EpO-Net: Exploiting Geometric Constraints on Dense Trajectories for Motion Saliency

Muhammad Faisal[1], Ijaz Akhter[2], Mohsen Ali[1], and Richard Hartley[3]

[1]Information Technology University, Pakistan
[2]KeepTruckin, Inc, [3]Australian National University, Australia

Abstract

The existing approaches for salient motion segmentation are unable to explicitly learn geometric cues and often give false detections on prominent static objects. We exploit multiview geometric constraints to avoid such shortcomings. To handle the nonrigid background like a sea, we also propose a robust fusion mechanism between motion and appearance-based features. We find dense trajectories, covering every pixel in the video, and propose trajectory-based epipolar distances to distinguish between background and foreground regions. Trajectory epipolar distances are data-independent and can be readily computed given a few features' correspondences between the images. We show that by combining epipolar distances with optical flow, a powerful motion network can be learned. Enabling the network to leverage both of these features, we propose a simple mechanism, we call input-dropout. Comparing the motion-only networks, we outperform the previous state of the art on DAVIS-2016 dataset by 5.2% in the mean IoU score. By robustly fusing our motion network with an appearance network using the input-dropout mechanism, we also outperform the previous methods on DAVIS-2016, 2017 and Segtrackv2 dataset.

1. Introduction

Segmenting object(s) with significant motion in a video is called Salient Motion Segmentation. In contrast, segmenting the most prominent object(s) in an image (or a video) is Salient Appearance Segmentation. While the data-driven approaches have been quite successful for the later, we argue, that the former suffers from the scarcity of the video-based training data and remains ill-posed. Specifically, for a moving camera, it remains hard to learn, whether the 2D projected motion field corresponds to a static object in the video, or the one having independent motion. To segment out the rigid background from the independently moving foreground objects, we exploit extensively studied geometric constraints [14], over the complete video, in a learning paradigm. Unlike the data-dependent learning, these constraints have closed-form solutions and can be computed very efficiently. In Fig. 1 we give an example from DAVIS [32], showing that the previous approaches give false detections on prominent static objects; whereas the proposed approach can disambiguate static and nonstatic objects. This clearly shows that the existing deep-networks are unable to automatically learn the geometric cues even when the optical flow was provided as an input.

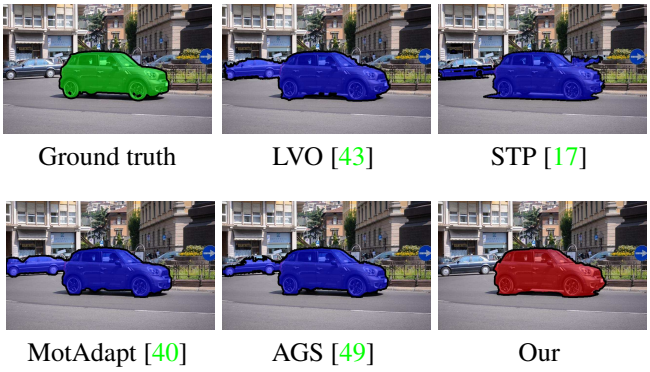


Figure 1: Existing methods fail to automatically learn geometric cues between the foreground objects and the rigid background. As a result, they often give false detections on prominent static objects, as shown here in an example from DAVIS [32]. Whereas by exploiting these constraints over the whole video, we avoid making such mistakes.

To exploit multiview geometric constraints, we convert optical flow between consecutive frames into dense trajectories, covering every pixel in the video, and then use trifocal tensors to find epipolar distances [14] for them. The trajectory epipolar distance serves as a measure of (non)rigidity: a small distance corresponds to the rigid background, and a

large distance implies a foreground object(s).

Trajectory epipolar distances, capture temporally global constraint on the foreground and background region, whereas optical flow only captures local temporal information. In essence, they both are complementary and by combining them, powerful features for motion saliency can be learned. Given trajectory epipolar distances and optical flow as an input, we build an encoder-decoder based network [36], called EpO-Net. We devise a strategy called input-dropout, enabling the network to learn robust motion features and handle failure cases of one of the two inputs.

EpO-Net brings two key advantages over the existing motion network, Mp-Net [42]. 1) EpO-Net exploits geometric constraints over a large temporal window, whereas Mp-Net makes suboptimal decisions based on temporally local optical flow information. Consequently, as we show, EpO-Net can be trained on smaller training data, while having better generalization than Mp-Net. 2) In contrast to Mp-Net, EpO-Net does not require any objectness score on top of the estimated motion saliency map. The main reason for this is, we prepare and train our network on a more realistic but synthetic data consisting of real backgrounds and synthetic foreground objects, called Real Background and Synthetic Foreground (RBSF) dataset. Whereas, Mp-Net was trained on unrealistic synthetic 3D flying objects [28].

Being a motion-only network, EpO-Net cannot handle a nonrigid background. To handle this case, we exploit appearance [3] along with motion-based features in the form of a joint network, EpO-Net+. Using the proposed input-dropout strategy, we show that the EpO-Net+ is robust against the failure cases of individual motion and appearance-based features.

To the best of our knowledge, ours is the first method to combine geometric constraints in a learning paradigm for motion segmentation. Our paper has three main contributions. 1) A motion only network based on trajectory epipolar distance and optical flow. 2) Our RBSF dataset, that can be used to train salient motion segmentation. Applications like video annotation [10], object tracking [54], and video anomaly detection [52], can use our network and the dataset, to exploit geometric constraints on the rigid world. The source code of our method, as well as the dataset, is publicly released[1]. 3) The input-dropout technique, which can be used to robustify early or late fusion of features in deep architectures. Our motion network outperforms Mp-Net on DAVIS-2016 [32] by a significant margin of 5.2% in mean IoU score and is quite close to other recent methods exploiting additional appearance features. The proposed joint network also demonstrates significant improvement over the previous methods on DAVIS (2016 [32] & 2017 [35]) and Segtrack-v2 [23].

[1] https://github.com/mfaisal59/EpONet

2. Related Work

Recently, video object segmentation (VOS) has been gaining interest [18, 42, 43, 41, 17, 6, 21], much credit to the new challenging benchmark datasets. One of the factors to categorize existing approaches could be the degree of supervision. Supervised approaches [29, 5] or interactive ones assume user input, in the form of scribbles, is available at multiple instances, helping algorithm refine the results. Semi-Supervised methods [17, 53, 16, 24, 1, 27, 25], assume that at least for the first frame, segmentation is given, reducing the problem to label propagation. For brevity, we discuss only a few prominent unsupervised methods.

In unsupervised settings, to make the problem tractable the motion-saliency constraint is enforced. Many methods try to capture motion information across the multiple frames, mostly by constructing long sparse point trajectories [2, 11, 31, 39]. Salient object segmentation is then reduced to clustering these trajectories [20] and converting them into dense points [30]. Among the other early methods, few methods [22, 23, 26, 55, 33] extract object proposals [8] and try to build the connection between the proposals temporally. These trajectory based methods are not robust because they heavily rely on feature matching, that may fail due to occlusion, fast motion, and appearance change.

Recently deep learning based methods have been used to solve the VOS problem. Broadly, these techniques have three components: 1) network to capture the motion, 2) extract appearance information, 3) a temporal memory so that the decision made at one frame is propagated to the others [43, 18, 6, 41]. Among all these approaches, Mp-Net [42] and LVO [43] are very close to our method. Mp-Net constructs an encoder/decoder based network to segment the optical flow into the salient and non-salient one. Encoder/decoder network is trained on large synthetic dataset [28] and then fine-tuned on DAVIS [32]. Since motion information they learn is not sufficient, they rely on an objectness score [34] to clean their results. LVO, builds on Mp-Net, using bi-directional ConvGRU to propagate the information across the other frames. Their results improve drastically (LSMO [44]) by just using a better optical flow estimation and appearance model (DeepLabv2 instead of Deep Lab v1). MotAdapt [40] used the teacher-student learning paradigm, where the teacher provides pseudo labels using the optical flow and the image as input.

AGS [49] explores the concepts of video saliency or dynamic fixation prediction, with an argument that unsupervised VOS is closely related to the video saliency [47]. Authors trained a visual attention module on the dynamic fixation data, collected by tracking viewers' eyes while they watch videos. Unlike AGS which required the data gathered by tracking the viewer's gaze, we try to model the concept of motion-saliency by exploiting the geometric constraints

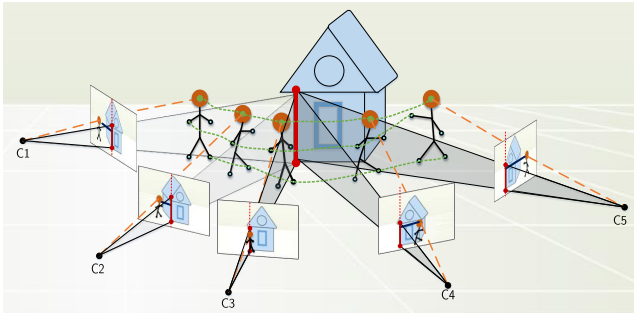


Figure 2: An illustration of multiview geometric constraints on rigid points. A 3D rigid line (red) is viewed by a moving camera. The projections of its 2D projections in 3D should meet at the actual line. In contrast, the 2D projections of a 3D nonrigid point (orange) are not constrained to lie on any 3D lines. This relationship can be captured in the form of trifocal tensors (or fundamental matrices) in the frames. In contrast to rigid points, the nonrigid point may not lie on the corresponding epipolar lines and their epipolar distances can serve as a measure of nonrigidity.

inside the video itself and do not require extra data.

An early method by Torr [45], Sheikh et. al. [38], and Tron and Vidal [46], try to exploit motion models. [38], and [46] exploited trajectory information to separate out the foreground and background objects. Many recent methods [21, 19, 17] have relied on the previous trajectory-based segmentation work, using the deep features for image saliency and optical flow for motion saliency to construct a neighborhood graph. [37] used optical flow-based point trajectories to propagate the user input scribbles. [48] clustered neighboring trajectories to create super-trajectories, and tracked the mask, provided as input, in the first frame of the video. However, they have not exploited the geometry-based constraints, rather rely on the heuristics and complex pipeline.

Our work relies on all the three techniques. We use optical flow to build trajectories and geometry-based technique to penalize the trajectories not following the geometric constraint. To make our deep learning models robust, we design the input-dropout technique for the training. To the best of our knowledge, we are the first one to combine CNNs and geometrical constraints for VOS.

3. Epipolar Constraints on Dense Trajectories

Existing methods for salient motion segmentation, use appearance, and optical flow based features to distinguish foreground from background. These features are not geometry inspired, learned from the data and alone do not provide enough constraints for the rigid background. We propose geometry inspired features and leverage them in a learning pipeline. We use trifocal tensors to constraint the rigid background in the video and propose epipolar distances for the dense trajectories as a measure of nonrigidity (See Fig. 2).

Figure 3: An illustration of exploiting the complete trajectories to find epipolar distances. Part of the bear remains static in this and the previous frame, giving small epipolar distance (middle). Since trajectories aggregate these distances over their full time-span, the trajectory-based epipolar distances are still high for almost the complete bear (right).

We first find forward and backward optical flow of F frames, each of height h and width w, using [4] and then convert it into T dense trajectories covering every pixel in the video. Each trajectory, $\mathbf{X}^i$, where $i \in \{1, \ldots, T\}$, is an $F \times 1$ vector of 2D image coordinates and may consists of missing values due to pixels' occlusion. $T \gg hw$, because for every occlusion new pixels appear. We use forward and backward optical flow consistency to find occluding regions. We stack all the trajectories into a $F \times T$ sparse matrix, $\mathbf{X}$.

Once trajectories are found, we estimate the dominant rigid background, by finding the trifocal tensors in every three consecutive frames, using the six-point algorithm [14][2], and RANSAC. We convert the trifocal tensor to the corresponding six pair-wise fundamental matrices, $\mathbf{F}_{12}, \mathbf{F}_{21}, \mathbf{F}_{13}, \mathbf{F}_{31}, \mathbf{F}_{23}, \mathbf{F}_{32}$ [14][3]. When the camera is static and optical flow is zero for the background, the estimation of the trifocal tensor can become degenerate. Any skew-symmetric matrix, in this case, would be a valid fundamental matrix. To avoid degeneracy, we first detect if the camera remains static, by checking if at least 50% of the pixels have zero optical flow, in the current triplet of frames. Then we initialize fundamental matrices to arbitrary skew-symmetric matrices.

We find the epipolar distances for the triplet as follows. Let $\mathbf{x}_{j1}, \mathbf{x}_{j2}$ and $\mathbf{x}_{j3}$ denote the homogenous 2D coordinates of the selected three frames in the j^{th} trajectory. We find the distance between $\mathbf{x}_{j1}$ and $\mathbf{x}_{j2}$ as,

$$\mathbf{l}_{21} = \mathbf{F}_{21}\mathbf{x}_{j1}, \tag{1}$$

$$d_{j12} = \mathbf{x}_{j2}^T \mathbf{l}_{21} / \sqrt{\mathbf{l}_{21}(1)^2 + \mathbf{l}_{21}(2)^2}, \tag{2}$$

where $\mathbf{l}_{21}$ is the epipolar line in frame 2 corresponding to the frame 1, $\mathbf{l}_{21}(i)$, its i^{th} component and d_{j12} is the distance between the line and $\mathbf{x}_{j2}$. By normalizing the line w.r.t its magnitude, gives the normlize epipolar distance. The triplet epipolar distance would be

$$d_{j123} = d_{j12} + d_{j21} + d_{j13} + d_{j31} + d_{j23} + d_{j32}. \tag{3}$$

[2] Algorithm 20.1 page 511, Hartley & Zisserman (2nd Ed)
[3] Algorithm 15.1, page 375, Hartley & Zisserman (2nd Ed)

The epipolar distance for the trajectory j is computed as the mean of all triplet epipolar distances along this trajectory. Concatenating all the trajectory epipolar distances gives a $1 \times T$ matrix, $\mathbf{D}$.

We assign the epipolar distance of a trajectory to all the constituent pixels. Hence, the proposed approach can deal with parts of the foreground object that remain static for a few frames but were in motion otherwise. As we show in Fig. 3 the epipolar distance estimated based on the current and the previous frame is quite small for the static part of the bear, whereas the trajectory-based epipolar distance can detect a significant part of the bear. Trajectory epipolar distances help us find powerful motion features for video segmentation, as we show in the next section.

4. Approach

The proposed pipeline consists of three distinct stages as illustrated in Fig 4. 1) Our motion network, **EpO-Net** takes optical flow and epipolar distances as input, and outputs ***motion-saliency-map***. 2) Parallel to this, we have a network to compute the appearance features to extract scene context and object information [3]. 3) Our joint network, **EpO-Net+** fuses the appearance features and the motion-saliency-map with a bidirectional-ConvGRU and outputs saliency mask.

4.1. Motion Images

Given an input video, we compute optical flow, convert it into dense trajectories, find trajectory epipolar distances and convert them into per-frame *Epipolar Distances* (ED). Having a *temporally* bigger receptive field, ED assigns a large weight to the foreground and lower to the background. However, it is sensitive towards optical flow errors because, during trajectory estimation, optical flow errors accumulate over time, affecting all the constituent pixels and their corresponding epipolar distances. Whereas, optical flow captures temporally local but relatively robust information containing motion patterns to distinguish foreground from background. Both of them are complementary and should be exploited jointly. We concatenate 2-channel optical flow vectors with ED, to get a 3-channel image, we call *motion-images*, as shown in Fig 5.

4.2. Epipolar Optical flow Network (EpO-Net)

Given the motion image as input, we design an encoder-decoder architecture, in the fashion of UNet [36] that outputs motion-saliency-map. The encoder latent space captures the context of the whole motion image, by jointly exploiting motion patterns and their relationship with ED. The decoding part on the other-hand has unraveled the context to decide about each pixel. The use of skip layers gives decoder access to local information ([51]) collected from the lower layers of the encoding-network and helps to exploit the context to decide the pixel level labels.

In our network, we use four encoders followed by four decoders, where each block consists of a convolution layer, followed by batch normalization, ReLU activation, and max-pooling layers. Different from Mp-Net, our much informative input allows us to have fewer channels before the final classification layer (128 instead of 512). Motion-saliency-map is produced using a sigmoid layer in the end. CRF is used to clean the output. A detailed architecture diagram showing the parameters of EpO & EpO+ is shown in the supplementary material.

4.3. Joint Network (EpO-Net+)

Any algorithm solely based on motion information will struggle with defining object boundaries and be confused by the non-rigid background. To exploit the additional appearance information, we use the pre-trained Deep-Lab [3] features and fuse them with our motion network, similar to LVO [43]. Although the FC6 layer of Deep-Lab is just $1/8$th of the spatial size of the original image, it still captures important information about the objects, their boundaries, and nonrigid background. Although customized appearance networks for video segmentation can produce better segmentation results, we choose to use rather generic appearance-based features, to demonstrate the significance of the proposed motion network.

We train the bottleneck layer to reduce the appearance features from 1024 to 128 and concatenate it with the down-sampled output of EpO-Net. To exploit temporal continuity in the joint-features and build a global context, we concatenate the bi-directional Convolutional Gated Recurrent Unit at the end of our network. To robustly handle motion network failures in the case of nonrigid background, we introduce input-dropout, discussed in Sec. 5.2.

5. Challenges in Training

The proposed architecture contains a fusion of mixed features, encapsulating information at varied spatial and temporal receptive fields, at different stages of the network. To enable the network to properly learn the concept of motion saliency, and robustly fuse these features, required contribution both in the training methodology and dataset.

5.1. RBSF Dataset

Training sequences in the DAVIS 2016 are too few to train a robust motion network. We find that F3DT [28] and PHAV [7] datasets are not very useful for us. F3DT has holes and the objects' motion is quite fast. PHAV is low resolution than DAVIS and the ground-truth optical flow is noisy because of jpeg compression. We create our own synthetic dataset, called **RBSF** (Real Background, Synthetic Foreground), by mixing 20 different foreground objects

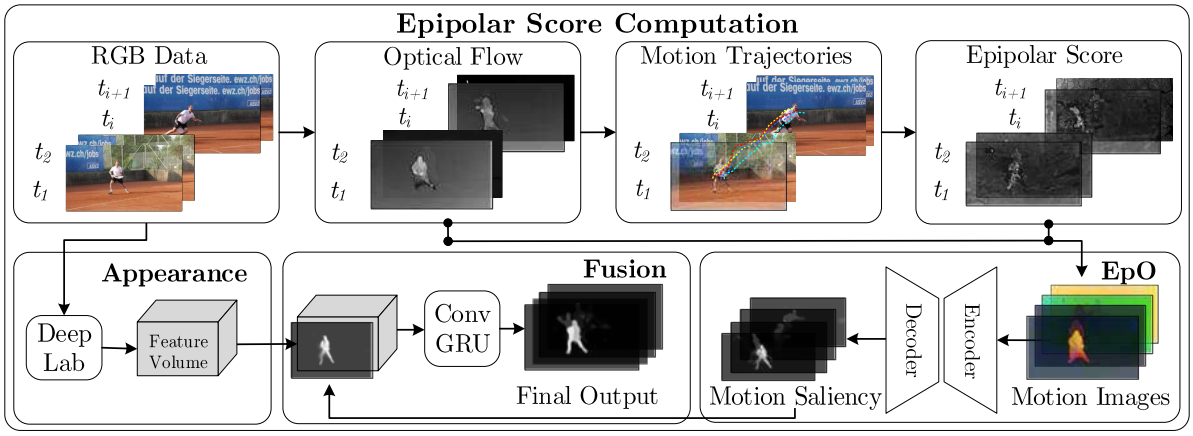


Figure 4: Flow diagram depicting different parts & information transition in the algorithm. Top Row: steps to compute the motion trajectories & Epipolar Distance. Bottom row: (Left) Deep-Lab based Appearance Network trained to compute the Appearance Features. (Right) Motion-Images (Optical Flow & Epipolar Distance) fed to EpO, which outputs motion saliency map. (Middle) Motion-saliency map concatenated with appearance features are fed into the bidirectional convGRU.

performing various movements with 5 different real background videos. With fairly large size objects (size: 30% to 50% of the frame) and reasonably fast motion, RBSF allows us to compute accurate optical flow and long trajectories. We observe that generating more data does not improve results, thanks to the well-constrained epipolar distances. After training on RBSF, we fine-tune EpO-Net on DAVIS-2016 [32]. For more details of the dataset, please see the supplementary material.

5.2. Feature Fusion & Input-Dropout

Robustly fusing optical flow and epipolar distances is a challenging task. Ideally, the network should be able to learn which feature to rely on the pixel-level granularity. But this requires contextual information that is only available in the deeper layers of the network, where the resolution is usually very small and the network has already mixed the input channels. In such a scenario, training with more data or for more iterations usually does not improve the results.

This problem is usually solved by introducing a mix of early and late fusion, requiring complex network designs, where skip layers are going from one part of the network to the others. Our proposed solution is rather quite simple, which we call **Input-Dropout**. While training EpO-Net, we randomly make complete ED-channel zero, for some of the sequences which have erroneous ED-maps (sequences with large motion and a considerable occlusion). For the rest, motion-images are unaltered. This is done for the initial 10 epochs, allowing the filters to give more importance to the optical flow. After that, we repeat the same procedure but instead of zero, we assign random values, forcing the network to learn the diverse enough filters to capture the motion information from the optical flow, ED and their combination, separately. With input-dropout EpO's mean IoU increases from 72.7 to 75.2 (Table 6).

Method	AC	DB	FM	MB	OCC	Mean
Mp-Net	0.71 *-0.02*	0.58 *0.14*	0.68 *0.04*	0.65 *0.10*	**0.69** ***0.01***	0.700
EpO	**0.77** ***-0.03***	**0.63** ***0.14***	**0.72** ***0.06***	**0.67** ***0.14***	0.67 *0.11*	**0.752**

Table 1: EpO-Net vs. Mp-Net [42] on DAVIS-2016 dataset.

We exploit the same input-dropout strategy for the late fusion of appearance and motion features in our joint network. We randomly set the motion-saliency-map to zero for a few frames of the sequences, where the motion network fails (sequences with dynamic background and occlusion). Using input-dropout, mean IoU improves from 79.4 to 80.6. The complete network, containing all the above stages and layers is called **EpO-Net+**.

6. Experiments

We train and evaluate on RBSF (Sec. 5.1), DAVIS2016 [32], DAVIS2017 [35] and Segtrack-v2 [23]. Below we detail our training parameters and evaluation results.

6.1. Implementation Details

EpO is trained using the mini-batch SGD with a batch size of 12, the initial learning rate is set to 0.001, with a momentum of 0.9, and a weight decay of 0.005. The network is trained from scratch for 50 epochs, with the learning rate decay factor set to 0.1, after every 5 epochs. The images are down-sampled by a factor of 0.5 to fit a batch size of 12 images in the GPU memory.

We train EpO in two stages: training on a synthetic dataset, RBSF (Sec. 5.1), and then fine-tuning on DAVIS-2016. For both of these training, we perform input-dropout for epipolar channel for only 20% of training data i.e. we randomly assign zero and add small random Gaussian noise in the epipolar channel. We call this final trained model,

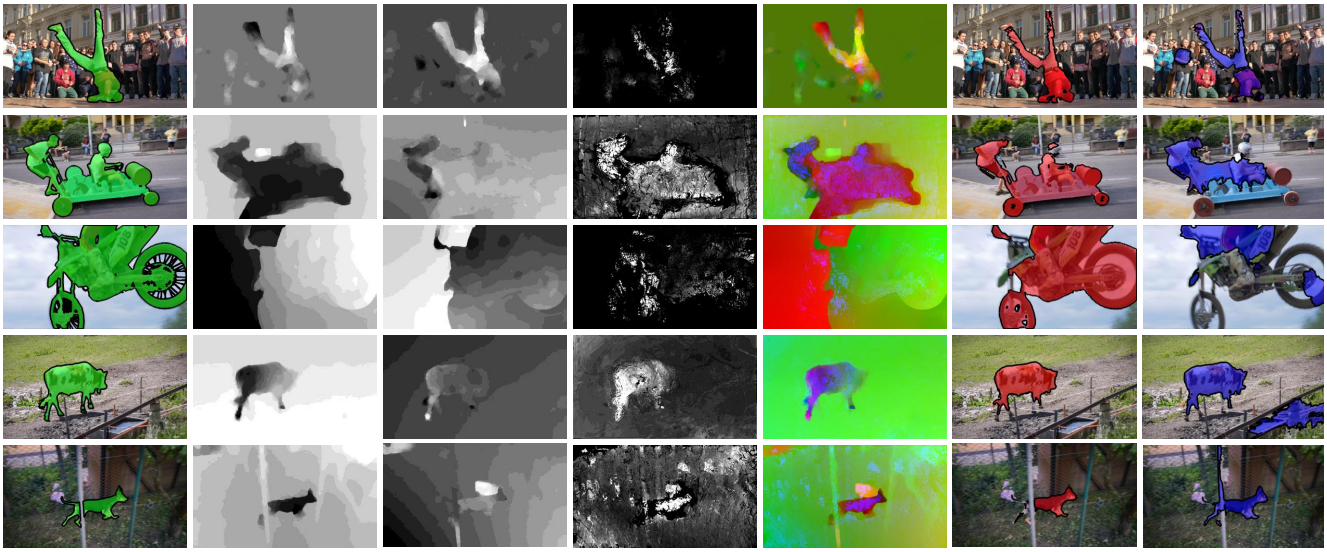


Figure 5: Qualitative Comparison of our EpO-Net with Mp-Net [42].

EpO and the one trained on RBSF, **EpO-RBSF**.

The fusion network is fully trained only on the DAVIS-2016's training set, resulting in **EpO+**. We use the batch size of 12 and an initial learning rate set to 0.001, which is decreased after every epoch with a factor $\frac{epoch}{50}$. The model is trained using the back-propagation through time [50] using binary cross-entropy loss and RMSProp optimizer. The weights of all the layers in the fusion network are initialized using the Xavier [12], except for those in ConvGRU, that is initialized using MSR initialization [15]. We clip the gradients to the [-50, 50], before each update step [13] to avoid numerical issues. For robust fusion, we again use the input-dropout mechanism by setting the motion-saliency-map to zero, for 20% frames of the sequence with fast motion and dynamic background. We also perform the random cropping and flipping of sequences during the training. The fusion network is trained for 50 epochs. The final output is refined using CRF, during inference. To test on DAVIS-2017, we fine-tine EpO-RBSF and EpO on the DAVIS-2017's training-set.

6.2. Evaluation

We follow the standard training & validation split, to train and evaluate using the protocol proposed in [32] and compute intersection-over-union $\mathcal{J}$, F-measures $\mathcal{F}$, and temporal stability $\mathcal{T}$, contour accuracy and smoothness of segmentation overtime respectively. The evaluation results are summarized in Table 2.

6.2.1 Motion Network

By exploiting geometric constraints in salient motion segmentation, our EpO (motion-only) network scores mean $\mathcal{J}$ of 0.752 over **DAVIS-2016 validation set**. This is much higher than 0.70 score of Mp-Net [42], which also relies on non-motion features (objectness score). MP-Net is trained on 45K frames using ground-truth optical flow, whereas EpO uses only 20K frames and an estimated optical flow on them. We observe that using more data does not improve the performance, thanks to the well-constrained epipolar distances. Moreover, our EpO score is competitive to LVO, which is using a bi-directional ConvGRU and the appearance information in addition to optical flow. Whereas EpO only uses motion-images (optical flow & ED).

Qualitative comparison of EpO-Net with Mp-Net is given in Fig. 5. It's evident from the 2^{nd} to 4^{th} columns that ED and optical flow are complimenting each other, and the results are robust against the failure of one of these inputs. In the case of optical flow being too small, or if the object motion is in the same direction as the camera motion (row-2), ED helps distinguish the object. Similarly, when the ED score is sporadically bad (row-1 & 3), optical flow information helps to distinguish the object, much due to the robust motion features learned with input-dropout training. Whereas Mp-Net makes local decisions, unable to recover from the optical flow errors (row 3 & 5). Their results also degrade when the camera and object have similar motion (row-3).

6.2.2 EpO+

Combining the motion-saliency map obtained from EpO with the appearance features and adding temporal memory, EpO+ outperforms its direct competitors LVO and LSMO, by a significant margin of 4.7% and 2.4% over mean IoU. EpO+ outperforms even recently published works, like AGS [49], which requires training on dynamic fixation

Measure	EpO+	EpO	AGS[49]	MOA[40]	LSMO[44]	STP[17]	PDB[41]	ARP[21]	LVO[43]	Mp-Net[42]	FSeg[18]	SFL[6]
Mean $\mathcal{M}\uparrow$	**0.806**	0.752	0.797	0.772	0.782	0.776	0.772	0.762	0.759	0.700	0.707	0.674
$\mathcal{J}$ Recall $\mathcal{O}\uparrow$	**0.952**	0.888	0.911	0.878	0.891	0.886	0.901	0.911	0.891	0.850	0.835	0.814
Decay $\mathcal{D}\downarrow$	0.022	0.053	0.019	0.050	0.041	0.044	0.009	0.070	**0.000**	0.013	0.015	0.062
Mean $\mathcal{M}\uparrow$	0.755	0.711	**0.774**	0.774	0.759	0.750	0.745	0.706	0.721	0.659	0.653	0.667
$\mathcal{F}$ Recall $\mathcal{O}\uparrow$	**0.879**	0.830	0.858	0.844	0.847	0.869	0.844	0.835	0.834	0.792	0.738	0.771
Decay $\mathcal{D}\downarrow$	0.024	0.043	0.016	0.033	0.035	0.042	**-0.002**	0.079	0.013	0.025	0.018	0.051
$\mathcal{T}$ Mean $\mathcal{M}\downarrow$	**0.185**	0.388	0.267	0.279	0.212	0.243	0.277	0.384	0.255	0.563	0.316	0.282

Table 2: Comparison of our motion (EpO) and fusion network (EpO+), with state-of-the-art on DAVIS-2016 with intersection over union $\mathcal{J}$, F-measure $\mathcal{F}$, and temporal stability $\mathcal{T}$. Best & second best scores have been bold and are underlined respectively. AGS uses eye-gaze data to train their network, whereas we only exploit information existent in the videos itself by enforcing the geomatrical constraints.

Attribute	EpO+	AGS[49]	MOA[40]	LSMO[44]	STP[17]
AC	**0.83** *-0.04*	0.80 *-0.01*	0.78 *-0.01*	0.78 *+0.00*	0.72 *+0.07*
DB	**0.72** *+0.10*	0.66 *+0.16*	0.61 *+0.20*	0.55 *+0.27*	0.66 *+0.15*
FM	**0.78** *+0.04*	0.77 *+0.04*	0.74 *+0.05*	0.73 *+0.08*	0.75 *+0.04*
MB	**0.78** *+0.06*	0.74 *+0.10*	0.71 *+0.10*	0.73 *+0.10*	0.74 *+0.06*
OCC	0.75 *+0.08*	0.76 *+0.05*	0.78 *-0.02*	0.74 *+0.06*	**0.81** *-0.05*

Table 3: Attribute-based analysis of top performing methods on DAVIS-2016 dataset. The mean IoU on all sequences with attributes: appearance cahnge (AC), dynamic background (DB), fast motion (FM), motion blur (MB), and occlusion (OCC), is computed. The smaller font values indicate the change in performance (gain or loss) for the method on the remaining sequences without that respective attribute.

dataset collected by tracking the gaze of viewers, both in mean IoU and its recall. Important to note is mean temporal stability, which is substantially better than rest explicitly indicating the effectiveness of our formulation. Our attribute analysis is given in Table 3. EpO+ outperforms the baselines in all categories except the occlusion.

Qualitative comparison of EpO+ with SOTA is presented in Fig. 6. AGS has failed to properly segment moving objects (2nd and 3rd row). Most of the errors in the previous methods are over-segmenting and are due to over-exploitation of appearance information. This we can attribute to the very basic reason of not being able to exploit/learn enough constraints for motion saliency.

While the proposed method, due to more informative proposed motion features (based on geometric constraints) and input-dropout training procedure, is being able to learn how to balance appearance and motion cues. For details see supplementary material.

6.2.3 Evaluation on other datasets

DAVIS-2017: We fine-tune EpO-RBSF and EpO+ on the DAVIS-2017's training sequences. We could not find the comparative results, but we are reporting ours for future comparison in Table 5.

Method	KEY	NLC	FSG	LVO	LSMO	STP	EpO	EpO+
Mean IoU	57.3	67.2	61.4	57.3	59.1	70.1	68.3	**70.9**

Table 4: EpO+ results on SegTrack-v2 dataset [23]. We only perform bad on one sequence (birdfall). Removing this increase our Mean IoU to 72.8.

Segtrack-v2: Evalaution resutls of EpO+ and EpO on SegTrack-v2 [23] dataset have been presented in Table 4. Our results are better than existing methods, including STP [17]. Although, it's with a small margin of 0.8%; this could be attributed to the difference in resolution of SegTrack-v2 videos vs that of DAVIS-2016. Removing birdfall, the only sequence we perform poorly, the results improves to 72.8%. AGS [49] uses both SegTrackv2 and DAVIS for training, therefore, do not evaluate on this. Note that, since NLC [9] reports results only on subset of sequences in their paper, results in Table 4 were taken from [43, 17].

Method	AC	DB	FM	MB	OCC	$\mathcal{J}$ Mean
EpO	0.67 *-0.02*	0.56 *0.10*	0.62 *0.04*	0.57 *0.11*	0.59 *0.08*	0.652
EpO+	0.79 *-0.04*	0.72 *0.05*	0.74 *0.03*	0.72 *0.06*	0.66 *0.13*	0.763

Table 5: Results on DAVIS 2017 dataset.

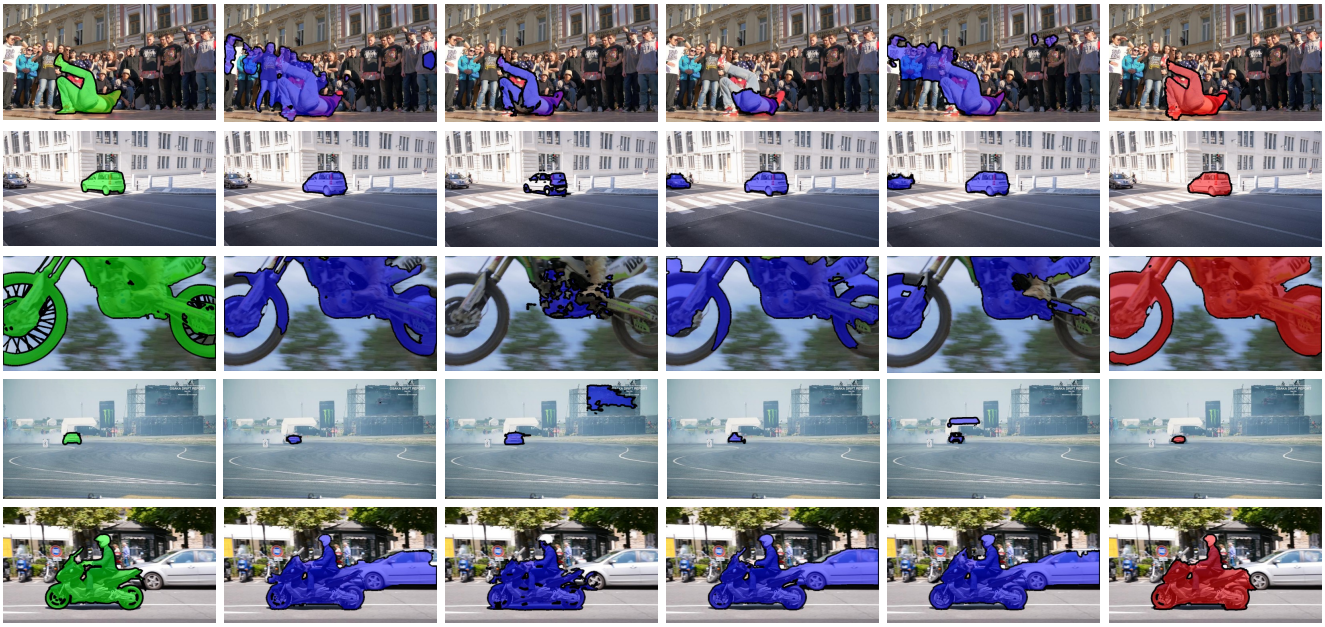
6.3. Ablation Study

In this section, we present the study on the impact and effectiveness of different design choices. We first analyze the influence of different input modalities and depth of the

#enc/dec	Input Modality		
	Ep	OF	Ep+OF
2	57.2	54.7	62.7
3	58.9	59.7	64.4
4	49.2	63.3	67.5

EpO Variant	Mean IoU
EpO(R)	48.5
EpO(D)	72.7
EpO(R)+Drop	50.6
EpO(D)+Drop	75.2

Table 6: **Left:** Studying the effects of different input modalities against network depth. **Right:** Effect of dropout in epipolar channel of motion images, R and D denote RBSF and DAVIS dataset respectively.



Ground truth | LVO [43] | STP [17] | MotAdapt [40] | AGS [49] | Our

Figure 6: Qualitative comparison with state-of-the-art methods on DAVIS-2016.

network architecture by training and validating on DAVIS-2016 dataset. Specifically, we use the single-channel ED, 2 channel optical flow i.e. X-Y displacement, and the combination of the both as 3 channel *motion images*. For each input modality, we train and validate EpO network with 2, 3 and 4-layer encoders/decoders to study which modality needs the deeper network.

In Table 6, we observe that ED being a very simple yet informative feature, the epipolar alone network requires fewer parameters to learn, implying that they should not require (i) deep network, ii) large datasets. In contrast, optical flow, being complex information for motion saliency, requires more encoders and decoders. Since small errors in optical flow, get accumulated in trajectories estimation and result in quite noisy epipolar distances, optical flow with 4 encoders/decoders architecture beats the epipolar network, with 63.3% mean IoU. However, when we combine both, in the form of motion images, the accuracy further improves by 4.2%. This shows that the combination can exploit both the global temporal geometric information and local temporal motion information distinguishing the foreground and background. Note that all the experiments are performed using the same hyper-parameters stated in Sec. 6.1, the input-dropout strategy is not used, and all models are trained for 30 epochs only.

Next, we demonstrate the effectiveness of our dataset RBSF and the input-dropout in Table 6. The mean IoU on DAVIS-2016 with the proposed dataset was 48.5%. That increases to 72.7% with fine-tuning on DAVIS. Comparing this with our Ep+OF's, the increase is 5.3%, showing the significance of the proposed dataset. With the proposed dropout the results further improve by 2.5%, showing the effectiveness of the input-dropout.

We also study the effect of GRU-sequence length. As expected, when we increase sequence length, from 6 to 12, the mean IoU improves from 77.3 to 79.4. A considerable improvement comes in the videos having occlusion. Finally, we observe that instead of the angle-magnitude representation of optical flow, the velocity representation gives better results. A qualitative review of the dataset, made us realize that the channel representing angle information is not robust to optical flow errors. Even for humans, inferring motion patterns by just looking at them, is quite difficult.

7. Conclusion

We exploit multiview geometric constraints to define motion saliency and propose trajectory epipolar distances, as a measure of non-rigidity. By combining epipolar distances with optical flow, we train a powerful motion network and demonstrate significant improvement over the previous motion networks. Unlike previous methods, the learned motion features avoid over-reliance on appearance-based features. Even without using RNNs and appearance features, our motion network is competitive to the existing state of the art. With them, our method gives state of the art results. An input-dropout mechanism has been proposed that allows network to learn robust feature fusion. The proposed learning paradigm, involving the strong geometric constraints, should be useful for a number of related applications.

References

[1] L. Bao, B. Wu, and W. Liu. Cnn in mrf: Video object segmentation via inference in a cnn-based higher-order spatio-temporal mrf. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5977–5986, 2018. 2

[2] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. In *European conference on computer vision*, pages 282–295. Springer, 2010. 2

[3] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 2, 4

[4] Q. Chen and V. Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4706–4714, 2016. 3

[5] Y. Chen, J. Pont-Tuset, A. Montes, and L. V. Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1189–1198, 2018. 2

[6] J. Cheng, Y.-H. Tsai, S. Wang, and M.-H. Yang. Segflow: Joint learning for video object segmentation and optical flow. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 686–695. IEEE, 2017. 2, 7

[7] C. De Souza, A. Gaidon, Y. Cabon, and A. Lopez Pena. Procedural generation of videos to train deep action recognition networks. In *CVPR*, 2017. 4

[8] I. Endres and D. Hoiem. Category independent object proposals. In *European Conference on Computer Vision*, pages 575–588. Springer, 2010. 2

[9] A. Faktor and M. Irani. Video segmentation by non-local consensus voting. In *BMVC*, volume 2, page 8, 2014. 7

[10] S. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *null*, pages 1002–1009. IEEE, 2004. 2

[11] K. Fragkiadaki, G. Zhang, and J. Shi. Video segmentation by tracing discontinuities in a trajectory embedding. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1846–1853. IEEE, 2012. 2

[12] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010. 6

[13] A. Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013. 6

[14] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 1, 3

[15] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1026–1034, 2015. 6

[16] Y. Hu, J. Huang, and A. G. Schwing. Videomatch: Matching based video object segmentation. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, pages 56–73, 2018. 2

[17] Y.-T. Hu, J.-B. Huang, and A. Schwing. Unsupervised video object segmentation using motion saliency-guided spatio-temporal propagation. In *Proc. ECCV*, 2018. 1, 2, 3, 7, 8

[18] S. D. Jain, B. Xiong, and K. Grauman. Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos. *arXiv preprint arXiv:1701.05384*, 2(3):6, 2017. 2, 7

[19] Y. Jun Koh, Y.-Y. Lee, and C.-S. Kim. Sequential clique optimization for video object segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 517–533, 2018. 3

[20] M. Keuper, B. Andres, and T. Brox. Motion trajectory segmentation via minimum cost multicuts. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 3271–3279. IEEE, 2015. 2

[21] Y. J. Koh and C.-S. Kim. Primary object segmentation in videos based on region augmentation and reduction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, page 6, 2017. 2, 3, 7

[22] Y. J. Lee, J. Kim, and K. Grauman. Key-segments for video object segmentation. In *IEEE International Conference on Computer Vision*, pages 1995–2002. IEEE, 2011. 2

[23] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg. Video segmentation by tracking many figure-ground segments. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2192–2199, 2013. 2, 5, 7

[24] S. Li, B. Seybold, A. Vorobyov, A. Fathi, Q. Huang, and C.-C. J. Kuo. Instance embedding transfer to unsupervised video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6526–6535, 2018. 2

[25] J. Luiten, P. Voigtlaender, and B. Leibe. Premvos: Proposal-generation, refinement and merging for video object segmentation. In *14th Asian Conference on Computer Vision, Perth, Australia, December 2-6, 2018, pages = 565–580*. 2

[26] T. Ma and L. J. Latecki. Maximum weight cliques with mutex constraints for video object segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 670–677. IEEE, 2012. 2

[27] K. Maninis, S. Caelles, Y. Chen, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. V. Gool. Video object segmentation without temporal information. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(6):1515–1530, 2019. 2

[28] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048, 2016. 2, 4

[29] N. S. Nagaraja, F. R. Schmidt, and T. Brox. Video segmentation with just a few strokes. In *2015 IEEE International Con-*

ference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015, pages 3235–3243, 2015. 2

[30] P. Ochs and T. Brox. Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In *IEEE International Conference on Computer Vision*, pages 1583–1590. IEEE, 2011. 2

[31] P. Ochs and T. Brox. Higher order motion models and spectral clustering. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 614–621. IEEE, 2012. 2

[32] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 724–732, 2016. 1, 2, 5, 6

[33] F. Perazzi, O. Wang, M. Gross, and A. Sorkine-Hornung. Fully connected object proposals for video segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3227–3234, 2015. 2

[34] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *European Conference on Computer Vision*, pages 75–91. Springer, 2016. 2

[35] J. Pont-Tuset, S. Caelles, F. Perazzi, A. Montes, K.-K. Maninis, Y. Chen, and L. Van Gool. The 2018 davis challenge on video object segmentation. *arXiv preprint arXiv:1803.00557*, 2018. 2, 5

[36] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2, 4

[37] N. Shankar Nagaraja, F. R. Schmidt, and T. Brox. Video segmentation with just a few strokes. In *Proceedings of the IEEE ICCV*, pages 3235–3243, 2015. 3

[38] Y. Sheikh, O. Javed, and T. Kanade. Background subtraction for freely moving cameras. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1219–1225. IEEE, 2009. 3

[39] J. Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *Computer Vision, 1998. Sixth International Conference on*, pages 1154–1160. IEEE, 1998. 2

[40] M. Siam, C. Jiang, S. W. Lu, L. Petrich, M. Gamal, M. Elhoseiny, and M. Jägersand. Video segmentation using teacher-student adaptation in a human robot interaction (HRI) setting. *CoRR*, abs/1810.07733, 2018. 1, 2, 7, 8

[41] H. Song, W. Wang, S. Zhao, J. Shen, and K.-M. Lam. Pyramid dilated deeper convlstm for video salient object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 715–731, 2018. 2, 7

[42] P. Tokmakov, K. Alahari, and C. Schmid. Learning motion patterns in videos. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 531–539. IEEE, 2017. 2, 5, 6, 7

[43] P. Tokmakov, K. Alahari, and C. Schmid. Learning video object segmentation with visual memory. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 4491–4500, 2017. 1, 2, 4, 7, 8

[44] P. Tokmakov, C. Schmid, and K. Alahari. Learning to segment moving objects. *International Journal of Computer Vision*, 127(3):282–301, 2019. 2, 7

[45] P. H. Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 356(1740):1321–1340, 1998. 3

[46] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007. 3

[47] W. Wang, J. Shen, F. Guo, M.-M. Cheng, and A. Borji. Revisiting video saliency: A large-scale benchmark and a new model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4894–4903, 2018. 2

[48] W. Wang, J. Shen, J. Xie, and F. Porikli. Super-trajectory for video segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1671–1679, 2017. 3

[49] W. Wang, H. Song, S. Zhao, J. Shen, S. Zhao, S. C. Hoi, and H. Ling. Learning unsupervised video object segmentation through visual attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3064–3074, 2019. 1, 2, 6, 7, 8

[50] P. J. Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990. 6

[51] Z. Wojna, J. R. R. Uijlings, S. Guadarrama, N. Silberman, L. Chen, A. Fathi, and V. Ferrari. The devil is in the decoder. In *British Machine Vision Conference 2017, BMVC 2017, London, UK, September 4-7, 2017*, 2017. 4

[52] T. Xiang and S. Gong. Video behavior profiling for anomaly detection. *IEEE transactions on pattern analysis and machine intelligence*, 30(5):893–908, 2008. 2

[53] H. Xiao, J. Feng, G. Lin, Y. Liu, and M. Zhang. Monet: Deep motion exploitation for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1140–1148, 2018. 2

[54] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *Acm computing surveys (CSUR)*, 38(4):13, 2006. 2

[55] D. Zhang, O. Javed, and M. Shah. Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 628–635. IEEE, 2013. 2