marple9	66.8
people03	52.8
people1	80.1
people2	87.3
rabbits02	49.8
rabbits03	41.3
rabbits04	50.2
tennis	72.6
Frames. Avg.	59.4

Table 7. Results given for every video of FBMS59 dataset. Each reported score is the average Jaccard score over annotated frames in the video. The very last row is the average over all the annotated frames and over all the videos.

6.4. DAVIS2017-motion

Sequence	J-Mean	F-Mean
bike-packing_1	0.072	0.370
bike-packing_2	0.276	0.393
blackswan_1	0.577	0.593
bmx-trees_1	0.520	0.766
breakdance_1	0.365	0.558
camel_1	0.716	0.683
car-roundabout_1	0.900	0.814
car-shadow_1	0.870	0.804
cows_1	0.778	0.675
dance-twirl_1	0.441	0.641
dog_1	0.481	0.456
dogs-jump_1	0.433	0.506
dogs-jump_2	0.018	0.174
dogs-jump_3	0.190	0.251
drift-chicane_1	0.381	0.562
drift-straight_1	0.811	0.739
goat_1	0.275	0.303
gold-fish_1	0.175	0.270
gold-fish_2	0.027	0.325
gold-fish_3	0.168	0.209
gold-fish_4	0.000	0.000
gold-fish_5	0.000	0.000
horsejump-high_1	0.562	0.783
india_1	0.057	0.066
india_2	0.047	0.110
india_3	0.143	0.186
judo_1	0.247	0.357
judo_2	0.417	0.522
kite-surf_1	0.422	0.418
lab-coat_1	0.337	0.313
libby_1	0.730	0.847
loading_1	0.166	0.226
loading_2	0.068	0.210
loading_3	0.407	0.416
mbike-trick_1	0.644	0.683
motocross-jump_1	0.509	0.481
paragliding-launch_1	0.339	0.237
parkour_1	0.682	0.753
pigs_1	0.253	0.394
pigs_2	0.019	0.313
pigs_3	0.369	0.432
scooter-black_1	0.856	0.750
shooting_1	0.575	0.500
soapbox_1	0.726	0.779

J&FMean	JMean	JRecall	JDecay	FMean	FRecall	FDecay
0.420	0.388	0.365	0.006	0.452	0.454	0.039

Table 8. Results given for every video of DAVIS2017-motion dataset. The very last row is the average score over all the videos for the different criteria.

References

[1] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 Davis challenge on video object segmentation. *arXiv:1704.00675*, 2017.

[2] J. Xie, W. Xie, and A. Zisserman. Segmenting moving objects via an object-centric layered representation. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[3] C. Yang, H. Lamdouar, E. Lu, A. Zisserman, and W. Xie. Self-supervised video object segmentation by motion grouping. In *International Conference on Computer Vision (ICCV)*, October 2021.