Salient Object Detection for Images Taken by People With Vision Impairments - Supplementary Materials

Jarek Reynolds*, Chandra Kanth Nagesh*, and Danna Gurari
* denotes equal contribution
University of Colorado Boulder

This document supplements the main paper with additional information concerning:

1. Dataset Creation (supplements Section 3.1)
    - Annotation Task Interface
    - Worker Qualification Task
    - Analysis of Workers' Annotation Differences
2. Experimental Design (supplements Section 4.1)
3. Experimental Results (supplements Sections 4.2-4.4)

A. Dataset Creation

A.1. Annotation Task Interface

The task interface displays five images within a tabbed container on the left and preliminary questions with task instructions on the right. A screenshot of the task interface (without instructions) is shown in Figure 1.

To account for occlusions and holes while keeping the task simple for annotators, we permitted annotators to generate multiple polygons. For occlusions, annotators could use as many polygons as necessary for demarcating foreground objects partitioned into multiple polygons. For holes, we apply an even-odd fill rule to images featuring foreground objects with holes. With an even-odd fill rule, every area inside an even number of enclosed areas becomes hollow, and every region inside an odd number of enclosed areas becomes filled [11]. By treating the image's four corners as the first enclosed area, the outermost boundary of the foreground object becomes the second enclosed area. Moreover, holes within foreground objects represent the third layer of enclosed areas and become filled, allowing annotators to demarcate foreground objects featuring holes. In practice, annotators first trace the outermost boundary of the foreground object and close the path by clicking the first point a second time. We then instructed annotators to trace any holes within the foreground object, and so those holes end up in odd-numbered layers.

A.2. Worker Qualification Task

We administered a qualification task for workers to support our collection of high-quality ground truth annotations. The qualification task required annotating five images, each of which features a distinct challenging annotation scenario. All five images are shown in Figure 2. The first two images show a table and a bench, offering examples with complex boundaries and holes. The next two images feature a person holding a coffee mug, to support educating a crowdworker about our expectations for annotating objects with complex geometries that have many curves and occlusions that require annotating multiple polygons. The final image is a spatula. This task verified a crowdworker's ability to correctly identify and annotate multiple holes that can arise within the salient object.

After crowdworkers annotated each qualification image, the backend code of our website checked if their annotation was sufficiently similar to the GT annotation (i.e., IoU similarity of at least 0.90). Crowdworkers could only proceed to the following image after they obtained an IoU ≥ 0.90 on the current image. Crowdworkers obtaining an IoU ≥ 0.90 on all five qualification assessment images on a per-image basis gave us substantial confidence that they would be able to successfully handle complex and challenging outlier cases within the original VizWiz Dataset.[1]

A.3. Analysis of Workers' Annotation Differences

We collected a larger number of redundant annotations per image for a random subset of images to better explore when and why annotation differences are observed from different workers. Specifically, for this analysis, we collected four annotations as opposed to two for a subset of 1,237 images. Examples of the redundant annotations collected per image are shown in Figure 3.

The first example (i.e., row 1 of Figure 3) highlights that

[1] Some crowdworkers did not pass the qualification assessment due to time constraints. In these cases, crowdworkers would contact us with the images they annotated. If we were confident in their annotation abilities, we manually added these crowdworkers to the qualified worker pool.

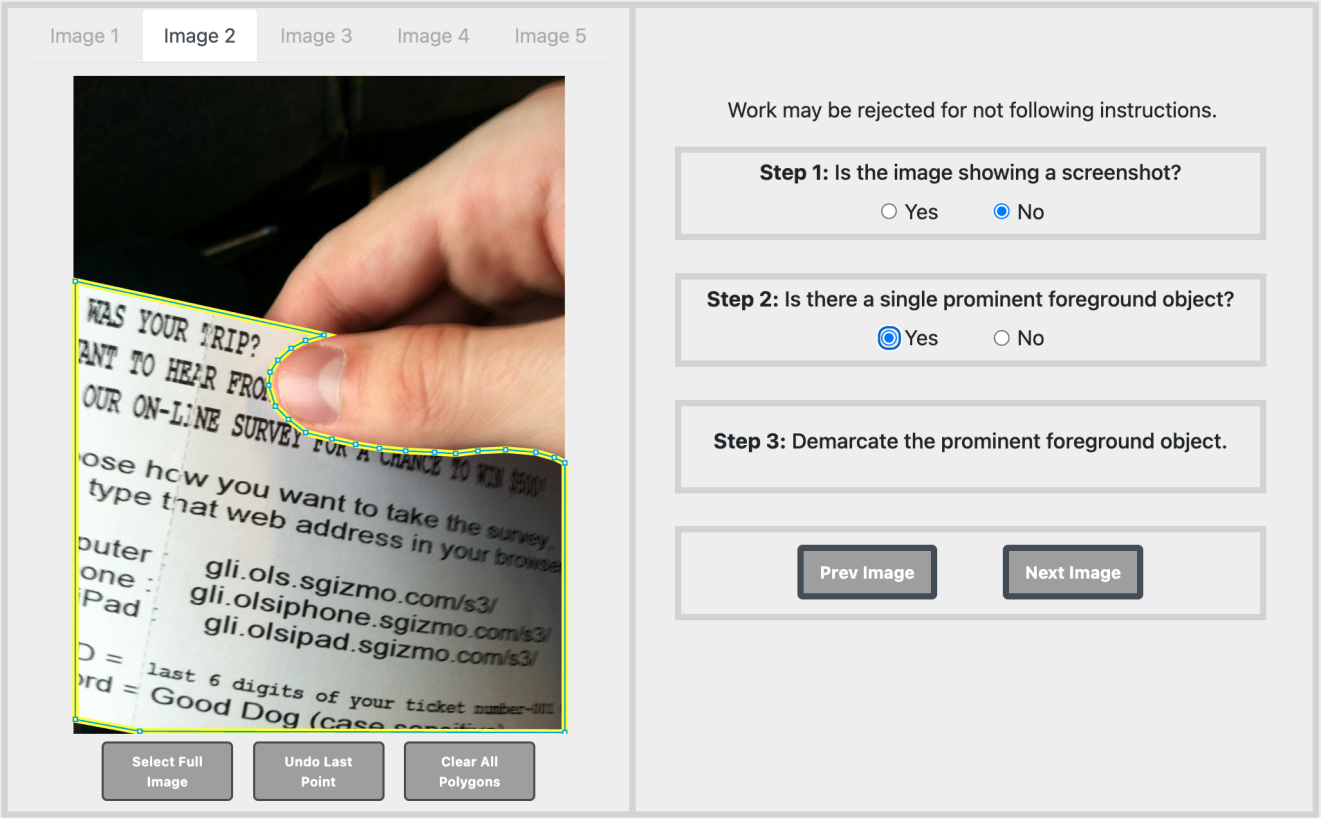

Figure 1. A screenshot of our annotation task interface.

Figure 2. The five images used for the worker qualification task. Each was selected to demonstrate a challenging annotation scenario such as complex boundaries, holes, and occlusions.

annotation differences can stem from challenging annotation scenarios where objects contain holes (e.g., in mug handle) or are occluded (e.g., by the straw). For instance, the hole was not annotated in the third annotation. Additionally, only the fourth annotation captured the occlusion that arises from the straw.

The second example (i.e., row 2 of Figure 3) highlights that annotation differences can stem from ambiguity regarding what is the salient object. As shown, the first two annotations flag the image as lacking a foreground object, the third annotation identifies the child holding the cup as the salient object, and the fourth annotation identified the child's cup as the salient object.

The third example (i.e., in row 3 of Figure 3) highlights that annotation differences also can arise for objects that simultaneously have complex boundaries and holes. In annotation one, the worker did not fully annotate the salient object, cutting out part of the object from the annotation. Only the third and fourth annotations accurately annotate all holes that are present in the salient object's boundary while also having tight boundaries in the annotation.

In summary, we found occlusions, holes, and saliency ambiguity to be the primary factors contributing to annotation differences. In the case of occlusions, worker differences can arise when deciding whether to include objects that are a composite part of the salient object. In the case of holes, annotation differences can arise regarding which holes to annotate. Last, we found that it can be ambiguous as to which object is the most salient. To facilitate future analysis of human performance, we will publicly share metadata with all humans' annotations for all images.

B. Experimental Design

We compute the five metrics used in the benchmarking section using the following definitions:

Mean Absolute Error [12] represents the average absolute difference between the predicted saliency map and its

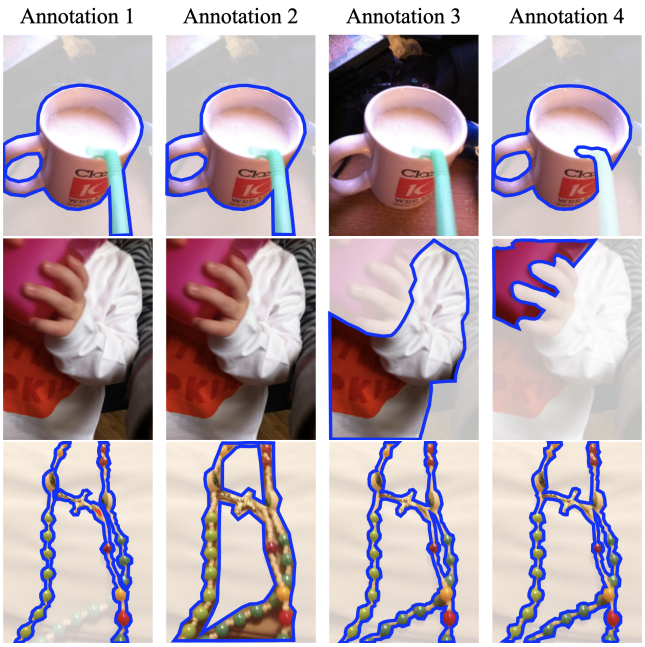

Figure 3. Example annotations from our random subset where we collected four annotations as opposed to two. We find worker differences primarily occur in challenging annotation scenarios such as holes, occlusions, complex boundaries, and object saliency.

ground truth per pixel. It can be given as:

$$MAE = \frac{1}{H * W} \sum_{r=1}^{H} \sum_{c=1}^{W} |pred(r,c) - gt(r,c)| \quad (1)$$

where $pred$ represents the predicted saliency map, gt represents the ground truth, (H, W) represents the height and width of the image, and (r, c) represents the pixel coordinates for the given image.

Structure Measure [3] is used to measure the similarity between the predicted saliency map and the ground truth. Since, we convert both the predictions and ground truths into the $[0, 1]$ range, we apply the formula directly to the predictions and maps. It can defined as follows:

$$S_m = (1 - \alpha)S_r + \alpha S_o \quad (2)$$

where, S_r is defined as the region aware similarity score, S_o is defined as the object aware similarity score, and α represents the weight that is used to sum up the values. We set $\alpha = 0.5$, therefore making sure that we see equal contribution from both region and object aware scores.

F-Measure [1] represents the precision and recall ratio for the given prediction. It can be represented as:

$$F_m = \frac{(1 + \beta^2) * Precision * Recall}{\beta^2 * Precision + Recall} \quad (3)$$

Here $precision = \frac{TP}{TP+FP}$ and $recall = \frac{TP}{TP+FN}$ on the entire prediction image by pixels. We set $\beta^2 = 0.3$ and report the average of all F-measures as F_m similar to previous works.

Enhanced Alignment Measure [4] is used as the metric to measure the effectiveness of the saliency prediction against the ground truth. It captures the pixel-level matching information and image-level statistics into one single metric by the means of an enhanced alignment matrix ϕ. It is defined as follows:

$$E_m = \frac{1}{H * W} \sum_{r=1}^{H} \sum_{c=1}^{W} \phi_{FM}(r,c) \quad (4)$$

where, ϕ_{FM} represents the enhanced alignment matrix for the foreground map, (H, W) represents the height and width of the image, and (r, c) represents the pixel coordinates for the given image.

Intersection over Union also known as Jaccard Index is used to determine the similarity between sample sets. In this case it captures the overlap between the ground truth and prediction map of the salient object. We convert the predictions in binary map and compute the Jaccard Index over two classes. It can be defined as follows:

$$IoU = J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (5)$$

where, A and B are images of same size, consisting of integer class values $\{0, 1\}$.

C. Algorithm Benchmarking

We provide more details about our algorithm benchmarking here. First, we report each model's backbone in Table 1. Second, we show results for SOD models mentioned in the paper that are older. Of note, for fine-tuning the InSPyReNet model and training the InSPyReNet model from scratch using VizWiz-SO and DUTS+VizWiz-SO, we modify the training hyperparameters to fit the GPU requirements available to us. Specifically, we reduce the `batchsize` to 4, `num_worker` to 4, `epochs` to 40, and `warmup_iterations` to 1000. We also report results for three variants of the second-best model, VST [15]: (1) pretrained model fine-tuned on VizWiz-SO (VST-FT), (2) algorithm trained from scratch on VizWiz-SO (VST-S), and (3) algorithm trained from scratch on DUTS [17] and VizWiz-SO (VST-DS). Overall results are shown in Table 2 and fine-grained analysis of these models are shown in Table 3.

We show qualitative examples for these models on VizWiz-SO in Figures 4 and 5. These examples feature a variety of challenges we observed for the models in our fine-grained analysis. Most models perform poorly in identifying larger salient objects (rows 4 and 5 in Figure 4 and

	HP	VST [8]	PGNet [19]	DIS [13]	ICON [20]	TRACER [7]	IPR [6]
Backbone	-	T2T-ViT	R-18+Swin	U2Net	Swin	ENet-7	Swin

Table 1. Details of the various backbones used by the algorithms used for benchmarking. (ViT=Vision Transformer [2]; R=ResNet [5]; Swin=Shifted window transformer [9]; ENet=EfficientNet [16])

		BASNet [14]	F3Net [18]	U2Net [15]	PFSNet [10]	VST-FT	VST-S	VST-DS
Attr.	Backbone	R-34	R-50	-	R-50	ViT	ViT	ViT
	Training set	D	D	D	VW	VW	D+VW	D
	Input size	256^2	352^2	320^2	352^2	224^2	224^2	224^2
	Size (MB)	333	98	4.7	120	171	171	171
VizWiz-SO	$MAE \downarrow$	0.28	0.28	0.26	0.32	0.19	0.21	0.23
	$S_m \uparrow$	0.59	0.55	0.61	0.48	0.64	0.63	0.58
	$F_m \uparrow$	0.77	0.74	0.80	0.70	0.74	0.72	0.68
	$E_m \uparrow$	0.64	0.65	0.65	0.60	0.77	0.70	0.70
	$IoU \uparrow$	0.62	0.53	0.63	0.48	0.70	0.69	0.64

Table 2. Quantitative comparison of off-the-shelf models (which are cited) as well as the VST model after being fine-tuned (-FT), trained from scratch on our VizWiz-SO dataset (-S), and trained from scratch on both DUTS and VizWiz-SO datasets (-DS). D=DUTS-TR [17]; VW=VizWiz-SO; R=ResNet [5]; VST=Visual Saliency Transformer [8]; ViT=Vision Transformer [2]

		BASNet [14]	F3Net [18]	U2Net [15]	PFSNet [10]	VST-FT	VST-S	VST-DS
Text Present	True	0.23	0.22	0.22	0.25	0.16	0.17	0.18
	False	0.35	0.38	0.32	0.42	0.24	0.26	0.28
Coverage	Small	0.06	0.16	0.07	0.16	0.09	0.11	0.14
	Medium	0.15	0.20	0.15	0.24	0.09	0.10	0.11
	Large	0.60	0.47	0.54	0.54	0.38	0.39	0.40
Boundary	High	0.15	0.21	0.15	0.24	0.11	0.12	0.12
	Low	0.38	0.34	0.35	0.38	0.26	0.27	0.28
Resolution	High	0.30	0.30	0.28	0.33	0.17	0.18	0.19
	Low	0.26	0.27	0.26	0.31	0.19	0.20	0.21
Quality	Good	0.22	0.23	0.21	0.26	0.16	0.17	0.19
	Poor	0.44	0.43	0.41	0.47	0.30	0.31	0.33

Table 3. Fine-grained analysis of off-the-shelf models (which are cited) as well as the VST model after being fine-tuned (-FT), trained from scratch on our VizWiz-SO dataset (-S), and trained from scratch on both DUTS and VizWiz-SO datasets (-DS). This covers analysis with respect to presence of text on the salient object ("Text Present"), relative size of the salient object in the image ("Coverage"), relative complexity of the salient object's boundary ("Boundary"), and image quality ("Quality") using the $MAE \downarrow$ metric. As shown, the models perform worse when salient objects lack text, occupy a large portion of the image, and have less complex boundaries as well as when the image quality is poor.

row 1 in Figure 5), but perform relatively well on images with smaller salient objects (row 2 in Figure 5). We also observe the most models perform better when salient objects contain text (rows 1 and 2 in Figure 4 and row 3 in Figure 5) versus lack text (rows 5 and 6 in Figure 4 and row 4 in Figure 5). Further, we see most models perform worse for images with complex boundaries (row 5 in Figure 5) and that are lower quality (rows 3, 4, and 5 in Figure 4 and rows 6 and 7 in Figure 5).

References

[1] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Süsstrunk. Frequency-tuned salient region detection. In *CVPR*, number CONF, pages 1597–1604, 2009. 3

[2] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is

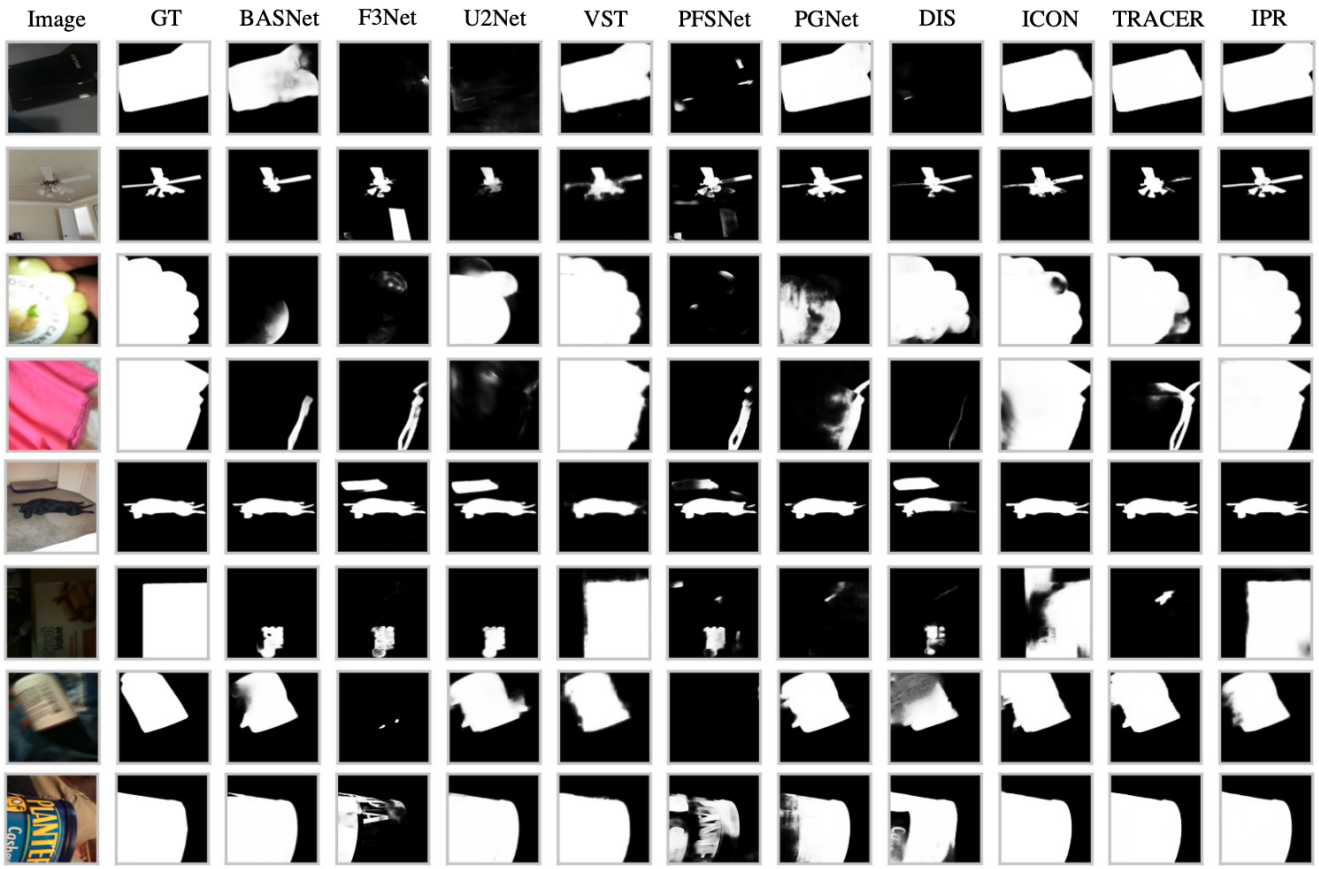

Figure 4. Examples of images with characteristics such as high coverage ratio, presence of text, less complex boundaries, and lower image quality. We show how the ten models perform on these cases as compared to the human annotation (GT=Ground Truth). We see that models such as PFSNet [10], DIS [13], and F3Net [18] do not always give us the correct salient objects or sometime no predictions at all. We also notice that VST [8], ICON [20], and InSPyReNet [6] usually predicts salient objects with better accuracy compared to other models. InSPyReNet is the closest to human performance across all the various images presented here. (HP=Human Performance, IPR=InSPyReNet)

worth 16x16 words: Transformers for image recognition at scale, 2020. 4

[3] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, pages 4548–4557, 2017. 3

[4] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In *IJCAI*, pages 698–704, 2018. 3

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 4

[6] Taehun Kim, Kunhee Kim, Joonyeong Lee, Dongmin Cha, Jiho Lee, and Daijin Kim. Revisiting image pyramid structure for high resolution salient object detection. In *Proceedings of the Asian Conference on Computer Vision*, pages 108–124, 2022. 4, 5, 6

[7] Min Seok Lee, Wooseok Shin, and Sung Won Han. Tracer: Extreme attention guided salient object tracing network, 2022. 4

[8] Nian Liu, Ni Zhang, Kaiyuan Wan, Ling Shao, and Junwei Han. Visual saliency transformer, 2021. 4, 5

[9] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021. 4

[10] Mingcan Ma, Changqun Xia, and Jia Li. Pyramidal feature shrinking for salient object detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(3):2311–2318, May 2021. 4, 5

[11] MDN. Fill-rule - svg: Scalable vector graphics: Mdn. 1

[12] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, pages 733–740, 2012. 2

[13] Xuebin Qin, Hang Dai, Xiaobin Hu, Deng-Ping Fan, Ling Shao, and Luc Van Gool. Highly accurate dichotomous image segmentation, 2022. 4, 5

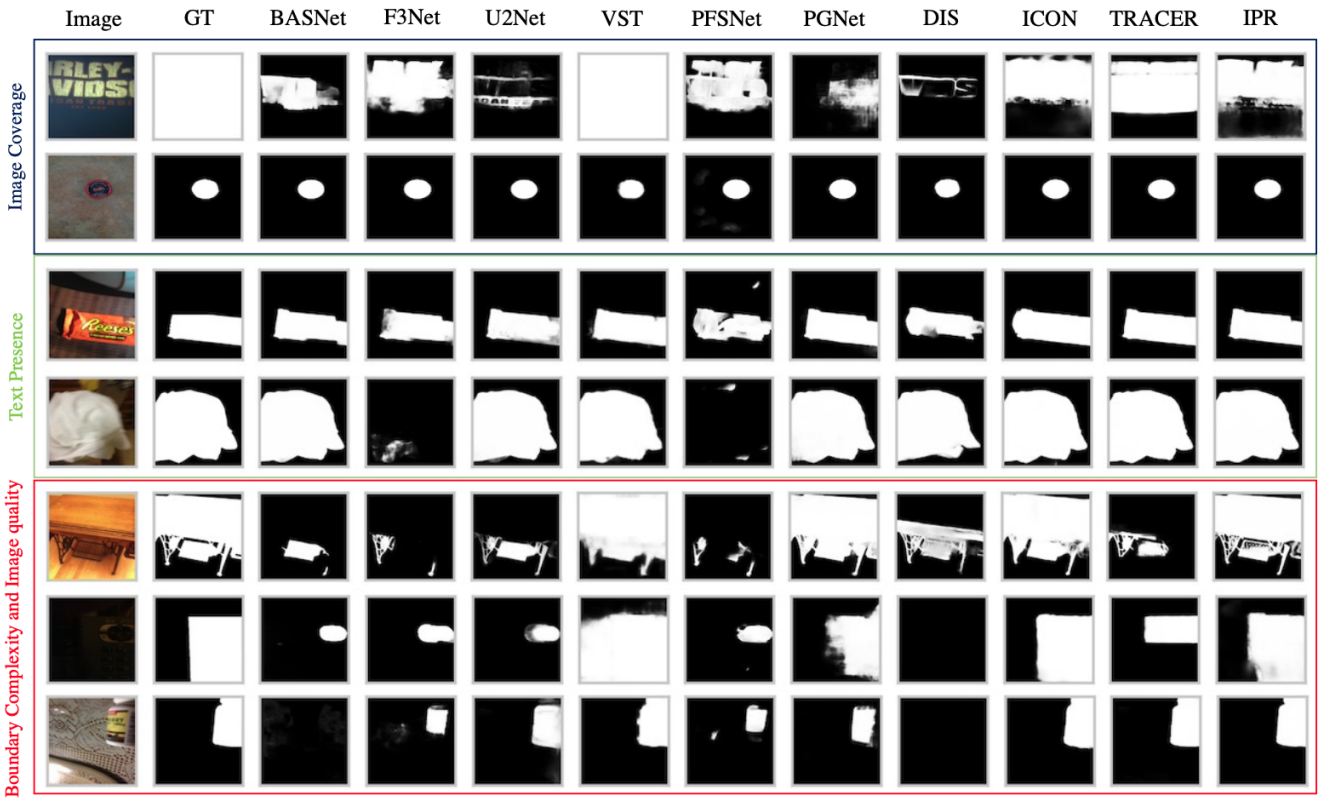

Figure 5. Examples of images with characteristics with respect to our fine-grained analysis that we found are challenging for modern salient object detection models. We show how the ten models perform on these cases as compared to the human annotations (GT=Ground Truth). We see that all models except InSPyReNet [6] seem to perform poorly for objects with high coverage ratio and well for objects with lower coverage ratio. Next, we see for presence of text, that models do not perform worse when there is no text on the salient object. Finally, we see that all models except InSPyReNet [6] can suffer from not detecting the correct salient object for salient objects with complex boundaries and for images with quality issues. Overall, InSPyReNet [6] is the closest to the human performance. (IPR=InSPyReNet)

[14] Xuebin Qin, Deng-Ping Fan, Chenyang Huang, Cyril Diagne, Zichen Zhang, Adrià Cabeza Sant'Anna, Albert Suàrez, Martin Jagersand, and Ling Shao. Boundary-aware segmentation network for mobile and web applications, 2021. 4

[15] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R. Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. *Pattern Recognition*, 106:107404, 2020. 3, 4

[16] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020. 4

[17] Wang, Lijun, Lu, Huchuan, Wang, Yifan, Feng, Mengyang, Wang, Dong, Yin, Baocai, Ruan, and Xiang. Learning to detect salient objects with image-level supervision. In *CVPR*, 2017. 3, 4

[18] Jun Wei, Shuhui Wang, and Qingming Huang. F3net: Fusion, feedback and focus for salient object detection, 2019. 4, 5

[19] Chenxi Xie, Changqun Xia, Mingcan Ma, Zhirui Zhao, Xiaowu Chen, and Jia Li. Pyramid grafting network for one-stage high resolution saliency detection, 2022. 4

[20] Mingchen Zhuge, Deng-Ping Fan, Nian Liu, Dingwen Zhang, Dong Xu, and Ling Shao. Salient object detection via integrity learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 4, 5