Inferring Attention Shift Ranks of Objects for Image Saliency
Supplementary Material

Avishek Siris[1], Jianbo Jiao[2], Gary K.L. Tam[1], Xianghua Xie[1], Rynson W.H. Lau[3]
Department of Computer Science, Swansea University[1]
Department of Engineering Science, University of Oxford[2] and City University of Hong Kong[3]
a.siris.789605@swansea.ac.uk, jianbo@robots.ox.ac.uk,
{k.l.tam, x.xie}@swansea.ac.uk, rynson.lau@cityu.edu.hk

In this supplementary material, we provide more details and comparisons of our implementation. These include:

- A more detailed description of the data collection process: the ground-truth generation methods (Sec. 1.1), the user study and participants (Sec. 1.2), and the final dataset analysis (Sec. 1.3),
- Some additional details of our implementation and design rationale (Sec. 2),
- More details of the effectiveness of our model (Sec. 3), and
- Some additional details when evaluating against state-of-the-arts [1, 2, 6, 7, 8] (Sec. 4), and further comparison with [2] (Sec. 5)

1. Saliency Rank Dataset from Attention Shift

First, we provide further details of the three main approaches that we propose to generate our ground-truth saliency rank annotations, and our user study.

1.1. Data Collection

To our knowledge, there are no large-scale dataset available for salient object ranking based on *attention shift*. Hence, we propose a new large-scale salient object ranking dataset, by combining the widely used MS-COCO dataset [5] with the SALICON dataset [3]. MS-COCO contains complex images with ground-truth object segmentation, whilst SALICON is built on top of MS-COCO to provide mouse-trajectory based fixations. The SALICON dataset [3] provides two sources of fixation data: 1) fixation point sequences and 2) fixation maps for each image. We exploit these two sources and consider three main approaches to generate our ground-truth saliency rank annotations.

Approach 1: For a given image, we follow each of the fixation points in a fixation sequence and assign descending saliency scores to the fixated image pixels. We repeat this scoring of pixels over all observer fixation data. The saliency rank of an object can be computed by aggregating these saliency scores that the object contains (*i.e.*, the higher the aggregated scores, the more salient the object and the higher its rank). The number of fixation points varies among observers and leads to a large difference in scores.

We first assign scores to pixel values using fixation points from the SALICON [3] dataset. Then we get the score for objects based on the values of pixels that belong to those objects. More specifically, for every image $I \in \mathbb{R}^{W \times H}$ of dimension $W \times H$, there are N number of observers. Let F^j be the fixation sequence obtained from one of the N observers $j \in [1, N]$ and a fixation f_i^j with index order $i \in [1, t]$ that represents the i^{th} fixation in the sequence F^j of length t. We then assign a score to image pixel p if the fixation f_i^j falls on p using:

$$v_p = \sum_{j}^{N} \sum_{i}^{t} g(f_i^j), \quad \text{if } f_i^j = p, \tag{1}$$

$$g(f_i^t) = 1 - \frac{i}{t}, \tag{2}$$

where v_p denotes the score at a pixel $p \in I$ aggregating from all N observers' fixation data. The function g takes the temporal order i^{th} of a fixation point in the sequence into account, and assigns lower values to fixation points if they are latter in the sequence.

Note that we are interested in the importance of the order of fixation points. We thus do not take into account the duration of fixation points in our formulation. There are large variances in the duration of fixations among different observers. Considering the durations of fixation points would cause the scoring to fluctuate greatly. Further, it is difficult (if not impossible) to obtain the exact duration of each fixation point whilst the fixations are obtained from a re-sampling process [3]. In contrast, using the order of fixation points would ensure that there is a consistent gap between the scores of each pair of consecutive fixation points, and lead to higher stability in the final object scoring.

Next, we try to accommodate the varying sizes of objects in an image. Larger objects may collect more fixations from observers and be considered more salient with higher ranks. However, small objects that are rare may also be more salient even if there are fewer fixations. We do not know which methods would reflect how humans rank multiple objects in term of saliency. We try four methods to aggregate scores for subsequent saliency ranks of objects, namely: *FixSeq-avg* (average score), *FixSeq-max* (maximum score), FixSeq-avgPmax (average + maximum score) and *FixSeq-avgMmax* (average $\times$ maximum score).

Let o be one of the objects in an image I, $|o|$ be the number of pixels in o, and v_p^o be the score of a pixel $p \in o$ inside an object. We define:

$$\textit{FixSeq-avg}(o, I) = \frac{1}{|o|} \sum_{p \in o} v_p^o, \tag{3}$$

$$\textit{FixSeq-max}(o, I) = \max_{p \in o}(v_p^o), \tag{4}$$

$$\begin{aligned} \textit{FixSeq-avgPmax}(o, I) = \textit{FixSeq-avg}(o, I) \\ + \textit{FixSeq-max}(o, I), \end{aligned} \tag{5}$$

$$\begin{aligned} \textit{FixSeq-avgMmax}(o, I) = \textit{FixSeq-avg}(o, I) \\ \times \textit{FixSeq-max}(o, I). \end{aligned} \tag{6}$$

For a given image, *FixSeq-avg* (Eq. 3) calculates the final score of an object by taking the average values of pixels belonging to the object. It takes into account the size differences between objects. In *FixSeq-max* (Eq. 4), the final score of an object is the maximum value v_p^o of all its pixels. It ranks objects higher if they are observed earlier in the fixation sequence. It does not concern the object sizes. For the methods *FixSeq-avgPmax* (Eq. 5) and *FixSeq-avgMmax* (Eq. 6), we consider weighting the final scores by performing addition or multiplication with the results from Eq. 3 and Eq. 4, respectively. The use of addition in *FixSeq-avgPmax* is a shorthand of averaging the effect of both *FixSeq-avg* and *FixSeq-max* values. *FixSeq-avgMmax* considers to weight *FixSeq-avg* by multiplying *FixSeq-max*.

In our user study, we use $T = 10$ as the number of top salient objects for ground-truth rank. Note that we only use top-5 during our prediction task. We then sort all objects in descending order of the saliency score, and each object is given a distinct rank.

Approach 2: This approach also considers temporal order. However, we only focus on the first T distinct objects and ignore repeated fixations on already visited objects. Moreover, we directly assign a score to the whole object if a fixation point resides in its segmentation. We term this method as *DistFixSeq*.

Specifically, we define a new sequence $\hat{f}_i^n$ by removing fixations that fall on objects that are already visited by earlier fixations in f_i^n. We then define *DistFixSeq*, for each object o in an image I as:

$$\textit{DistFixSeq}(o, I) = \frac{1}{N} \sum_{j}^{N} \sum_{i}^{T} h(\hat{f_i^j}), \quad \text{if } \hat{f_i^n} \in o \tag{7}$$

$$h(\hat{f_i^n}) = T - i, \tag{8}$$

where $T = 10$. Function h assigns higher scores to objects if they are observed earlier. Eq. 7 takes into account only the first T objects, then average it across all N observers. We then obtain the ranks of objects in the order of descending scores.

Approach 3: We use fixation maps in this approach as the source for saliency score. We directly take intensity values from the fixation map as pixel scores v_p. Similar to *Approach 1*, we expand this approach into four methods to generate the final scores for each object. Accordingly, we have *FixMap-avg* (average score), *FixMap-max* (maximum score), *FixMap-avgPmax* (average + maximum score) and *FixMap-avgMmax* (average $\times$ maximum score). These four methods compute the final scores of objects in the same way to their counterparts in *Approach 1* (as in Eq. 3-6). Again, we consider the first distinct T objects, and assign the saliency rank in the order of descending scores.

Saliency Map: In addition to assigning a distinct rank to each object, we also produce a saliency map for each image. Objects are given an initial saliency value according to their rank (*i.e.*, Rank 1 = 1, Rank 2 = 0.9, Rank 3 = 0.8, ..., Rank 10 = 0.1). These saliency values are further multiplied by 255 and the results are assigned to the corresponding object pixels to generate the final saliency map subsequently.

1.2. User Study

We conduct a user study with 11 participants (8 male, 3 female), in order to find out which of the 9 methods produces the best attention shift order that respects human judgement. We take the best method as our technique to generate the final ground-truth saliency rank in our dataset.

For each image, the participants were presented with the image and the nine corresponding saliency rank maps arranged in a grid. Fig. 1 shows a screenshot example of the annotation tool used in the user study. After a briefing session on how to use the annotation tool, every participant is told to observe the image first, then pick the maps that show objects with "order of decreasing attractiveness". Participants are not told how the maps are generated. Each participant was asked to annotate a set of 2500 images. These images are randomly sampled from our dataset. Participants

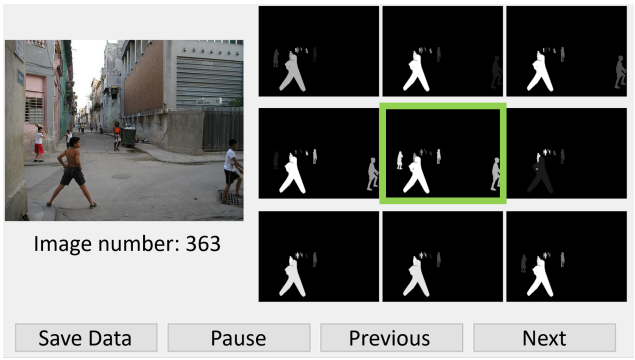

Figure 1: Screenshot of the annotation tool used by the participants during the user study. Participants are not told how the maps are generated. They are asked to pick the map that best respects their "order of attractiveness". The green box indicates the map picked by one of the participants.

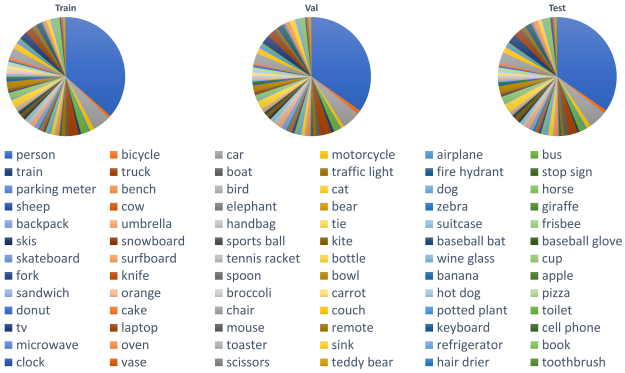

Figure 2: Distribution of ground-truth salient instances of all object categories in each data split of our dataset.

annotate them in 5 sessions (500 images each). Each annotation session lasts under an hour on average. After all the annotations, participants were rewarded with a £25 Amazon gift voucher for their time. The annotation result is shown in Fig. 3 in the main paper. It shows that human judgement of saliency rank (decreasing attractiveness) correlates very well to the maps generated by human attention shift.

1.3. Dataset Analysis

Our dataset is adapted from MS-COCO [5] and SALICON [3], and thus share similar characteristics (Sec. 5 in the main paper). All existing popular datasets (ECSSD, DUTS-OMRON, PASCAL-S, HKU-IS, DUTS) target binary salient object detection while ours focuses on **salient object ranking**. Our dataset contains more complex images and is larger in size. Note that all other datasets do not include individual object labels, making them ill-suited for our task.

We report that the average number of objects per image in our dataset is around 11 (maximum of 68). The "person" object category occurs the most throughout the dataset. However, many images contain crowd of people with small individual annotations, causing the total count to be 4-16 times greater than other categories. Similarly, "person" objects receive the most instances of ground-truth saliency which aligns to previous observations that humans usually attract attention [4]. This is shown in Fig. 2, which provides the distribution of ground-truth salient instances of each object category in our dataset. Fig. 3 shows the average rank of each object category based on instances, given ground-truth saliency. From the figure we can see that large objects (*e.g.*, "train", "airplane") with fewer instances per image and some animal categories (*e.g.*, "cat", "dog", "elephant") have a larger rank average score than a "person" object. We also find object categories relating to appliances (*e.g.*, "refrigerator", "microwave") have quite high scores, which mainly come from indoor scenes with no other object(s) of interest.

2. Implementation Details

Pre-processing and Training: In the main paper, we report our network results based on the training from a pre-processing strategy. Our pre-processing step outputs features from the backbone (Sec. 4.2, main paper) to save computation and training time. Consideration of this strategy also stemmed from the issue that our earlier network designs cannot fit into the memory of a single GPU card (NVIDIA GTX 1080 Ti 11GB) for training.

Our pre-processing strategy first generates object proposals for each image. We take the top M object proposals, whose probability scores are larger than 0.5. We chose $M = 30$, as it covers all objects appearing in an image for majority of our dataset. Next, we generate the corresponding object features and segmentation output for each object proposal. During the pre-processing step, we also generate the "$P5$" pyramid features from the backbone network, which we later use in the Selective Attention Module (Sec. 4.3, main paper). Finally, we train the rest of our network for saliency ranking using these pre-generated features as input.

Inference: In our current implementation, the object proposals come from the backbone network pre-trained for binary saliency prediction only. That is, it does not consider multiple saliency ranks. As a consequence, we do not use the confidence score of the object proposals (from binary classification) during our inference stage for rank prediction. Instead, we choose to use the softmax rank classification probabilities as our initial scores for distinct ranking (the last step in Sec. 4.5 in the main paper).

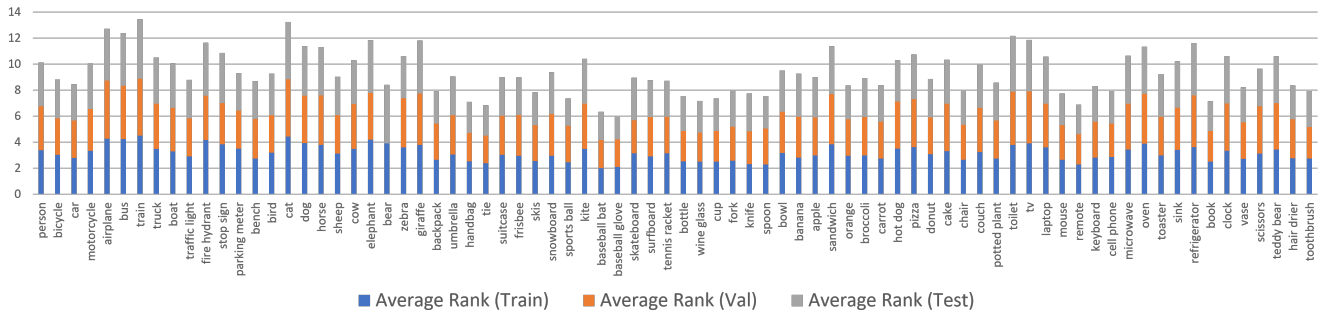

Figure 3: Average rank of each object category in the proposed dataset.

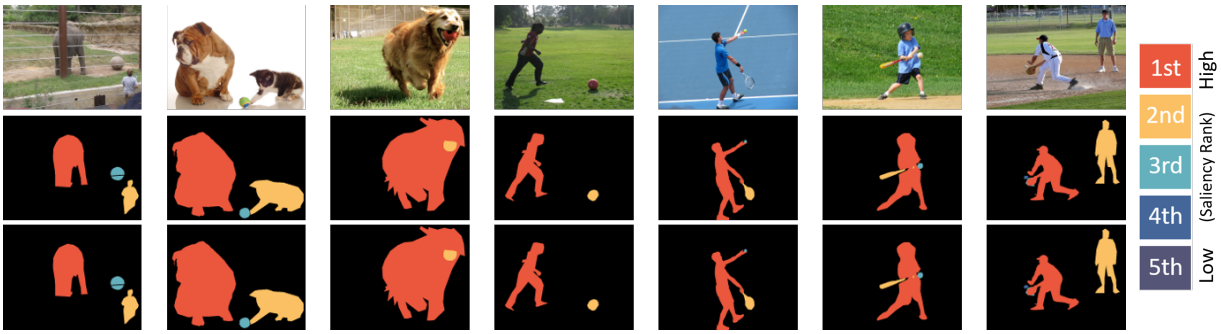

Figure 4: Example scenes containing "sports ball" object category. Images from our dataset (Top row), GT Ranks (Middle row), our network rank prediction (Last row).

3. Saliency Ranking on Different Contexts

Our study is the first deep network to model human attention shift. Our main focus is bottom-up and top-down inference that aligns closely to human visual processing. In the design, we have not fully explored scene context (we have only used spatial context and global image features), yet the results is promising. Exploring scene context will be an interesting future work.

Our network learns to reason the saliency rank of individual object features against the global features of an image scene. Such learning can also capture relationships between separate image features and corresponding saliency ranked objects. Fig. 4 showcases examples of different image scenes containing "sports ball". Our network is able to learn relationship between the object category and various image scenes, while correctly rank the object categories.

4. Comparison with State-of-the-Arts

As noted by the caption of Table 1 in the main paper, we directly evaluate RSDNet [1] on our dataset using their pre-trained weights, for two reasons: First, the idea and model of RSDNet are based on the agreement of twelve observers on binary saliency prediction. Our training dataset, however, is based on attention shift order of the most five salient objects. Their training strategy does not fit well to the nature of our dataset. Second, practically, when we try to train their model on their dataset, or to adapt and train their model on our dataset (using their available source code), both cases do not converge. We thus use their model with pre-trained weights to evaluate on our dataset.

Table 1: Quantitative comparison with S4Net for the task of salient instance detection on our dataset. Note that we do not include comparison with RSDNet, BASNet, CPD-R and SCRN since they are unable to perform this task.

Method	mAP^r@0.5 ↑	mAP^r@0.7 ↑
S4Net [2]	16.9 %	10.7 %
Ours	**57.4 %**	**48.3 %**

For S4Net [2], we modify the prediction layer in the salient object detection and segmentation heads from binary prediction (salient, background) to multiple saliency rank prediction (5 ranks, 1 background), and train on our dataset. We find that S4Net mostly predicts the same saliency rank (rank 1) during inference with standard classification. We apply the same inference method involved in our network (Sec. 4.5 in the main paper) to S4Net. This allows S4Net to produce distinct saliency rank predictions and enable fair comparison with our network.

Here we provide more qualitative comparisons between RSDNet [1], S4Net [2], BASNet [6], CPD-R [7], SCRN [8] and ours in Fig. 5 and 6.

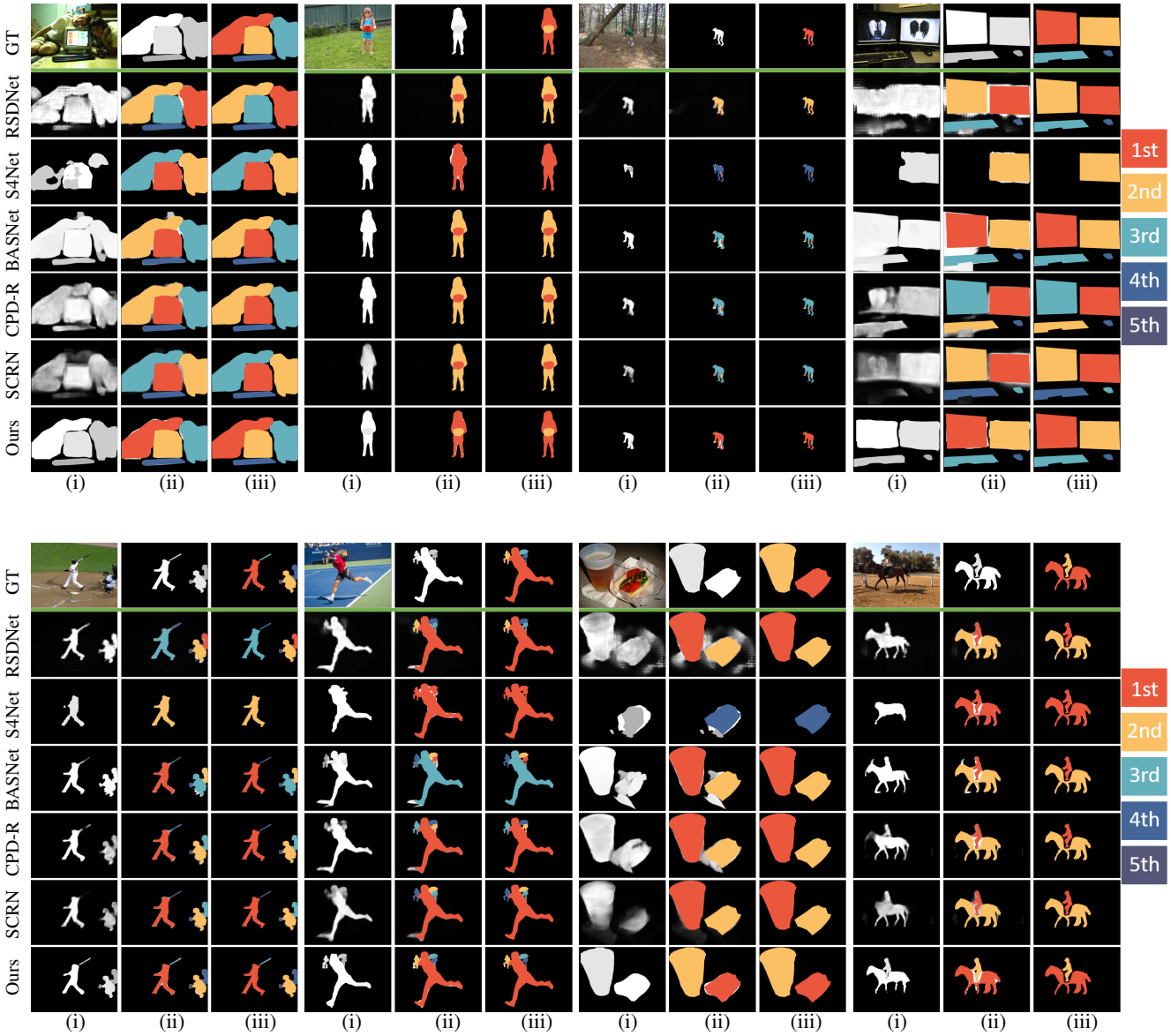

Figure 5: Further qualitative comparisons between RSDNet [1], S4Net [2], BASNet [6], CPD-R [7], SCRN [8] and our network. The top row (GT) in each of the 3 sub-figures shows 4 sets of examples. In each of the examples, we show respectively the image, the ground truth saliency map and the ground truth ranks. Each row of the 6 networks shows their respective results: (i) saliency prediction map, (ii) saliency prediction map with predicted rank of ground-truth object segments colourised on top, and (iii) corresponding map that contains only the predicted rank of ground-truth objects. Specifically, in each example, (i) provides a direct comparison of the predicted saliency maps (greyscales) against the ground-truth saliency map. The column (ii) visualises the false saliency and rank prediction from each methods. The column (iii) compares predicted saliency rank of ground-truth objects and their corresponding ground-truth rank. We use (iii) ground-truth object segmentation to obtain their predicted saliency ranks for numerical evaluation.

5. Further Comparison with S4Net

Like S4Net [2], our network is able to generate individual segmentation for each salient object instance. We further compare our network to S4Net on the task of salient instance detection. We do not include comparison with RSDNet [1], BASNet [6], CPD-R [7] and SCRN [8] as they are unable to produce output of salient object instances. We use the mean Average Precision (mAP^r, $r = 0.5/0.7$) to measure the performance similarly as in [2]. Table 1 reports the results between S4Net and our network for salient instance detection on our dataset. The table shows that our network outperforms S4Net by a large margin. The results reveal that S4Net is not able to handle the primary task of salient object ranking, which is the focus of this paper. S4Net predicts very few salient objects when compared to

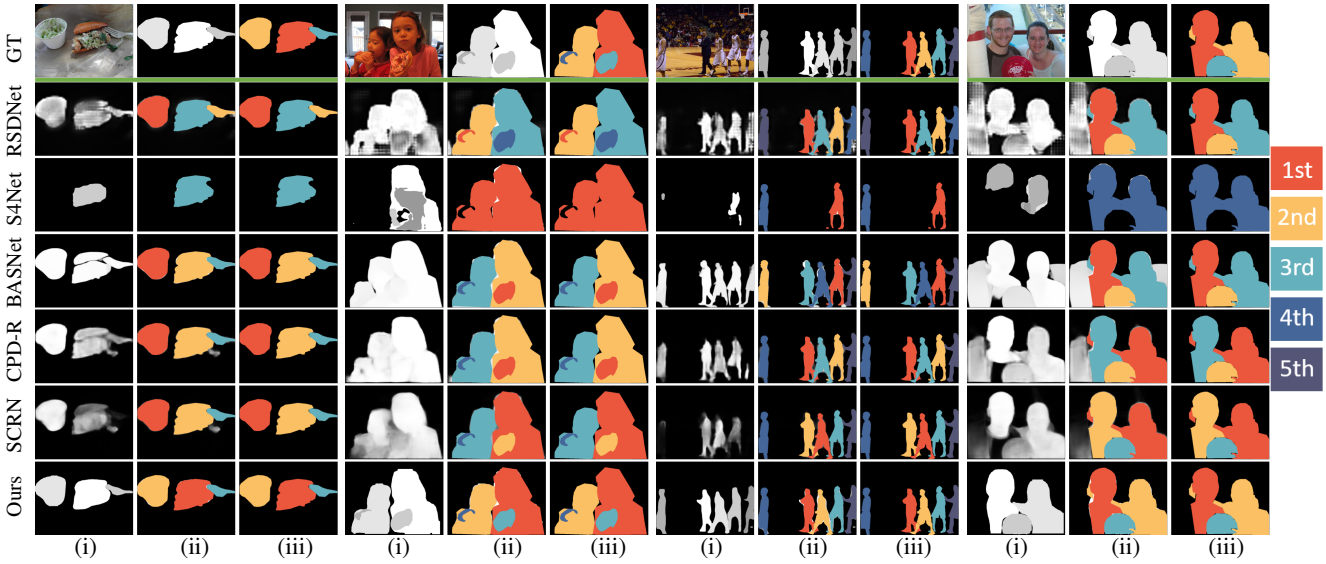

Figure 6: Further qualitative comparisons between RSDNet [1], S4Net [2], BASNet [6], CPD-R [7], SCRN [8] and ours (Fig. 5 *cont.*).

our network (see Fig. 5 and 6) and misses the prediction of saliency towards corresponding ground-truth objects in over one third of the test set (indicated by #Images used in Table 1 in the main paper).

References

[1] Md Amirul Islam, Mahmoud Kalash, and Neil D. B. Bruce. Revisiting salient object detection: Simultaneous detection, ranking, and subitizing of multiple salient objects. In *CVPR*, pages 7142–7150, 2018.

[2] Ruochen Fan, Ming-Ming Cheng, Qibin Hou, Tai-Jiang Mu, Jingdong Wang, and Shi-Min Hu. S4net: Single stage salient-instance segmentation. In *CVPR*, pages 6103–6112, 2019.

[3] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *CVPR*, pages 1072–1080, 2015.

[4] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *ICCV*, pages 2106–2113, 2009.

[5] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014.

[6] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *CVPR*, pages 7479–7489, 2019.

[7] Zhe Wu, Li Su, and Qingming Huang. Cascaded partial decoder for fast and accurate salient object detection. In *CVPR*, pages 3907–3916, 2019.

[8] Zhe Wu, Li Su, and Qingming Huang. Stacked cross refinement network for edge-aware salient object detection. In *ICCV*, pages 7264–7273, 2019.