Representing Objects in Video as Space-Time Volumes by Combining Top-Down and Bottom-Up Processes

Filip Ilic Axel Pinz
Graz University of Technology
{filip.ilic, axel.pinz}@tugraz.at

Abstract

As top-down based approaches of object recognition from video are getting more powerful, a structured way to combine them with bottom-up grouping processes becomes feasible. When done right, the resulting representation is able to describe objects and their decomposition into parts at appropriate spatio-temporal scales. We propose a method that uses a modern object detector to focus on salient structures in video, and a dense optical flow estimator to supplement feature extraction. From these structures we extract space-time volumes of interest (STVIs) by smoothing in spatio-temporal Gaussian Scale Space that guides bottom-up grouping. The resulting novel representation enables us to analyze and visualize the decomposition of an object into meaningful parts while preserving temporal object continuity. Our experimental validation is twofold. First, we achieve competitive results on a common video object segmentation benchmark. Second, we extend this benchmark with high quality object part annotations, DAVIS Parts[1]*, on which we establish a strong baseline by showing that our method yields spatio-temporally meaningful object parts. Our new representation will support applications that require high-level space-time reasoning at the parts level.*

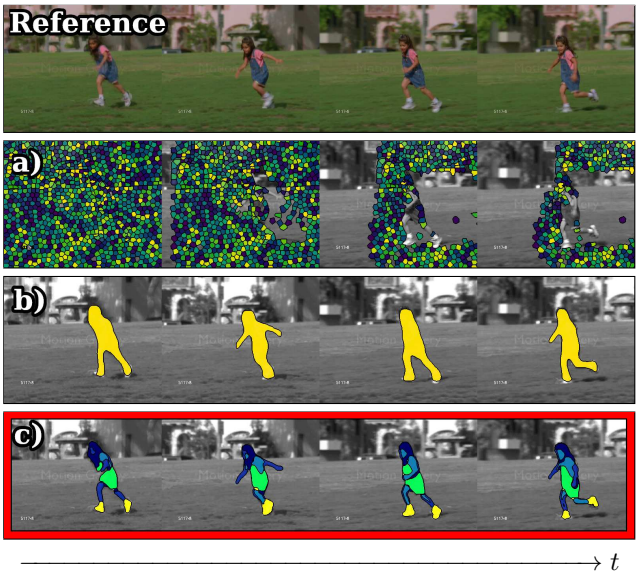


Figure 1. Our approach (c) bridges the conceptual gap between low-level video representations such as (a) TSPs [4] which tend to fall apart over time and do not model objects directly, and instance segmentation methods such as (b) Mask R-CNN [10] which lack a decomposition into parts, and temporal consistency. Our unsupervised approach is able to detect salient object parts based on motion and appearance. Sequence taken from SegTrack v2 [16].

1. Introduction

Recent research has achieved a lot of progress in Video Object Segmentation, both, with respect to the actual segmentation accuracy in each frame, but also with respect to the temporal consistency of the tracked object. A neglected sub-task is the separation and decomposition of generic objects into their salient components, where object parts can often be inferred by their difference in motion or appearance w.r.t. their surroundings.

The core novelty of our method is the principled *local* use of spatio-temporal Gaussian Scale Space. The enabling top-down methods include a generic instance segmentation framework to detect objects of interest, and an optical flow estimator to track the object and to extract motion information. These top-down anchors are used to guide the bottom-up grouping of features, extracted from appearance and motion in Gaussian Scale Space. This generates consistent regions, which correspond to meaningful object parts. See Fig. 1(c) as to how our approach models different parts of a running girl with different spatio-temporal tubes (same color means same tube). Compare this with Fig. 1(b), where each object instance is modeled by a singular object tube, and Fig. 1(a) where Temporal Superpixels (TSPs) provide no spatial grouping, and fall apart over time. Our approach

[1] Available: http://f-ilic.github.io/STVI

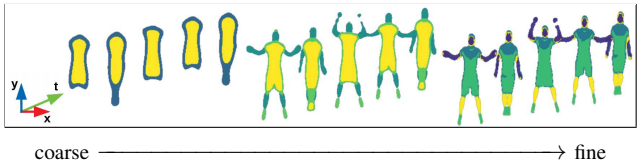


Figure 2. STVIs extracted at increasing spatial and temporal scales from a jumping-jack sequence. Note how finer scales capture more detail at the cost of temporal and spatial coherence, whereas coarser scales model more fundamental patterns. Two scales have been selected manually, the representation in the middle is achieved by automatic scale selection.

is able to extract the parts of objects from video in a completely unsupervised manner, including object localization, tracking, and automatic selection of spatio-temporal scales. If required, manual scale selection allows for the explicit extraction of low or high frequency features from motion and appearance structures, see Fig. 2.

Such a general representation at the object and parts level could be highly desired in many applications of video understanding, e.g. to analyze motion patterns in sports, to classify actions, or to describe complex dynamics in cluttered scenes with many moving objects. To the best of our knowledge, there is no comparable approach that segments individual objects from video into their parts based on their spatio-temporal saliency.

2. Related Work

Frame-based object proposals, e.g. the region proposal network (RPN) [21] provide bounding boxes that inevitably contain background. With the emergence of high quality datasets for instance segmentation [17, 16, 29], methods such as Mask R-CNN [10] improve on those approaches by not only providing tight bounding boxes but also the segmentation mask. However, these masks tend to be imprecise at the object boundaries and temporal correspondence is hard to obtain, especially when partial occlusions occur; therefore, simple temporal stacking of masks from individual frames does not suffice to represent objects in video reliably. Approaches that inherently work on video data, such as SiamMask [27] are able to generate temporally consistent object segmentations through utilizing a Siamese network structure for matching/tracking objects in adjacent frames [3, 15], only requiring the initial object bounding box. Such methods, however, still yield monolithic segmentation masks where the notion of individual salient object parts is not incorporated.

Many excellent solutions exist for object recognition, as well as two-stream architectures for video analysis that use appearance and motion information [6, 7]. These approaches achieve close to human performance in object and action recognition, and object tracking. The stunning performance, however, comes at a price: Besides the huge effort required to train these networks, what they learn is represented implicitly in the millions of parameters that are tuned and thus make it exceedingly difficult to build explicit reasoning on top of them.

One main benefit of our method is that it can recover salient object parts in an unsupervised manner via bottom-up grouping processes. Varquez et al. [26] propose a multiple hypothesis video segmentation algorithm that operates on superpixel flow, in which different superpixelations are created from which overlapping and persistent regions are matched. This indicates that these regions must form some sort of salient area, and are therefore spatio-temporally significant structures. Generally, superpixel/voxel algorithms, such as Simple Linear Iterative Clustering (SLIC) [1, 2], are used to abstract individual pixels into larger groups to reduce the complexity of visual tasks [22]. Temporally Consistent Superpixels (TCS) [23] and Temporal Superpixels (TSP) [4] are for instance such approaches that leverage superpixels created by clustering in the appearance-space. These are considererd low-level video representations as they extend image based superpixels to the video domain, modeling the entire video-volume. Therefore, neither objects nor their motion are modeled explicitly in such representations. While TCS and TSP consider appearance, Levinshtein et al. [14] work with optical flow and extract spatio-temporally closed regions (STC), which is an important step to ensure a temporally consistent representation. The aforementioned approaches are impressive in their own right, but they do not operate at the object level, and instead describe the whole scene with spatio-temporal structures. This gap between object representations and lower-level video representations exemplifies the current void that we aim to fill, and which we show in Fig. 1.

We deem the notion of scale, i.e. levels of detail at which an object is represented, an essential aspect of a good representation. We employ the well researched Spatio-temporal Gaussian Scale Space [12, 18] to achieve our goal of variable object scale. In essence, videos at different scales can be obtained by smoothing the video-volume with Gaussians of varying σ in space, and τ in time. Our method extends scale-space to the object level to estimate appropriate spatial and temporal scales at which an object is represented and decomposed.

Early work conducted by Gorelick et al. [9] demonstrated the feasibility of extracting motion-tracks of objects from clean video, and using them to classify human actions such as jumping, running, and walking. There are even object-based spatio-temporal representations [25] that, similar to our approach, use object detectors as "top-down anchors". Other approaches such as [8] use optical flow to anchor an object. However, the distinguishing feature and novelty of our approach is that we are able to repre-

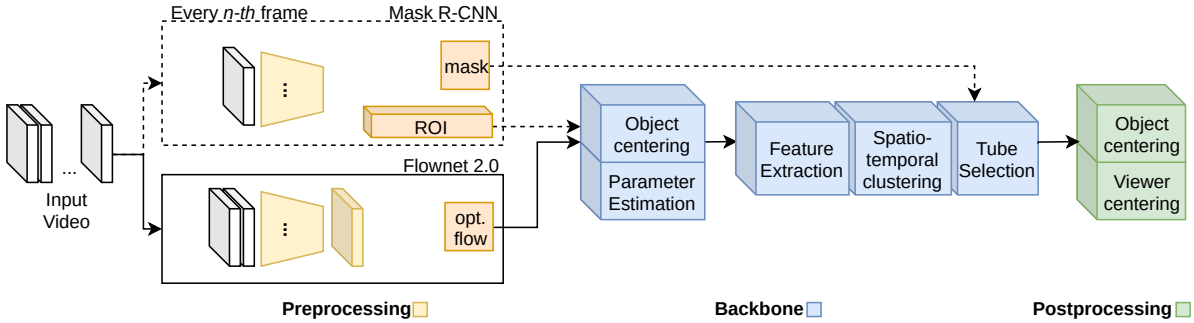


Figure 3. The pipeline of our approach is split into three distinct parts. The preprocessing uses an off-the-shelf object detector and a dense optical flow estimator; the per object ROI is used to crop appearance and flow around the current object of interest which results in an object centered subvolume. The backbone then generates spatio-temporally salient structures by smoothing in spatio-temporal scale space with parameters estimated by the object size and motion. Groups of coherent spatio-temporal features are then selected by utilizing mask predictions every *n-th* frame. This naturally tends to yield spatio-temporally significant regions in the context of the object. Hence, our method is able to handle multiple salient objects per video by running multiple times, with almost no overhead, as most of the computational complexity resides in the feature extraction, which can be shared among the objects.

sent salient objects in a decomposable, well-behaved, and scalable manner. Seguin et al. even note that *"most of the visual unpleasant artifacts"* [of their approach] *"are due to the use of superpixels"* ([25] caption Fig. 4); this observation holds true for most of the bottom-up driven grouping processes. We aim to avoid such artifacting, and to produce smooth space-time volumes which render them feasible for an incorporation in other applications.

3. Space-Time Volumes of Interest

Our approach uses temporally sparse instance segmentation masks and optical flow to build a representation of salient objects and their components, by locating and extracting spatio-temporally salient regions. These regions form structures that resemble *tubes* which model the spatial and temporal extent of objects and their components over time. Instance masks are used to focus on the objects of salience - akin to an attention mechanism, which we refer to as "anchoring". These masks are provided by Mask R-CNN [10], trained on COCO [17]. Dense optical flow provided by FlowNet2.0 [11] is used to track the object until the next anchor mask is supplied. It is also used to cancel background motion and to support the generation of spatio-temporal tubes, which are extracted by clustering appearance and motion information on the anchored object. The pipeline of our proposed approach is shown in Fig. 3.

3.1. Preprocessing

Mask R-CNN is used in the first frame to detect objects of interest, and provides us with an initial Region of Interest (ROI). Because we evaluate our approach on datasets that only contain one annotated object, we restrict our approach to that one object per video, though it is capable of handling multiple objects of interest per video by running multiple times. Dense optical flow is computed for every frame pair to track the object, and will later be used to ensure temporal consistency. Every n frames another ROI is supplied to account for drift over time. By stacking the tracked ROIs we create a sub-volume aligned along each of the ROI's center points, and padded to fit to the dimensions of the largest ROI in the video. This yields an object-centered sub-volume around the object.

Because optical flow deals with apparent motion in video, one cannot infer the motion of objects in the scene, but only the relative motion between scene, camera and homogeneously moving regions. We are only interested in representing salient objects, and therefore cancel out the optical flow w.r.t. the object of interest. We do this by subtracting the background's optical flow:

$$\hat{F}_i = \left[F_i - mean(F_i \odot \neg M_i)\right]_{obj}, \tag{1}$$

where $\left[\,\cdot\,\right]_{obj}$ denotes the cropping of the volume to the sub-volume around the salient object, F_i the optical flow at frame i, M_i the binary segmentation mask at frame i, and $\odot$ the element-wise multiplication.

The preprocessing step results in two sub-volumes of identical extent (x,y, and t). The first one contains appearance whereas the second one contains the object flow, $\hat{F}_x$ and $\hat{F}_y$. These two sub-volumes are combined, such that each point in the sub-volume holds 8 values: position (x, y, t), color (L, a, b) in the Lab color-space, and flow $(\hat{F}_u, \hat{F}_v)$. With the preprocessing complete, c.f. Fig. 3, we proceed with the extraction of salient tubes.

3.2. Backbone

Spatio-Temporal Smoothing The 8 dimensional feature-volume is processed at the object level, in a bottom-up fashion. To facilitate the creation of connected intra-frame

structures of an object, we smooth spatially; to create structures that are temporally consistent and represent the same region even through slight changes in appearance or partial occlusions, we smooth temporally. The combination of these smoothing operations provides a means to weight and emphasize certain object- and motion-patterns in the sub-volume. Gaussian Scale Space with its extension to the spatio-temporal domain and the notation of a space-time scale space family $\mathscr{L}$ as proposed by Laptev and Lindeberg [13] is a cornerstone of our approach:

$$\mathscr{L}(\cdot;\sigma^2,\tau^2) = g(\cdot;\sigma^2,\tau^2) * f(\cdot), \tag{2}$$

where the input f is convolved with a spatio temporal Gaussian filter g:

$$g(x,y,t;\sigma^2,\tau^2) = \frac{1}{\sqrt{(2\pi)^3\sigma^4\tau^2}} \cdot e^{\frac{-(x^2+y^2)}{2\sigma^2} - \frac{t^2}{2\tau^2}}. \tag{3}$$

Here σ determines the width of the spatial kernel (same for x, y direction), and τ the width along the temporal dimension. With large σ, τ, only lower-frequency spatio-temporal structures remain; convolution with small σ, τ preserves the higher-frequency structures. While it is well-known that particular scales globally emphasize particular space-time structures, we explore the power of scale space *locally* to support the decomposition of objects into meaningful components.

Grouping spatio-temporally salient regions Following smoothing, we cluster the sub-volume containing appearance and optical flow information, encoded as 8-dimensional vectors, see Eq. 4. We apply SLIC clustering to the x,y,t sub-volume which contains the feature vector, $\phi(x,y,t)$, at each location:

$$\phi(x,y,t) = \begin{bmatrix} \alpha(x,y,t)^T \\ (L,a,b)^T \\ \beta(\hat{F}_u,\hat{F}_v)^T \end{bmatrix} \text{ dimensionality: } 8\times 1 \ . \tag{4}$$

The scalar values α, β allow us to stretch and compress the sub-volume in which SLICs are generated. High values of α create more compact regions by prioritizing spatial proximity, whereas β is a trade-off between image intensities and flow components, with higher values prioritizing the flow component. We do not enforce spatial connectivity during clustering. Since we work in the 2D projection of 3D data, objects that are connected in 3D might not appear to be in 2D. This more general approach results in tubes that might appear disconnected in individual frames, but are connected somewhere in the temporal sequence. Later additional reasoning could be added for splitting or merging individual tubes.

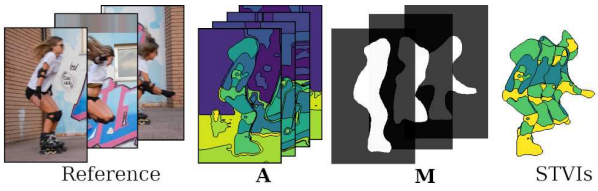


Figure 4. We select tubes from the clustered spatio-temporal features **A**, by intersecting individual tubes with the instance masks **M**, which are extracted every n frames. The remaining tubes (STVIs) model the object and its parts. Lower values of n improve the quality (more instance masks), but also incur a larger run-time penalty. In this work we choose n=20.

Tube Selection The smoothed and clustered volume, **A**, contains clustered spatio-temporal features that we call tubes (v_i), i.e. $\{v_1, ..., v_n\} \in \mathbf{A}$. In most cases background clutter is present, making it necessary to discard some tubes.

We do this by using instance masks from Mask R-CNN every n-th frame, referred to as **M**. We compute the overlap of each tube with **M** and add it to the set of STVIs if more than a certain (volumetric) threshold, ω, is contained. This guarantees that the selected structures are temporally coherent, because they are contained within the instance masks over many frames, see Fig. 4.

In our experiments, we verified that tube selection is insensitive to the threshold parameter ω if it is chosen within $0.5 \leq \omega \leq 0.9$. This can be attributed to the short videos that we work with (usually around 80 frames), where single tubes are able to capture and track parts of objects through the whole sequence. In all our experiments reported, we set $\omega = 0.7$. As video length increases, ω needs to be decreased, to allow for tubes that approximate the object well through one section of the video but do not persist over the entirety of the video. Another possible way to deal with longer videos is to split them into shorter chunks. The chunks would then present good short-term volumes, which would need to be merged. We leave this for future work.

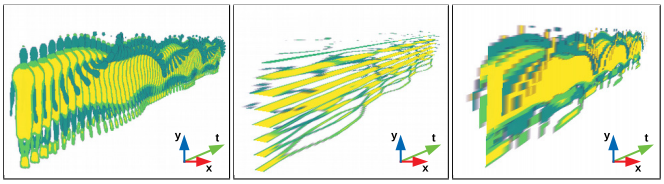
Parametrization The parameters to process a video can be estimated based on the appearance and motion of the object in the video alone. This means that our approach can be run without human interaction (0-Parameter), which is what we use for evaluating our method. However, the parameters can be fine tuned based on the application; e.g. reducing the temporal scale for high-frame-rate footage, or forcing coarser spatial scales. Ablation experiments in which we show the influence of each mentioned parameter are performed in Section 4.

- **The spatial scale** σ determines the amount of spatial smoothing that is applied, and is symmetric in the x, y direction. It is set proportial to the object's size, and defaults to the radial approximation, i.e., half of the diagonal of the average bounding box around the object.

- **The temporal scale** τ determines smoothing along the temporal axis and is proportional to the maximum object flow. By default we try to capture all the motion of the object, and therefore set the temporal scale according to the largest object motion. However, to be robust to outliers, the 95^{th} percentile of $||\hat{F}||$ is used.
- **The two clustering coefficients** α, β as introduced in Eq. 4 can be used to prioritize either the color- or the flow-components. By default they are set proportional to the spatial and temporal scales ($\alpha \propto \sigma, \beta \propto \tau$).
- **The maximum number of components** k determines into how many spatio-temporal tubes the object may be decomposed at most. For our experimental evaluation, k is set to 15, which is suitable for common object categories, as demonstrated in Section 4.

3.3. Postprocessing

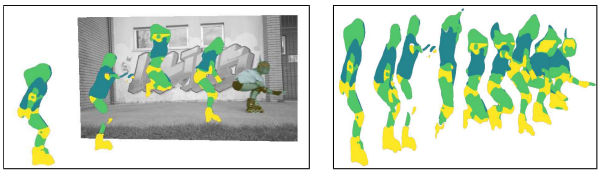
STVIs are visualized as sliced tubes that, when taken together, model the space-time extent of an object. In contrast, other approaches [9, 30], show object- and motion-tracks as 3D meshes, only showing the hull of the space-time object segmentation. Our visualizations show the different parts of an object, and are especially useful because they allow for a slicing along any of its dimensions, emphasizing motion patterns by looking at different cross-sections, see Fig. 5.



(a) t-slices (b) y-slices (c) x-slices

Figure 5. Interesting patterns revealed by slicing along any of the STVI's dimensions; e.g. the oscillating motion of the legs in (b).

We can also switch between object-centered and viewer-centered perspective, see Fig. 6. Since we obtain individual object parts, such an object centered perspective can reveal relative motion w.r.t. the objects' centroid.



(a) viewer-centered (b) object-centered

Figure 6. The trajectory w.r.t. the observer is shown in (a). The relative motion of object parts w.r.t. the object centroid is shown in (b). This is useful *because* objects are represented as the sum of their parts, and not just as instance masks. For instance, notice the roller blades that move towards the object centroid during the jump.

4. Experiments & Results

Our experimental validation is split into two major parts: 1) Conventional Video Object Segmentation on the DAVIS [20] dataset, which provides excellent ground truth masks, using well established metrics. In addition, we provide a detailed ablation study.
2) An evaluation of the automatic detection of object parts, for which we extend DAVIS with high-quality, pixel accurate segmentations masks - DAVIS Parts. This new database it is to our knowledge the first Video Object Part Segmentation dataset with pixel accuracy. We evaluate all competing methods on this dataset with metrics that allow us to establish a first baseline for unsupervised object part detection and segmentation.

Video Object Segmentation - Evaluation The metrics we use to evaluate spatio-temporal object representations have to cover a wide range of characteristics. The obvious candidates are segmentation metrics. We also choose to include metrics that quantify interframe contour consistency:

- 3D Segmentation Accuracy **(ACC)**, and Undersegmentation Error **(UE)**, measuring the fraction of correctly identified pixels belonging to the object, and the fraction of pixels extending past the object boundary, respectively, introduced in [28]
- Normalized Temporal Extent **(TEX)** introduced in [4], measuring how long the generated tubes persist in time.
- The Jaccard index $\mathcal{J}$ [5] measuring region similarity and the F-measure $\mathcal{F}$ [19, 20] measuring contour accuracy.

Because lower-level representation approaches such as as TSP [4], TSC [23], SLIC [2] and STC [14] model the whole video volume and not only the objects, a slight adaptation is needed to select appropriate supervoxels which correspond to the objects. We employ the same tube selection process as described in Section 3.2, where we use an intersection criterion w.r.t. the Mask R-CNN segmentation to decide which individual supervoxels are selected for the final object representation. For the evaluation we consider two ways to select the tubes: using masks 1) in the first frame, and 2) every 20 frames, in the following denoted by the ⚓ symbol. We also include orthogonal - currently more popular - approaches that do not rely on explicit bottom-up grouping processes, to put the obtained results into perspective: Mask-RCNN [10] which yields framewise full object masks and bounding boxes and SiamMask [27] which yields temporally consistent object segmentations. Note that these methods are unaware of part decomposition.

Table 1 shows the evaluation of the methods on the DAVIS [20] benchmark dataset. We consider Mask R-CNN

			ACC ↑	UE ↓	TEX ↑	$\mathcal{J}$ ↑	$\mathcal{F}$ ↑
MRCNN [10]			0.82	0.29	—	0.62	0.63
SiamMask [27]			**0.86**	0.28	—	**0.69**	0.67
SLICO [2]			0.30	0.10	0.69	0.27	0.24
		⚓	0.33	0.12	0.71	0.29	0.25
TSP [4]	k_{100}		0.32	0.23	0.62	0.27	0.28
	k_{100},	⚓	0.44	0.12	0.46	0.41	0.37
	k_{800}		0.38	0.05	0.46	0.36	0.37
	k_{800},	⚓	0.63	**0.03**	0.39	0.61	0.60
Ours			0.58	0.15	0.87	0.46	0.56
		⚓	0.78	**0.03**	**0.93**	0.66	**0.68**

Table 1. Our method provides competitive results across the board on the DAVIS dataset, and sets a very strong baseline w.r.t. TEX. Our method also achieves the lowest Undersegmentation Error, at the cost of a slightly decreased Accuracy. Other methods that perform high w.r.t. Accuracy tend to generate instance masks that are coarse and blob like.

trained on COCO [17], a single frame segmentation baseline. TEX is left blank, because the generated frame segmentations are not temporally connected. SiamMask yields temporally connected segmentation masks, however, because it does not provide object components, but rather a single instance segmentation, the temporal extent of the single tube representing the whole object is rather meaningless, and also left blank. SLICO, the 0-Parameter SLIC is also tested as a simple baseline. TSPs, a more sophisticated bottom-up video representation are evaluated in variations k_{100} and k_{800} which refers to the free hyper parameter k which determines the number of components per frame.

Our results show that our approach establishes a very strong TEX score of 0.93 outperforming the next best approach by a considerable margin of 22 percentage points. TSPk_{800} and our method achieve a stunning UE of only 0.03, almost an order of magnitude better than Mask R-CNN and SiamMask. This large UE of Mask R-CNN - indicating that larger than necessary masks for objects are created - also tends to result in a higher ACC, as it is more likely to cover the object, if the mask is larger in the first place.

We group all 50 DAVIS videos into five "meta-classes", i.e. Human, Animal, Bike-like, Car-like, and Miscellaneous. Class specific metrics are shown in Table 2, and provide insights into how the algorithms differ: The UE of the class "Bike-like" is considerably higher for Mask R-CNN and SiamMask than for any other class. This can be attributed to the data that both of these networks were trained on, which causes them to generate blob-like masks that segment whole bike wheels as discs. Our approach, on the other hand, does not have this tendency, but rather generates slightly too small regions, because of the non-edge preserving spatial and temporal smoothing that is applied.

			Human	Animal	Bike-like	Car-like	Misc.
	# Videos		15	15	7	8	5
ACC↑	MRCNN [10]		**0.85**	0.87	0.86	**0.84**	0.86
	SiamMask [27]		0.82	**0.90**	**0.88**	**0.84**	**0.89**
	TSP [4] k_{800}	⚓	0.61	0.68	0.50	0.66	0.65
	Ours	⚓	0.78	0.77	0.79	0.79	0.85
UE↓	MRCNN [10]		0.22	0.23	0.44	0.19	0.37
	SiamMask [27]		0.19	0.29	0.46	0.23	0.40
	TSP [4] k_{800}	⚓	**0.03**	0.04	0.05	**0.02**	**0.03**
	Ours	⚓	0.04	**0.03**	**0.01**	0.03	**0.03**
TEX↑	TSP [4] k_{800}	⚓	0.36	0.49	0.20	0.41	0.39
	Ours	⚓	**0.87**	**0.94**	**0.75**	**0.81**	**0.90**
$\mathcal{J}$ ↑	MRCNN [10]		0.65	0.65	0.50	0.59	0.62
	SiamMask [27]		**0.69**	**0.72**	0.64	**0.69**	0.64
	TSP [4] k_{800}	⚓	0.59	0.66	0.48	0.64	0.63
	Ours	⚓	**0.69**	0.68	**0.66**	0.64	**0.77**
$\mathcal{F}$ ↑	MRCNN [10]		0.64	0.70	0.56	0.54	0.63
	SiamMask [27]		0.70	**0.72**	0.63	**0.62**	0.58
	TSP [4] k_{800}	⚓	0.56	0.64	0.54	0.61	0.66
	Ours	⚓	**0.71**	0.69	**0.68**	0.60	**0.72**

Table 2. Detailed per class performance of the tested approaches.

Video Object Segmentation - Ablation Study During ablation, Fig. 7, one parameter is subject to changes within a certain range, while the others are selected automatically, see Section 3.2.

We observe that increasing σ has the effect of decreasing ACC, decreasing TEX, and no change in UE. This is in line with our intuition that spatial low frequency tubes are worse at segmentation and tracking. Nonetheless, they enable us to represent complex objects at coarser scales with fewer and smoother tubes. Coarser temporal scales (larger τ) on the other hand correspond to slightly higher TEX, and considerably lower ACC, and no change in UE. As larger temporal smoothing is applied, fine spatial structures are lost and cannot be recovered.

The compactness of the tube, determined by α, see Eq. 4, does not play a significant role in the segmentation performance. It rather influences the qualitative appearance of the generated tubes; in most instances of the videos there is enough contrast in appearance or optical flow of the salient object. Changes w.r.t. β, on the other hand, do have a notable impact; larger values correspond to increasing ACC and decreasing TEX, as the generated tubes incorporate more optical flow information which can cause a rupture in a motion tube; e.g. a back and forth motion is split into two tubes (forward - backward) based on the flow.

k, which is responsible for the number of generated tubes, is more sensitive to change: As more tubes are used to represent the object, they can more closely model spatio-temporal structures, which leads to increasing ACC. This in turn means that smaller, higher frequency structures are modeled which are more likely to disappear over time again, leading to the observed decreasing TEX.

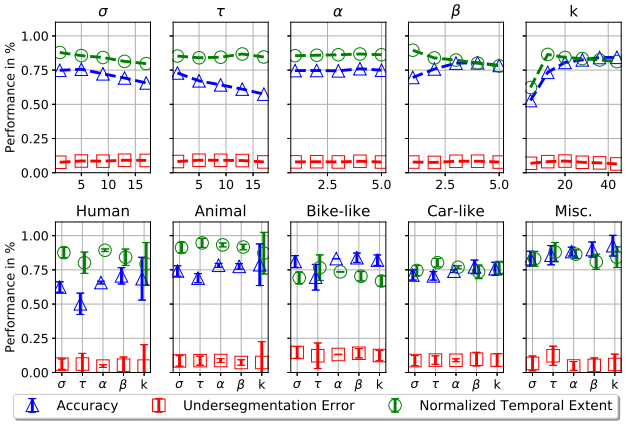


Figure 7. Parameter ablation: the top graphs show the ablation w.r.t. the individual parameter, whereas the bottom graphs show the sensitivity of the parameters w.r.t. all classes in DAVIS. σ and τ, parameters responsible for the scale at which features are extracted show a decrease in performance once a certain threshold is crossed at which objects are represented at too coarse scales, which tends to decrease the segmentation accuracy.

We observe a very constant and low UE. This is, as previously mentioned, inherent to our approach as it tends to select spatio-temporal tubes within object contours, rarely incorporating tubes that exceed object boundaries.

The class variation, Fig. 7 bottom row, shows expected results: more compact object classes tend to have a higher ACC because it is easier to approximate compact blobs than classes with deformable parts, such as humans and animals. These results are in line with our expectations towards a well-behaved spatio-temporally scalable representation which clearly shows the blob-like structures at coarse scales, with finer scales allowing for a more detailed, albeit shorter lived object part representation.

Object Parts - Dataset The major benefit of our method is the automatic decomposition into object parts. To evaluate our claim that the decomposition indeed corresponds to meaningful parts, we extend DAVIS2016 with per-frame, object part annotations. We name this new dataset DAVIS Parts. To our knowledge this is the first dataset providing pixel precision object part segmentation in video. Some example annotations are shown in Fig. 8.

Object Parts - Evaluation In principle, we want to evaluate the $\mathcal{J}$ and $\mathcal{F}$ measures w.r.t. each individual part in the ground truth labels. To enable a fair comparison between the different methods, slight modifications to these measures are required. For each ground truth label we first need to determine which tubes model it. This is done by selecting those tubes that have a significant intersection with the ground truth tube (we set this significance to 30% overlap in our experiments, with this threshold being insensitive to change). This tube selection allows us to compute $\mathcal{J}$ and $\mathcal{F}$ measures, which we additionally normalize by the number of tubes modeling each particular part. This way we can meaningfully compare approaches which create different numbers of tubes. This score per part is averaged over all parts in the ground truth. We name these two modified measures $\mathcal{J}_P$ and $\mathcal{F}_P$.

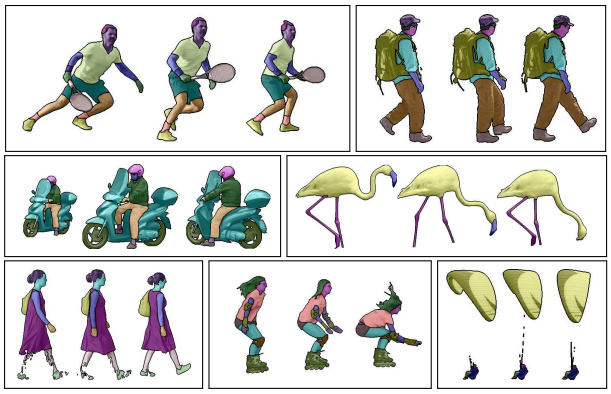


Figure 8. Samples from our new DAVIS Parts dataset, showing three still frames per video overlayed with the annotated parts.

			$\mathcal{J}_P \uparrow$	$\mathcal{F}_P \uparrow$	$\mathcal{C}_P \downarrow$	$\mathcal{M}$
MRCNN [10]			0.05	0.20	—	—
SiamMask [27]			0.02	0.17	—	—
SLICO [2]		↧	0.04	0.18	0.57	16.2
TSP [4]	k_{100}	↧	0.05	0.20	0.29	5.3
	k_{800}	↧	0.06	0.16	0.49	78.8
Ours		↧	**0.16**	**0.24**	**0.21**	3.7

Table 3. Baseline results on our dataset, no other approach has yet tackled the decomposition of arbitrary objects into their meaningful parts, with $\mathcal{M}$ being the average number of components that are created.

Additionally, we introduce a measure for label inconsistency, $\mathcal{C}_P$. Note how in Fig. 9 (left), the dress is modeled by two tubes (light- and dark green). While this results in lower $\mathcal{J}_P$ and $\mathcal{F}_P$ values, we want to quantify the fact that the light- and dark green tubes model a single ground truth object part, throughout the whole video sequence, i.e. the dress always consists of the same tubes. $\mathcal{C}_P$ is calculated by accumulating the absolute deviation from the average distribution of labels at each frame. This score is furthermore normalized by the number of frames. To account for changes in the area of each ground truth label over time, the changes of distribution are measured w.r.t. the percentage of the area of a given label in each frame.

The results of our part based evaluation are shown in Table 3. STVIs outperform all competing approaches, most notably w.r.t. the Jaccard index $\mathcal{J}$. This performance is especially noteworthy considering that the object segmentation experiments, performed in Section 4 showed a much

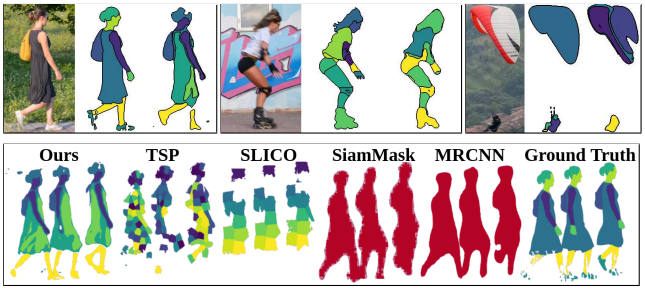


Figure 9. Qualitative comparison of part segmentation results. The top row shows three stills from different videos, each with the reference frame, ground truth, and the result of our method. The bottom row shows the results of different approaches, where the improved temporal consistency and object part detection of our method is evident.

narrower performance gap between the methods. Normalizing the scores by the number of generated tubes is vital to compare the methods in a meaningful way. We therefore also list $\mathcal{M}$, the average number of tubes in Table 3. We can see that our method produces the lowest $\mathcal{M}$ of 3.7 tubes per video, whereas TSP k_{800} produces the most tubes - 78.8. This leads to the observation that large $\mathcal{M}$ tend to incur a high label inconsistency, $\mathcal{C}_P$. This is also in line with the observed results in the object part ablation study below.

Object Parts - Ablation Ablation experiments that measure performance of object part segmentation, Fig. 10, are performed in the same manner, and in the same parameter ranges as those for segmentation performance.

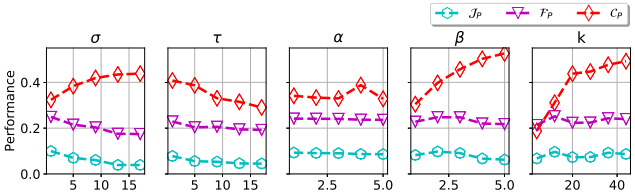


Figure 10. Ablation study results for object parts. We observe similar behaviour as in the object segmentation ablation study, where large σ, τ decrease the performance, whereas α, β are more stable, with large β only incurring larger label inconsistency, $\mathcal{C}_P$. Interestingly, k does not effect $\mathcal{J}_P, \mathcal{F}_P$, as drastically as shown in the object segmentation ablation study.

We observe very gradual changes to $\mathcal{J}_P$, and $\mathcal{F}_P$, across all varied parameters, similar to the ablation results w.r.t. segmentation performance. Changes to $\mathcal{C}_P$ are more sensitive, especially those w.r.t. k. This stems from the fact that increasing the number of generated tubes results in tubes that are more likely to be less stable over time, in a similar fashion to TSPs. We conclude that our proposed method is able to achieve a balance between whole object, and object part segmentation, as it shows to be robust in both areas.

	Object Representation	Temporally Consistent	Decomposable	Variable Scale Spatially	Variable Scale Temporally
MRCNN [10]	✓	✗	✗	✗	✗
SiamMask [27]	✓	✓	✗	✗	✗
SLICO [2]	✗	✓	✓	✗	✗
TSP [4]	✗	✓	✓	✓	✗
Ours	✓	✓	✓	✓	✓

Table 4. Our approach offers a spatially and temporally scalable method for extracting a decomposable object representation unlike monolithic approaches that yield segmentation masks, or more traditional video representations that do not focus on salient objects. Furthermore, no existing object representation offers a temporal scale which supports the creation of spatio-temporal tubes at certain frequencies.

5. Discussion and Conclusion

A qualitative comparison of our results in both segmentation performance and part based segmentation is shown in Fig. 9. We can see that both instance segmentation networks tend to yield masks that are slightly too large, blob-like, do not have crisp object borders, and do not provide object parts. In contrast, our approach yields natural looking object boundaries, and a separation into meaningful parts, where "meaningfulness" is context dependent and established through delineation in appearance and/or motion. Table 4 summarizes key properties of all tested approaches.

We have presented a general method to decompose objects into meaningful spatio-temporal parts using relatively simple features. Our method bridges the gap between bottom-up processes used to build spatio-temporally consistent tubes, and video object segmentation networks that yield instance segmentation masks. This novel object representation at different spatio-temporal scales, unlike other approaches, yields temporally coherent object components delineated by motion and/or appearance. We have evaluated our approach on a common Video Object Segmentation dataset where we achieve competitive results. We furthermore have extended this benchmark with high-quality, pixel level annotated individual object parts, where we set a strong baseline with our approach. We hope that our work of unsupervised detection of object parts in combination with the dataset will inspire development of new methods that are able to automatically detect coherent object parts. In general, our part-based decomposition can enable interesting applications in the realm of reasoning about objects, their interactions, or representations tailored to specific tasks, such as feature extraction from spatio-temporal tubes for action recognition and action localization tasks.

Acknowledgments This work was supported by the ERC starting grant HOMOVIS,No. 640156.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels. Technical Report 149300, EPFL, 2010. 2

[2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012. 2, 5, 6, 7, 8

[3] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision*, pages 850–865. Springer, 2016. 2

[4] J. Chang, D. Wei, and J. W. Fisher III. A video representation using temporal superpixels. In *In Proc. CVPR*, pages 2051–2058. IEEE, 2013. 1, 2, 5, 6, 7, 8

[5] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 5

[6] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016. 2

[7] C. Feichtenhofer, A. Pinz, and A. Zisserman. Detect to track and track to detect. In *Proc. ICCV*, 2017. 2

[8] K. Fragkiadaki, P. Arbelaez, P. Felsen, and J. Malik. Learning to segment moving objects in videos. In *In Proc. CVPR*, pages 4083–4090, 2015. 2

[9] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2247–2253, 2007. 2, 5

[10] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *In Proc. ICCV*, pages 2980–2988. IEEE, 2017. 1, 2, 3, 5, 6, 7, 8

[11] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *In Proc. CVPR)*, volume 2, 2017. 3

[12] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005. 2

[13] I. Laptev and T. Lindeberg. Space-time interest points. In *Proc. ICCV*, 2003. 4

[14] A. Levinshtein, C. Sminchisescu, and S. Dickinson. Spatiotemporal closure. In *Asian Conference on Computer Vision*, pages 369–382. Springer, 2010. 2, 5

[15] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8971–8980, 2018. 2

[16] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg. Video segmentation by tracking many figure-ground segments. In *ICCV*, 2013. 1, 2

[17] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. 2, 3, 6

[18] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Springer, 1994. 2

[19] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (5):530–549, 2004. 5

[20] 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 *In Proc. CVPR*, 2016. 5

[21] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *In Proc. NIPS - Volume 1*, pages 91–99, Cambridge, MA, USA, 2015. MIT Press. 2

[22] X. Ren and J. Malik. Learning a classification model for segmentation. In *In Proc. ICCV*. IEEE, 2003. 2

[23] M. Reso, J. Jachalsky, B. Rosenhahn, and J. Ostermann. Temporally consistent superpixels. In *In Proc ICCV*, pages 385–392, 2013. 2, 5

[24] A. Sauer, E. Aljalbout, and S. Haddadin. Tracking holistic object representations. *arXiv preprint arXiv:1907.12920*, 2019.

[25] G. Seguin, P. Bojanowski, R. Lajugie, and I. Laptev. Instance-level video segmentation from object tracks. In *In Proc. CVPR*, pages 3678–3687. IEEE, 2016. 2, 3

[26] A. Vazquez-Reina, S. Avidan, H. Pfister, and E. Miller. Multiple hypothesis video segmentation from superpixel flows. In *In Proc. ECCV*, pages 268–281. Springer, 2010. 2

[27] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr. Fast online object tracking and segmentation: A unifying approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019. 2, 5, 6, 7, 8

[28] C. Xu and J. J. Corso. Evaluation of super-voxel methods for early video processing. In *In Proc. CVPR*, pages 1202–1209. IEEE, 2012. 5

[29] N. Xu, L. Yang, Y. Fan, J. Yang, D. Yue, Y. Liang, B. Price, S. Cohen, and T. Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 585–601, 2018. 2

[30] X. Zhang, T. Dekel, T. Xue, A. Owens, Q. He, J. Wu, S. Mueller, and W. T. Freeman. Mosculp: Interactive visualization of shape and time. In *The 31st Annual ACM Symposium on User Interface Software and Technology*, pages 275–285. ACM, 2018. 5