Bayesian-Inspired Space-Time Superpixels

Supplementary Material

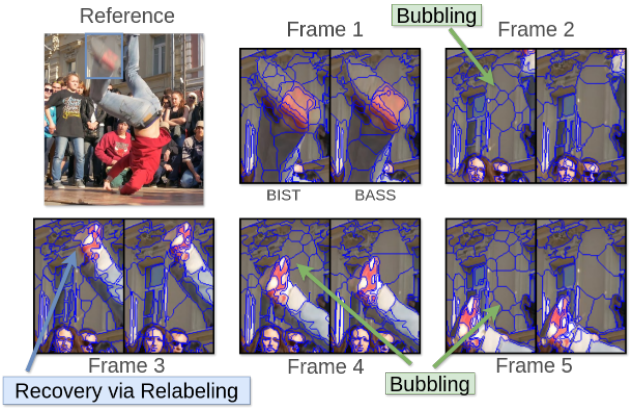

Figure 9. **Limitation #1: Bubbling.** When more superpixels are used than necessary to explain the underlying scene (as compared with BASS [32]), we call the phenomena *bubbling*. This occurs in BIST because propagated superpixels cannot be merged together to ensure temporal consistency.

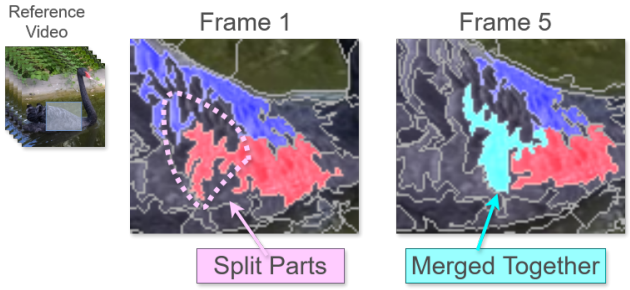

Figure 10. **Limitation #2: Temporal Fragmentation.** BIST adapts the number of superpixels to each image by splitting and merging superpixels, but this can lead to the unnecessary relabeling of a region which breaks methods that naively track a segmentation through superpixel labels alone. In this figure, two superpixels were split, and their pieces were later merged. The new segmentation is almost equally as good as the original, but now uses a new superpixel that is not propagated through time.

7. Limitations

Bubbling. One limitation of BIST we coin *bubbling* is depicted in Figure 9. Bubbling is when more superpixels are used than necessary to explain the scene. This happens because two propagated superpixels cannot be merged together, and it is often caused by improper optical flow estimation. In the figure, the dark region of the foot in frames 1 and 3 were not correctly estimated. This was recovered in the third frame by the relabeling step, since a foot reappears in the same area. This marks the region with new superpixels that can be merged, so unnecessary superpixels are removed. However, in frame 4 the foot moves again but is improperly tracked by optical flow. Unnecessary superpixels are not removed before the processing frame 4 is complete, so they propagate to frame 5. These extra superpixels will not be removed in the remaining frames of the video, since two propagated superpixels cannot be merged and it so happens that no moving object passes through the region, which would allow for relabeling and then merging of the unnecessary superpixels.

Temporal Fragmentation. Adapting the number of superpixels to each frame can needlessly lead to a failure in tracking comparable regions, and this is depicted in Figure 10. Even though the split-merge steps create another almost equally good superpixel segmentation, the teal-colored region no longer tracks the previous parts of the swan's feathers since it is assigned a new superpixel label. This limits the temporal extent of BIST and could deteriorate its utility for tracking-based applications. However, it seems likely heuristic post-processing steps could mitigate this problem. For example, one could record the lineage of each superpixel and optionally attach new superpixels to the propagated segmentation mask via classification.

8. Additional Details about BIST

8.1. Parameter Updates

To compactly present BIST, the boundary update step includes both computing the maximum likelihood estimates and re-classifying superpixel labels. For completion, this subsection details how the maximum likelihood estimates are computed using fixed superpixel labels $\boldsymbol{z}$. Lets write summary statistics of superpixel s as follows: $(n_s, \boldsymbol{a}_s^{'}, \boldsymbol{l}_s^{'}, \boldsymbol{l}_s^{''}) = \sum_{i:z_i=s}(1, \boldsymbol{a}_i, \boldsymbol{l}_i, \boldsymbol{l}_i\boldsymbol{l}_i^{\intercal})$. Notice these quantities can be easily summed using within a single kernel. Then we have,

$$\mu_s^{\text{app}} = \frac{\boldsymbol{a}_s^{'}}{n_s}, \quad \mu_s^{\text{shape}} = \frac{\boldsymbol{l}_s^{'}}{n_s}, \tag{6}$$

$$\widehat{\Sigma}_s = \frac{\boldsymbol{l}_s^{''} - \frac{\boldsymbol{l}_s^{'}\boldsymbol{l}_s^{'\intercal}}{n_s} + \lambda_s^4 \boldsymbol{I}_{2\times 2}}{n_s + 50\,\lambda_s^2 - 3} \tag{7}$$

The variance of the appearance term is a fixed hyperparameter, rather than estimated from data. The update term for the shape parameter is the posterior mean of an Inverse-Wishart prior with parameter set to $\lambda_s^4 \boldsymbol{I}_{2\times 2}$ with $50\lambda_s^2$ degrees of freedom. These choices are a major contribution

from a related work [32]. The term λ_s^2 represents the prior size of the superpixel and changes with each split/merge step of the superpixel. Specifically, say a superpixel has not undergone any splits/merges so the prior is the initial superpixel size, $\lambda_s = \lambda$. After a split, the prior is halved, $\lambda_s \leftarrow \lambda_s/2$. After a merge, the prior is the sum of the two priors involved, $\lambda_s \leftarrow \lambda_s + \lambda_{s'}$. This ensures that superpixels which have experienced the same number of splits/merges are approximately the same size.

8.2. Conditioned Parameter Updates

While the default operating regime of BIST uses the parameter updates detailed in the previous subsection, we also explored using parameter updates conditioned on the previous frame's summary statistics. The motivation is to encourage superpixels to remain the same shape across multiple frames. Experimentally, (Sec 10) we found this yielded a consistent improvement over standard parameter estimates, but the improvement was too marginal to use in the default case. This section presents the details about how the conditioning is done.

Let $(n_{t,s}, \boldsymbol{l}'_{t,s}, \boldsymbol{l}''_{t,s}) = \sum_{i:z_{t,i}=s}(1, \boldsymbol{l}_{t,i}, \boldsymbol{l}_{t,i}\boldsymbol{l}_{t,i}^\intercal)$ be the summary statistics for the 2D Gaussian's covariance term at frame t. The posterior mode for the covariance term in frame t uses the summary statistics from frame $t-1$,

$$\widehat{\Sigma}_{t,s} = \frac{\boldsymbol{l}''_{t,s} - \frac{\boldsymbol{l}'_{t,s}\boldsymbol{l}'^\intercal_{t,s}}{n_{t,s}} + \lambda_{t,s}^4 \widehat{\Sigma}_{t-1,s}}{n_{t,s} + 50\,\lambda_{t,s}^2 - 3} \tag{8}$$

This is simply a weighted average of the current timestep's sample covariance and the previous frame's estimated covariance. In addition to the modified parameter update, we explore setting the prior size to the size of the superpixel after the shift step; $\lambda_{t,s} \leftarrow \sum_{i:\tilde{z}_i=s} 1$. Once again, the motivation is to encourage superpixels to not change across time.

8.3. Merging

To reduce the number of superpixels, merge steps combine two superpixels into one [3, 13, 32]. BIST's merge step is identical to BASS, with the exception that two superpixels propagated from a previous frame cannot be merged together. This leaves only newly split or relabeled superpixels available for merging.

Consider all superpixels eligible for merging that neighbor superpixel s as $\mathcal{N}(s)$. Then superpixel s is proposed to be merged into the neighbor with the maximum superpixel label, $s' = \max \mathcal{N}(s)$. A Hastings ratio determines if the two superpixels should be merged. It uses a marginal likelihood term similar to Equation 4 for the split step. Denote the marginal likelihood of the two un-merged superpixels and the merged one as $f(\boldsymbol{x}^s)$, $f(\boldsymbol{x}^{s'})$, and $f(\boldsymbol{x}^{s,s'})$. Then write the marginal likelihood term,

$$f(\boldsymbol{x}^s) = \frac{\cancel{|\kappa + n_s|^{\frac{1}{2}}} b^a \Gamma(a + n_s/2)}{\cancel{|\kappa|^{\frac{1}{2}}} \hat{b}^{(a+n_s/2)} \Gamma(a) \pi^{\frac{n_s}{2}} 2^{n_s}} \left|\frac{1}{n_s}\right|^{\frac{1}{2}} \tag{9}$$

where the red right-most term is a heuristic modification fixed in previous work [32]. One way of understanding this term by considering the marginal likelihood of the shape parameters (See Eq 266 in [20]). The expression includes a counting term, $\left|\frac{\kappa}{\kappa+n_s}\right|$, which cancels out with the counting term from the marginal likelihood of the appearance parameters, $\left|\frac{\kappa+n_s}{\kappa}\right|^{\frac{1}{2}}$. The result is $\left|\frac{\kappa}{\kappa+n_s}\right|^{\frac{1}{2}} \rightarrow \left|\frac{1}{n_s}\right|^{\frac{1}{2}}$, but since the κ term is fixed to zero and/or dropped only the size of the superpixel is left in the denominator. Denoting $\alpha_s = \frac{\alpha}{2} + n_s$, the Hastings ratio of the merge step is then written as,

$$\frac{\cancel{\Gamma(n_s + n_{s'})}\Gamma(\alpha)\Gamma(\alpha_s)\Gamma(\alpha_{s'})}{\Gamma(n_s)\Gamma(n_{s'})\cancel{\Gamma(\alpha_s + \alpha_{s'})}\Gamma(\frac{\alpha}{2})^2} \frac{f(\boldsymbol{x}^{s,s'})}{\alpha f(\boldsymbol{x}^s) f(\boldsymbol{x}^{s'})} > e^{-2} \tag{10}$$

Notice how the merge step includes the Hastings parameter (α), which is a hyperparameter to the BIST algorithm. This allows for direct control over the number of superpixels for a particular image. However, as seen in the main paper's experiment 4.3, using this as a means of control for space-time superpixels is less effective than our proposed temporal consistency term presented in Section 3.3.

8.4. Relabeling

A relabeling step is common to existing space-time superpixel methods [3, 13], and generally BIST's relabeling step matches previous work. There are two types of relabeling: re-identification and new label creation. Active superpixels are superpixels with a non-zero count in the present frame, while inactive superpixels have a non-zero superpixel count for any previous frame but a zero count in the current frame. Relabeling allows active superpixels to be re-assigned to an inactive superpixel, with the goal of propagating superpixel labels through frames of total occlusion. New label creation allows active superpixels that were propagated from a previous frame to be re-assigned a new label, with the goal of ensuring temporally coherent superpixels have consistent appearance values. This allows for BIST to accommodate inaccurate optical flow, since improperly propagated superpixels can be dropped.

Re-identification. For re-identification, the difference between the appearance and shape means of each active superpixel is compared with all inactive superpixels. For example, take s and s' to be an active and inactive superpixel, the difference is then

$$d(s, s') = \|\mu_s^{\text{app}} - \mu_{s'}^{\text{app}}\|_2^2 + 0.01\|\mu_s^{\text{shape}} - \mu_{s'}^{\text{shape}}\|_2^2 \tag{11}$$

Let $s_{\text{re-id}} = \arg\min_{s'} d(s, s')$ be the inactive superpixel with the minimum difference to s, and $d_{\text{min}} = \min_{s'} d(s, s')$ be this minimum difference value. The relabeling update can then be expressed as:

$$s \leftarrow \begin{cases} s_{\text{re-id}} & \text{if } d_{\text{min}} < \varepsilon_{\text{re-id}} \\ s & \text{otherwise} \end{cases} \tag{12}$$

New Label Creation. For new label creation, each current superpixel's means are compared with its previously computed means from the most recent previous frame:

$$d(s) = \|\mu_s^{\text{app}} - \mu_{s,\text{prev}}^{\text{app}}\|_2^2 + 0.01\|\mu_s^{\text{shape}} - \mu_{s,\text{prev}}^{\text{shape}}\|_2^2 \tag{13}$$

If this difference exceeds some threshold (ε_{new}), then the current superpixel is assigned a new unique label:

$$s \leftarrow \begin{cases} s_{\text{new}} & \text{if } d(s) > \varepsilon_{\text{new}} \\ s & \text{otherwise} \end{cases} \tag{14}$$

Here, $s_{\text{new}} = S$ represents a newly generated label, where S was the total number of unique superpixel labels used so far in the video. This ensures that each new label is unique, since the superpixel labels take values in the set $\{0, 1, ..., S\}$ after the addition of a new label.

8.5. Achieving a Target Number of Superpixels

Both BIST and BASS [32] adaptively estimate the number of superpixels for a particular scene, and their estimates are robust to hyperparameters like the initial superpixel size (λ) and the Hastings hyperparameter for the merge step (α). While this is a desirable feature for applications, it makes standardized evaluation of their superpixel quality challenging since the number of superpixels significantly influences the quantitative superpixel metrics. Therefore, we designed a subroutine to ensure the number of superpixels of BIST and BASS is within 5% of a desired number of superpixels. This procedure is used to produce Figures 6 and 12. The core of this procedure are new constant offset terms which increase/decrease the chance of splitting/merging two superpixels. In this subroutine, the chance of splitting and merging varies depending on the current number of superpixels relative to the target. For example, if there are more superpixels than desired, the changes of splitting decreases and the chances of merging increase. In BIST, the chance of relabeling a superpixel as new one also grows, but only once every hundred iterations. The methods iterate until the number of superpixels is within 5% of the target or until 5000 iterations have been completed. The average number of superpixels across all videos for both datasets is within the tolerance. More details for this subroutine can be found in the open-source code.

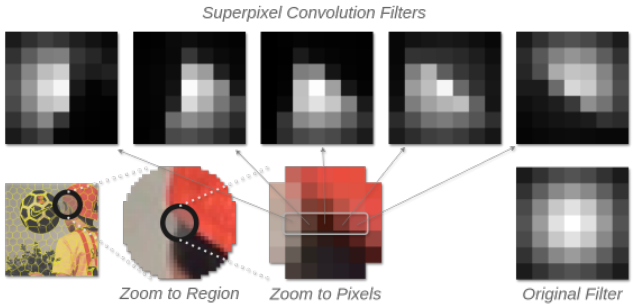

Figure 11. **Superpixel Convolution.** This figure illustrates superpixel convolution. One filter is adapted to each pixel using superpixel information, which allows the module to ignore information unrelated to the current pixel. Pictured are five re-weighted convolution filters associated with the five pixels within the center-bottom box. The filters centered at gray pixels ignore red pixels (left), and the filters centered at red pixels ignore gray pixels (right).

9. Superpixel Convolution

Today, few methods use superpixel-based modules within their deep neural network architecture. We believe this is due to a lack of easy-to-use code and few superpixel-enabled modules. This paper's superpixel method is Pytorch-friendly, and, superpixel convolution provides simple superpixel-enabled modules. This paper presents superpixel convolution as an application of space-time superpixels to garner interest in using them within deep neural networks.

Superpixel Convolution. A recent paper demonstrates that re-weighting an attention map with superpixel probabilities yields the optimal denoiser [8]. This paper extends the idea of re-weighting a kernel to convolution. In superpixel convolution, a convolution filter is re-weighted by superpixel similarities as below,

$$f_{\text{s-conv}}^{(i)}(\boldsymbol{x}; \boldsymbol{\pi}) = \sum_{j \in \mathcal{N}(i)} \gamma_{i,j} \boldsymbol{k}_j \boldsymbol{x}_j \tag{15}$$

$$\gamma_{i,j} = \frac{\sum_{z=1}^{S} \boldsymbol{\pi}^{(i,z)} \boldsymbol{\pi}^{(j,z)}}{\max_{j' \in \mathcal{N}(i)} \sum_{z=1}^{S} \boldsymbol{\pi}^{(i,z)} \boldsymbol{\pi}^{(j',z)}} \tag{16}$$

$$\boldsymbol{\pi}^{(i,z)} = \frac{\exp\left\{-\|\boldsymbol{x}_i - \boldsymbol{\mu}_z^{\text{app}}\|\right\}}{\sum_{z'} \exp\left\{-\|\boldsymbol{x}_i - \boldsymbol{\mu}_{z'}^{\text{app}}\|\right\}} \tag{17}$$

where $\boldsymbol{\pi}^{(i,z)}$ is the similarity between pixel i and superpixel z computed as a softmax applied the z dimension. A learnable kernel ($\boldsymbol{k}$) is re-weighted by the superpixel similarities. The advantage of such an operator is that the convolution kernel does not mix perceptually unrelated information. Figure 11 illustrates this concept.

Experimental Setup. We demonstrate the value of superpixel convolution on the single-image denoising task within

σ^2	Conv	BASS+Conv	BIST+Conv
10	34.46/0.885	36.14/0.927	35.94/0.923
20	30.81/0.797	32.68/0.866	32.33/0.854
30	28.91/0.733	30.80/0.818	30.43/0.803
Deno Params	612	612	612
Aux Params	0	5.691k	5.691k
Fwd Time (ms)	29	440	281

Table 3. **Denoising with Superpixel Convolution [PSNR↑/SSIM↑].** The superpixel convolution module improves the denoising quality compared to standard convolution. Streaming superpixel estimates are faster, and yield an approximately equally good superpixel estimate. Note that the auxiliary network is only used for superpixel estimation, and is otherwise disconnected from the denoised output image.

a small deep neural network. The network alternates between a standard convolution and an alternative convolution layer, ending with a convolution layer. The depth is fixed to 7 total layers (4 conv + 3 alt-conv), and the number of features is 6. For superpixel-enhanced convolution layers, the network uses a UNet-like model [27] to build features for superpixel estimation. Notably, these features are only connected to the denoiser through the superpixel-based re-weighting term in the convolution layer. The networks are trained on the DAVIS training dataset [21] for 30k iterations using Adam [11] and a batch size of five. We report the denoising quality on the DAVIS test set using PSNR/SSIM [35].

Results. Table 3 reports superpixel convolution yields significantly better denoising results compared to standard convolution. BASS superpixels yield a slightly superior denoising quality to BIST, but both methods dramatically outperform standard convolution. The improved denoising quality of BASS over BIST may be because the DNN module is a single-image method, and so it benefits from BASS's marginally better single-image superpixel quality. The wall-clock runtime of both superpixel convolutions is significantly larger than the wall-clock runtime of standard convolution. However, BIST is 40% faster than BASS.

10. Additional Experiments for BIST

Benchmark Results on DAVIS. In Figure 12, we quantitatively compare BIST with several superpixel methods on the DAVIS dataset. Similar to the SegTrackv2 results (Fig 6), we find that BIST yields excellent superpixel benchmark results. Interestingly, the temporal extent of BIST is actually larger than TSP. We suspect this is due to the more complex dynamics in the DAVIS dataset over the SegTrackv2 dataset. The limitations of BIST that cause a shorter temporal extent are more evident on the SegTrackv2 dataset. However, on the more complex DAVIS dataset, the temporal extent is limited by the motion within the video and so there is not enough simple motion for the limitations of BIST to have a quantitative effect. On this dataset, we did not run StreamingGBH because of its slow wall-clock runtime. The tabular representation of this information for BIST, BASS, and TSP is presented in Table 4.

Conditioned Parameter Estimates. In Table 7, we demonstrate the small but consistent effect of using conditioned parameter estimates. The experiments show that the updated covariance term and prior counts improve the quality of the superpixel metrics, particularly the SA-3D and UE-3D terms. We believe the effect is small since the propagated covariance term ($\widehat{\Sigma}_{t-1,s}$) is still dominated by the circular prior with weight λ_s^2 from the initial estimate in Equation 6. This step is included by default in BIST, but could be conceivably removed to further reduce the wall-clock runtime.

Relabeling Hyperparameters. Table 10 shows the dramatic effect of new superpixel creation by varying the threshold (ε_{new}) on both the number of superpixels and the temporal extent. A smaller value more readily re-classifies superpixels as new, which effectively creates a new set of superpixels for each frame. This keeps the number of active superpixels in each frame small, since there is no temporal consistency enforced. In contrast, a larger threshold makes correcting for errors challenging. This leads to improperly propagated superpixels which restricts the ability for BIST to split/merge superpixels as it erroneously enforces temporal consistency. Overall, this leads to lower-quality superpixels. The highest-quality space-time superpixels are when $\varepsilon_{\text{new}} \approx 0.01$, but we report results using $\varepsilon_{\text{new}} = 0.05$ on the DAVIS dataset for the improved temporal extent so our results are more comparable to TSP. Notably, the re-identification threshold ($\varepsilon_{\text{re-id}}$) has an inconsistent and marginal effect on the result. While conceptually the goal is to re-identify superpixels through occlusion, we suspect more work would be necessary for this step to be meaningful. For example, a proper method should account for the changes in appearance values due to changed viewpoint and object motion, and the predicted new location of the superpixel rather than using the previously identified location.

Boundary Shape Parameters. For completeness, in Table 9 we include results of BIST using a variety of terms which control the shape of the superpixels. A larger appearance variance (σ^2_{app}) and smaller Potts term encourages thin, snake-like superpixels, while the converse encourages rounder, more symmetric superpixels. The flow of thinner superpixels is more difficult to estimate, and this is reflected in the smaller temporal extent. So if superpixels are to be used for tracking, using a larger appearnace variance and Potts term would be beneficial.

The Impact of the Optical Flow Method. Table 8 reports the ablation experiments for different optical flow meth-

Method	# Spix	TEX (%)	SZV	Pool (dB)↑	SA-2D↑	SA-3D↑	UE-2D↓	UE-3D↓
BIST	1223 ± 77	30.2 ± 3.0	120 ± 8	27.84 ± 0.38	0.918 ± 0.017	0.853 ± 0.020	4.90 ± 0.43	8.24 ± 0.75
TSP [3]	1109 ± 16	28.6 ± 3.1	66 ± 3	24.01 ± 0.38	0.873 ± 0.024	0.849 ± 0.023	7.96 ± 0.68	10.14 ± 0.91
BASS [32]	1104 ± 63	-	-	27.59 ± 0.41	0.905 ± 0.018	-	5.88 ± 0.53	-

Table 4. Benchmark metrics on the DAVIS dataset for BIST, BASS, and TSP with standard error (SE = SD/$\sqrt{30}$). UE-2D and UE-3D values are scaled by 10^{-3} for clarity. While the defaults BIST parameters yields about 120 more superpixels than the alternative methods, the benchmark metrics are significantly better as well. In particular, the SA-2D and UE-2D scores of BIST is significantly better than TSP.

Method	# Spix	TEX (%)	SZV	Pool (dB)↑	SA-2D↑	SA-3D↑	UE-2D↓	UE-3D↓
BIST	446 ± 52	49.8 ± 4.8	96 ± 9	30.01 ± 0.67	0.920 ± 0.014	0.827 ± 0.028	4.77 ± 0.47	8.00 ± 0.73
TSP [3]	389 ± 12	53.8 ± 6.0	50 ± 6	27.43 ± 0.60	0.888 ± 0.012	0.819 ± 0.024	7.89 ± 1.00	10.47 ± 1.52
BASS [32]	342 ± 38	-	-	29.63 ± 0.67	0.899 ± 0.018	-	6.00 ± 0.65	-

Table 5. Benchmark metrics on the SegTrackv2 dataset with standard error (SE = SD/$\sqrt{18}$). UE-2D and UE-3D values are scaled by 10^{-3} for clarity.

ods used during the Shift-and-Fill step (Sec 3.1). The best optical flow method is actually the classical optical flow method, instead of the more recent deep learning alternatives. We suspect this is because a space-time superpixel method benefits from tracking appearance values, rather than estimated motion.

Temporally Coherent Split Step. In the main paper, Table 2 shows the value of the proposed temporally coherent split step compared to the naive alternative. Here, Table 6 reports the hyperparameters for each result.

Qualitative Results. Finally, we include several superpixel results for qualitative comparison. Please visit the project page for more examples[2]. In the first set of Figures (13,14,15,16), a reference image is compared with BIST, TSP [3], and BASS [32]. The groundtruth segmentation from the DAVIS dataset is included in the reference image. For each method's image, the superpixels overlapping the grountruth segmentation are colored. For the video methods (BIST and TSP), the overlap is computed only for the first frame, and then tracked over time. For the single-image method (BASS), the overlap is computed at every frame. There are two takeaways. First, BIST and TSP have comparable temporal coherence quality across these sequences. Second, BIST's superpixels adapt to the image content similarly to BASS.

# Spix					
1300		1180		1065	
$\gamma\rho_s$	α	$\gamma\rho_s$	α	$\gamma\rho_s$	α
$(4.0, 10)$	$(0, -5)$	$(4.0, 0.1)$	$(0, -20)$	$(8.0, 1/2)$	$(0, -50)$

Table 6. **Hyperparameters Selection.** This table shows the hyperparameters used to create the results in Table 2. To find these values, we ran a larger grid of negative α values when $\gamma = 0$ and a grid over less extreme α values for $\gamma \in \{4, 8\}$. Once the experiments were executed, we matched experiments with zero gamma values to those with non-zero gamma values based on the average number of superpixels.

[2] https://gauenk.github.io/bist_website/

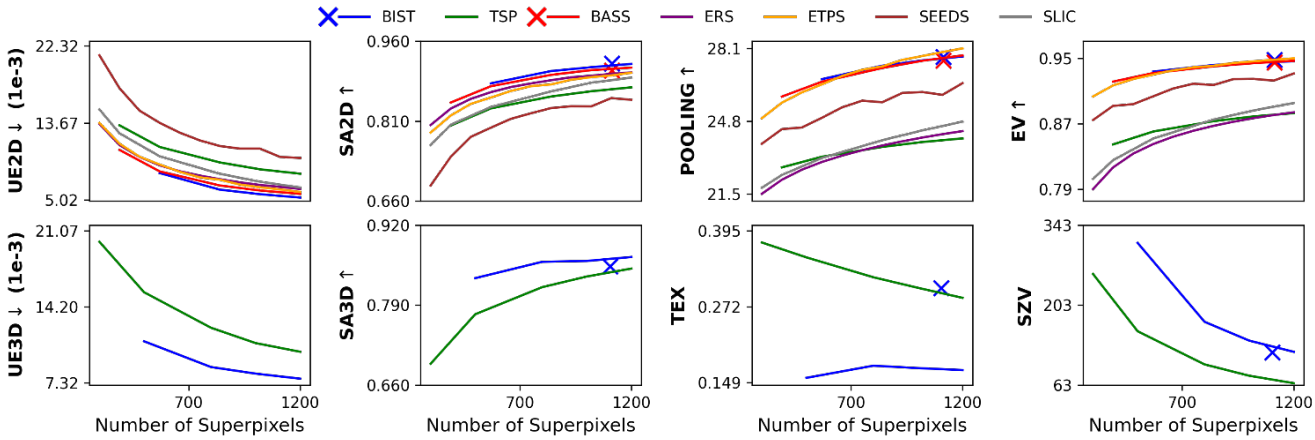

Figure 12. **Quantitative Superpixel Comparison on DAVIS.** This figure quantitatively compares BIST against existing superpixel methods on standard benchmarks. BIST shows exceptional results on the superpixel metrics but suffers from a shorter temporal extend (TEX) than TSP, meaning the superpixels stay alive for fewer frames. As BIST and BASS adaptively estimate the number of superpixels to each scene, their default results are marked as x.

$\alpha_{t,s} \leftarrow n_{t-1,s}$	(Σ_{t-1})	# Spix	TEX (%)	SZV	Pool (dB)↑	SA-2D↑	SA-3D↑	UE-2D↓	UE-3D↓
✗	✗	1232 ± 24	30.4 ± 0.9	127 ± 3	27.83 ± 0.12	0.917 ± 0.005	0.848 ± 0.007	4.97 ± 0.14	8.47 ± 0.25
✗	✓	1236 ± 24	29.8 ± 0.9	121 ± 2	27.81 ± 0.12	0.918 ± 0.005	0.853 ± 0.006	4.92 ± 0.14	8.33 ± 0.23
✓	✗	1226 ± 24	30.4 ± 0.9	127 ± 3	27.85 ± 0.12	0.918 ± 0.005	0.848 ± 0.007	4.95 ± 0.13	8.52 ± 0.25
✓	✓	1224 ± 24	30.4 ± 0.9	121 ± 3	27.85 ± 0.12	0.918 ± 0.005	0.854 ± 0.007	4.87 ± 0.13	8.20 ± 0.23

Table 7. **Conditional Parameter Updates.** This table reports the effect of using conditioned parameter estimates in the boundary update step (Sec 8.2). The left column indicates if the prior superpixel sizes were updated to the count from the previous timestep, and the next column indicates if the prior spatial covariance term was used. Both the SA-3D and UE-3D metrics are improved when using both modifications together, while the 2D metrics remain consistent across configurations. The scale for UE-2D and UE-3D is 10^{-3}. The results are reported on the DAVIS dataset with standard errors computed from over 10 runs; (SE = SD/$\sqrt{30 \cdot 10}$).

Flow	# Spix	TEX (%)	SZV	Pool (dB)↑	SA-2D↑	SA-3D↑	UE-2D↓	UE-3D↓
C. Liu [3]	393 ± 47	49.9 ± 4.9	104 ± 11	29.89 ± 0.66	0.917 ± 0.015	0.827 ± 0.028	5.01 ± 0.53	8.15 ± 0.83
SpyNet [22]	377 ± 42	43.0 ± 6.1	103 ± 11	29.72 ± 0.68	0.911 ± 0.017	0.806 ± 0.039	5.21 ± 0.55	8.51 ± 0.77
RAFT [31]	374 ± 40	43.7 ± 6.2	104 ± 11	29.63 ± 0.69	0.909 ± 0.017	0.790 ± 0.042	5.33 ± 0.57	8.96 ± 0.82

Table 8. **Ablation of the Optical Flow Method.** This table reports the effect of using different optical flow algorithms with BIST. The Default flow represents our implementation, while SPyNet and RAFT are alternative methods. The SA-3D metric shows significant differences between methods, with our default implementation achieving the best performance. The scale for UE-2D and UE-3D is 10^{-3}. The results are reported on the SegTrackv2 dataset with standard errors; (SE = SD/$\sqrt{18}$).

σ^2_{app}	β	# Spix	TEX (%)	SZV	Pool (dB)↑	SA-2D↑	SA-3D↑	UE-2D↓	UE-3D↓
0.0045	1.0	1784 ± 118	28.2 ± 2.7	103 ± 7	28.84 ± 0.42	0.926 ± 0.016	0.855 ± 0.019	4.45 ± 0.39	7.75 ± 0.71
0.0045	10.0	1802 ± 118	28.9 ± 2.8	94 ± 6	28.83 ± 0.41	0.926 ± 0.016	0.857 ± 0.020	4.33 ± 0.39	7.52 ± 0.69
0.0045	20.0	1820 ± 118	29.4 ± 2.9	90 ± 6	28.81 ± 0.41	0.928 ± 0.016	0.860 ± 0.019	4.30 ± 0.38	7.38 ± 0.68
0.0090	1.0	1197 ± 77	28.8 ± 2.7	131 ± 9	27.97 ± 0.39	0.916 ± 0.017	0.851 ± 0.020	5.01 ± 0.44	8.57 ± 0.79
0.0090	10.0	1224 ± 77	30.1 ± 3.0	119 ± 8	27.84 ± 0.38	0.918 ± 0.017	0.855 ± 0.019	4.91 ± 0.43	8.31 ± 0.76
0.0090	20.0	1249 ± 77	30.8 ± 3.1	115 ± 8	27.69 ± 0.37	0.918 ± 0.017	0.856 ± 0.020	4.86 ± 0.42	8.07 ± 0.72
0.0180	1.0	867 ± 45	31.4 ± 2.8	154 ± 9	26.83 ± 0.37	0.905 ± 0.019	0.839 ± 0.021	5.70 ± 0.50	9.57 ± 0.88
0.0180	10.0	902 ± 50	33.6 ± 3.2	151 ± 10	26.47 ± 0.36	0.909 ± 0.018	0.842 ± 0.021	5.55 ± 0.49	9.16 ± 0.87
0.0180	20.0	925 ± 53	33.6 ± 3.2	149 ± 10	26.17 ± 0.35	0.906 ± 0.019	0.844 ± 0.021	5.56 ± 0.50	9.14 ± 0.87

Table 9. **Ablation of BIST Boundary Shape Parameters.** This table reports the effect of varying the appearance variance parameter σ^2_{app} and the boundary smoothness term β. Higher σ^2_{app} values lead to fewer but larger superpixels, while higher β values create smoother boundaries and improve temporal consistency (SA-3D and UE-3D). The highest pooling score is achieved with low σ^2_{app} and high β. The scale for UE-2D and UE-3D is 10^{-3}. Results are reported on the DAVIS dataset with standard errors; (SE = SD/$\sqrt{30}$).

ε_{new}	ε_{re-id} (10^{-6})	# Spix	TEX (%)	SZV	Pool (dB)↑	SA-2D↑	SA-3D↑	UE-2D↓	UE-3D↓
0.001	0.1	770 ± 51	4.8 ± 1.1	86 ± 6	26.71 ± 0.42	0.895 ± 0.020	0.911 ± 0.018	6.38 ± 0.54	6.47 ± 0.55
0.001	1.0	762 ± 50	4.8 ± 1.1	87 ± 7	26.63 ± 0.42	0.895 ± 0.020	0.911 ± 0.017	6.41 ± 0.53	6.50 ± 0.54
0.001	10.0	769 ± 51	4.8 ± 1.1	92 ± 7	26.68 ± 0.42	0.896 ± 0.020	0.910 ± 0.018	6.37 ± 0.53	6.46 ± 0.53
0.01	0.1	1059 ± 68	16.1 ± 2.4	102 ± 7	27.58 ± 0.37	0.911 ± 0.018	0.912 ± 0.016	5.28 ± 0.45	6.21 ± 0.54
0.01	1.0	1058 ± 68	16.1 ± 2.4	103 ± 7	27.58 ± 0.38	0.912 ± 0.018	0.912 ± 0.016	5.27 ± 0.45	6.17 ± 0.53
0.01	10.0	1055 ± 68	16.2 ± 2.4	102 ± 7	27.56 ± 0.38	0.913 ± 0.018	0.913 ± 0.015	5.29 ± 0.46	6.25 ± 0.56
0.05	0.1	1223 ± 78	30.3 ± 3.0	121 ± 8	27.84 ± 0.38	0.918 ± 0.018	0.854 ± 0.021	4.88 ± 0.42	8.29 ± 0.77
0.05	1.0	1223 ± 78	30.3 ± 3.0	120 ± 8	27.84 ± 0.38	0.917 ± 0.018	0.852 ± 0.021	4.90 ± 0.42	8.28 ± 0.73
0.05	10.0	1225 ± 76	30.1 ± 2.9	120 ± 8	27.85 ± 0.38	0.917 ± 0.017	0.856 ± 0.020	4.90 ± 0.42	8.23 ± 0.75
0.1	0.1	1277 ± 81	35.2 ± 3.0	126 ± 8	27.90 ± 0.38	0.920 ± 0.017	0.807 ± 0.026	4.78 ± 0.41	9.37 ± 0.83
0.1	1.0	1276 ± 82	35.4 ± 3.0	126 ± 8	27.90 ± 0.38	0.918 ± 0.017	0.808 ± 0.025	4.81 ± 0.43	9.34 ± 0.84
0.1	10.0	1274 ± 82	34.8 ± 2.9	126 ± 8	27.89 ± 0.38	0.918 ± 0.018	0.807 ± 0.026	4.76 ± 0.41	9.16 ± 0.79

Table 10. **Ablation of BIST Relabeling Parameters.** This table reports the effect of varying the new superpixel threshold ε_{new} and the re-identification threshold ε_{re-id}. Higher ε_{new} values increase the number of superpixels and improve most metrics. The relabeling threshold has minimal impact on performance. The scale for UE-2D and UE-3D is 10^{-3}. Results are reported on the DAVIS dataset with standard errors; (SE = SD/$\sqrt{30}$).

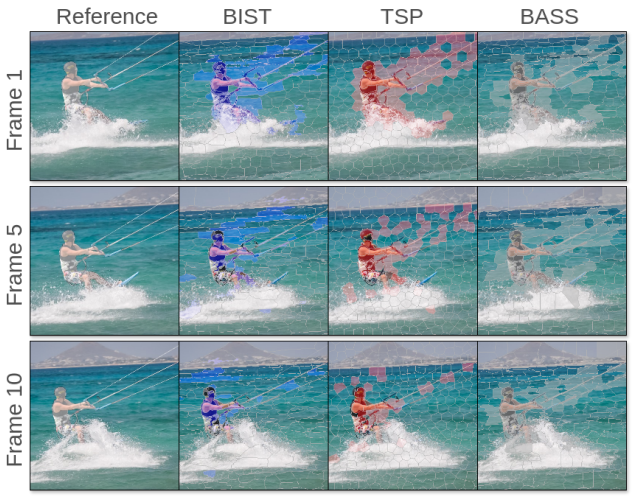

Figure 13. **Kite-Surf from DAVIS.**

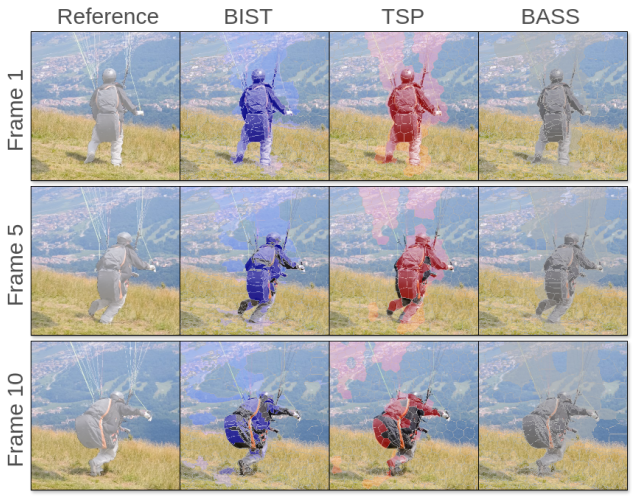

Figure 14. **Paragliding-Launch from DAVIS.**

Figure 15. **Dance-Twirl from DAVIS.**

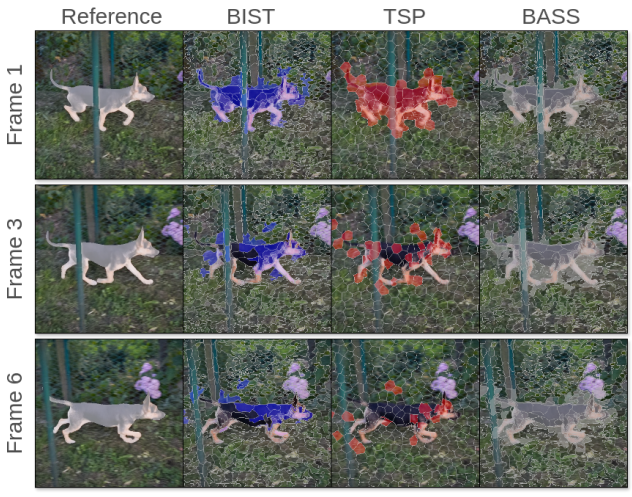

Figure 16. **Libby from DAVIS.**

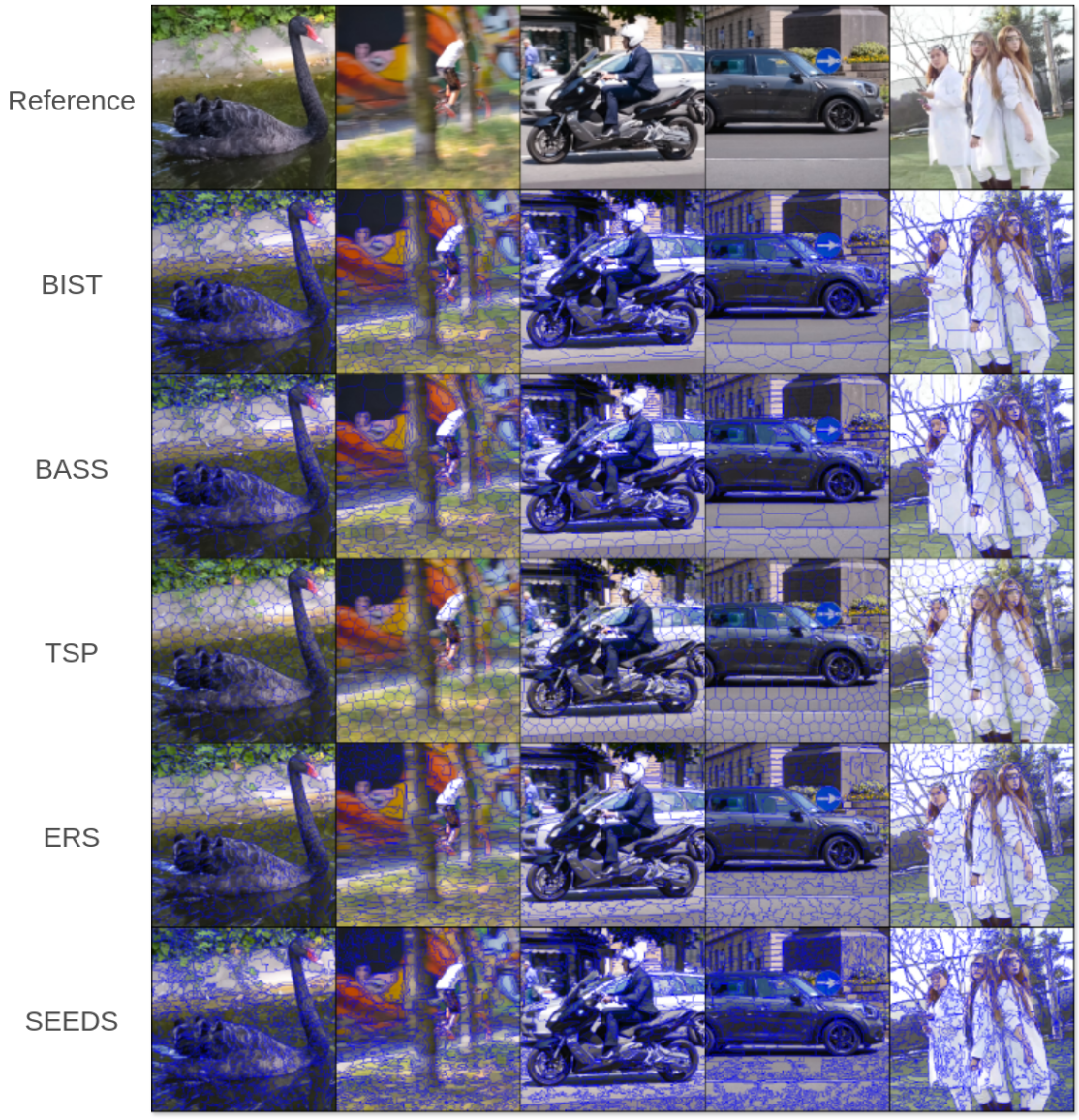

Figure 17. **Qualitatively Compare Superpixel Results on DAVIS.** Visualized are the superpixels estimated for a frame in the middle of five sequences from the DAVIS dataset. See the supplemental material for png files.

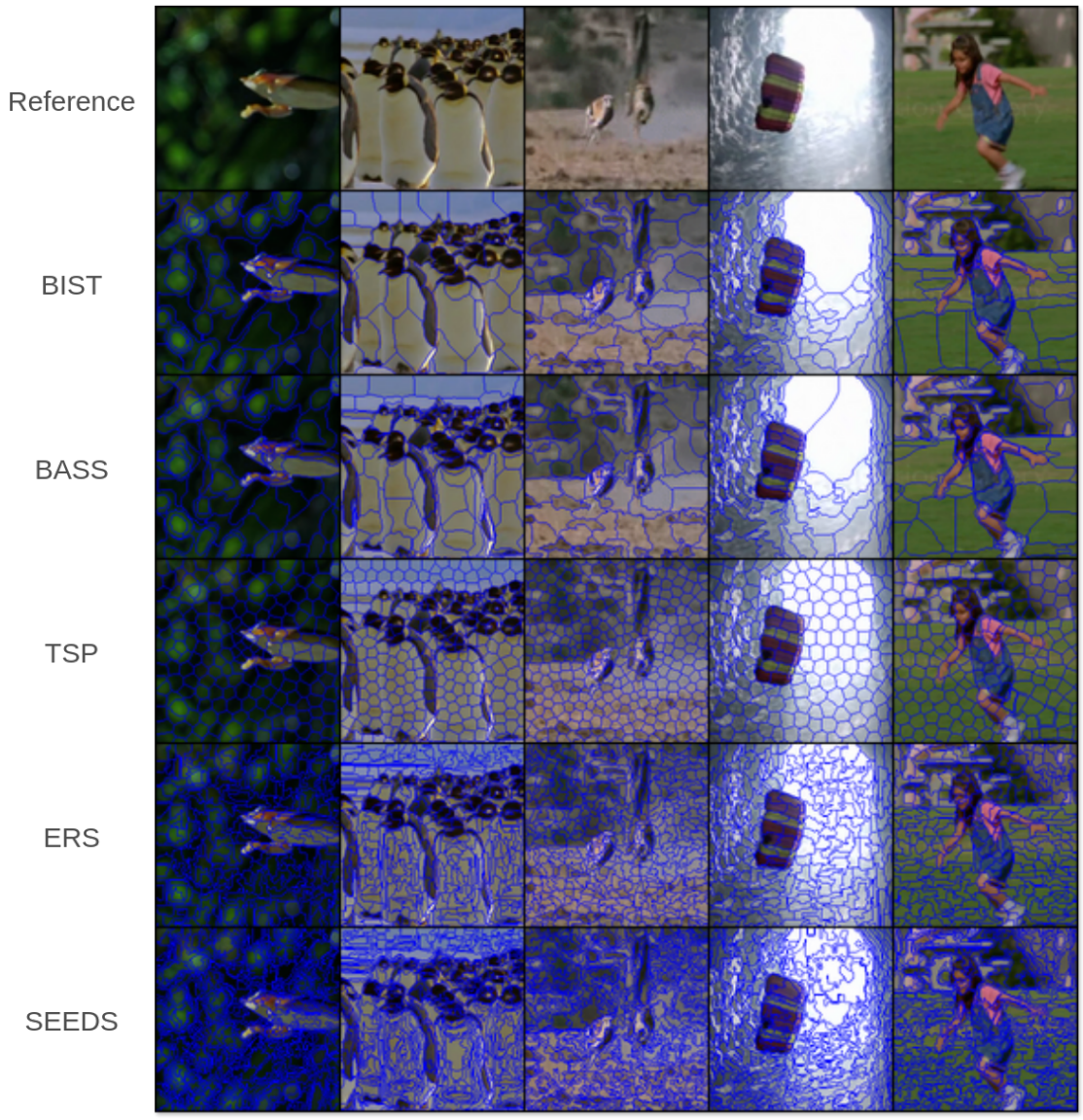

Figure 18. **Qualitatively Compare Superpixel Results on SegTrackv2.** Visualized are the superpixels estimated for a frame in the middle of five sequences from the SegTrackv2 dataset. See the supplemental material for png files.