Realistic Saliency Guided Image Enhancement

Supplementary Material

S. Mahdi H. Miangoleh[1] Zoya Bylinskii[2] Eric Kee[2] Eli Shechtman[2] Yağız Aksoy[1]

[1] Simon Fraser University [2] Adobe Research

In this supplementary document, we present (i) formal definition of the edit operators used in our method, (ii) architectures, training details, and dataset generation process for our image editing and realism networks, (iii) an extended evaluation of our methods generalization to multi-mask inputs in Table 1, (iv) details on our user study photographers and complementary user study analysis in Table 2 and Figure 2, (v) expanded figures and details of our edit optimality experiment in Figure 4, (vi) numerical evaluations using no-reference image quality metrics in Table 3, (vii) additional results and comparisons to the state-of-the-art in Figures 5, 6, 7, 9, 8, 10, and 11.

A. Editing Operations

To generate training data for our Realism network as described in Section 3 of the main paper we employ commonly used image editing operations– exposure, saturation, color curve, and white balancing (global color). Given input image I the formal definition of each of the image editing operations is as following:

Exposure It is implemented by multiplying all pixel values by a single scalar.

$$I' = p_{exp} \cdot I \tag{1}$$

Color Curve We use the monotonic piecewise-linear curve representation proposed in [3] for color curve representation.

$$f(x, \vec{p}\,) = \frac{1}{\sum_{i=0}^{L} p_i} \sum_{i=0}^{L-1} clip(L{\cdot}x - i, 0, 1) * p_i \tag{2}$$

We set $L = 8$ and use three sets of curves parameters ($\overrightarrow{p_{cc}}$) for each R, G, and B channels.

$$I'_r = f(I_r, \overrightarrow{p^r_{cc}}), \quad I'_g = f(I_g, \overrightarrow{p^g_{cc}}), \quad I'_b = f(I_b, \overrightarrow{p^b_{cc}}) \tag{3}$$

Saturation We multiply a scalar value p_{sat} to the S channel in HSV color space and convert it back to RGB.

White Balance (global color) We use a channel-wise multiplier $\overrightarrow{p_{wb}}$.

$$I'_r = I_r * p^r_{wb}, \quad I'_g = I_g * p^g_{wb}, \quad I'_b = I_b * p^b_{wb} \tag{4}$$

B. Training Details

B.1. Image Editing Network Architecture

The parameter estimation network regresses multiple sets of parameters, one for each edit operation, given an input RGB image and a concatenated region mask. The permutation order in which the edits will be applied is also provided to the network as a vector. Figure 1 illustrates our network architecture. We use an EfficientNet-lite3 [11] backbone to encode the image and mask (The first convolutional layer of EfficientNet is modified to accept four input channels) into a vector that is concatenated with a corresponding vector that describes the input permutation. The image encoding vector is average pooled from the EfficientNet feature tensor that precedes the EfficientNet convolutional head yeilding a 384-D feature vector. The input permutation vector is encoded by a linear layer that takes as input a 4-D permutation vector encoded with integer indices in $[0, 3]$, and which outputs a 32-D vector of features.

This 416-D, concatenated image and permutation feature vector is next projected to 128 dimensions by a two layer MLP with 128 hidden dimensions and a leaky ReLU non linearity. The resulting 128-D vector is shared among the parameter estimation heads. Each head is a linear layer with a sigmoidal activation that bounds the parameter ranges. Specifically, each sigmoidal output is transformed into a parameter value via a manually designed affine transform. The transformed ranges were chosen to encompass a large but plausible range of values for each parameter, with a single affine transform for each parameter estimation head. The target ranges for the affine transformed parameters are $[0.1, 1]$ white balance, $[0.5, 2]$ saturation, $[0, 2]$ color curve values, and $[0.5, 2]$ exposure. In addition, in order to keep the overall luminance of the image unchanged during applying the while balance operation, we keep the green chan-

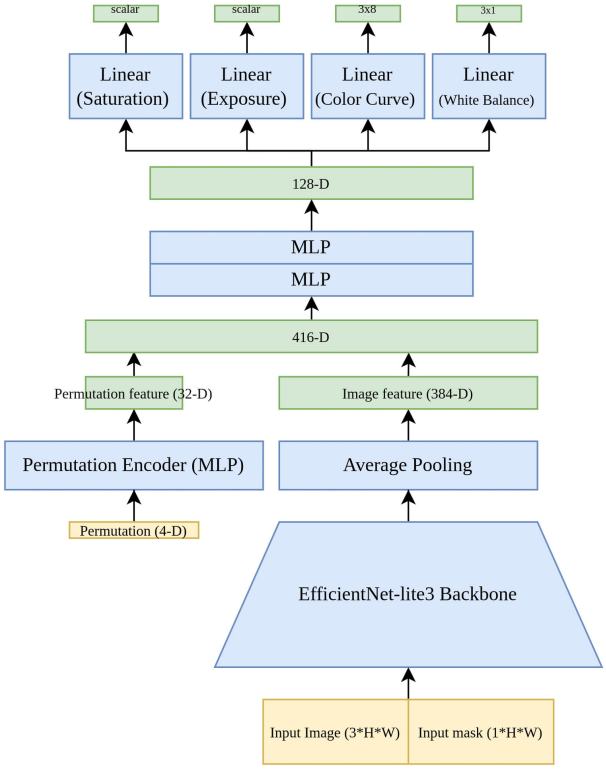


Figure 1. Image editing network architecture

nel untouched by normalizing the parameters ($\overrightarrow{p_{wb}}$) by the green channel value.

B.2. Dataset

We use MS-COCO [5] to generate input image and mask pairs as our training data. MS-COCO provides a segmentation mask for each of the objects present in the scene.

We start with 118K segmented MS-COCO images, discard masks that are either too small ($< 1\%$ of the image area) or too large ($> 40\%$, same thresholds as in [8]), and then omit images with less than 3 masks. Since the saliency of one object is relative to the presence of other objects in the scene [8], we make sure that for each selected object, there will be at least one more object in the scene with a lower, and one with a higher, saliency. To achieve that using SalNet [4] we generate the saliency heatmap for each image and sort the segments by their mean saliency value. We discard the first and last segments (corresponding to the image regions with highest and lowest saliency in the image) and use the rest of the segments as masks during training.

This resulted in 108,587 image-mask training pairs. Since the edited image is generated by Alpha-blending the edited image region with the input image using the mask, the realism of the result is affected by mask boundary imperfections. We used an out-of-box Matting [2] method to generate alpha mattes from input binary masks to avoid artifacts around mask boundaries during training.

Table 1. Test performance of multi-region edits compared to GazeShift [8]. Edits to multiple regions can be performed simultaneously (union) or iteratively (iter). GazeShift furthermore incorporates a background edit, which interferes with iterative processing, and is therefore removed (iter-bg). Iterative processing improves saliency for both methods and both tasks, but degrades realism for GazeShift. Realism is however relatively unaffected by our method, as background edits are not required to modulate saliency.

	Attenuation		Amplification	
	$S\,(\%)\downarrow$	$(\Delta R)\uparrow$	$S\,(\%)\uparrow$	$\Delta R\uparrow$
GazeShift (union)	3.6	-0.21	21.0	-0.01
GazeShift (iter)	3.3	-0.33	41.0	-0.15
GazeShift (iter-nobg)	-3.3	-0.30	37.0	-0.15
Ours (union)	-16.0	-0.10	32.0	0.00
Ours (iter)	-17.0	-0.06	38.0	0.01

B.3. Procedure

Fist we train our Realism network and freeze its weights during training of the Parameter estimation network. We use the dataset of input image and mask pairs described in Section B.2 to generate extreme real and fake edited images as described in Section 3 of the main paper. Using the ADAM optimizer (learning rate of 1e-5) we train the network for 50 epochs with a batch size of 64.

We train two two separate networks to estimate parameters for attenuation and amplification tasks. We train the networks using the ADAM optimizer (learning rate of $1e-5$) for 50 epochs with a batch size of 16. We clip $\mathcal{L}_{\text{sal}}$ (main paper, Equation 4) for values bigger than 10 during the training to prevent training instability. A random ordering of the 4 edit operations (out of 24 possible permutations) is sampled for each training batch and provided to the networks as input.

C. Experiments and Results

C.1. Generalization to multiple image regions

For each validation image on MS-COCO [5] we selected multiple objects as regions of interest (within the same size constraints as before). Then we evaluated the performance of our method on the union of the masks (all at once) compared to running it on the masks in an iterative fashion. For this evaluation we use the value ΔR (main paper, Equation 2) as an automated measure of realism and the relative saliency change S (main paper, Equation 3) as a measure of effectiveness. Table 1 shows that the iterative approach generates a result that is more effective and realistic than when running it on all the masks at once.

Table 2. Eight professional photographers were recruited from UpWork for our user study. Table provides details about their backgrounds.

	UpWork Position Title	Experience	Gender	Country
1	Photo Retoucher and Editor	5+ years	Male	Serbia
2	Photoshop Expert	15+ years	Male	Ukraine
3	Photo Retoucher and Editor	5+ years	Male	Greece
4	Digital Image Editor, Photographer	7+ years	Male	Armenia
5	Artist and Photoshop Expert	20+ years	Male	India
6	Photo Retoucher and Editor	12+ years	Male	Philippines
7	Photo Editor, Visual Storyteller	13+ years	Male	Armenia
8	Photo Retoucher and Editor	6+ years	Female	Greece

We tried the same approach on the competing GazeShift [8] model. As mentioned in the main paper, Gazeshift edits the whole image by estimating two sets of edit parameters, one for the region of interest (foreground) and one for the background. This makes iterative editing impractical. For a more practical comparison, we omit background edits when running GazeShift. Table 1 shows that [8] GazeShift performance suffers on an iterative saliency enhancement task, but our results remain robust.

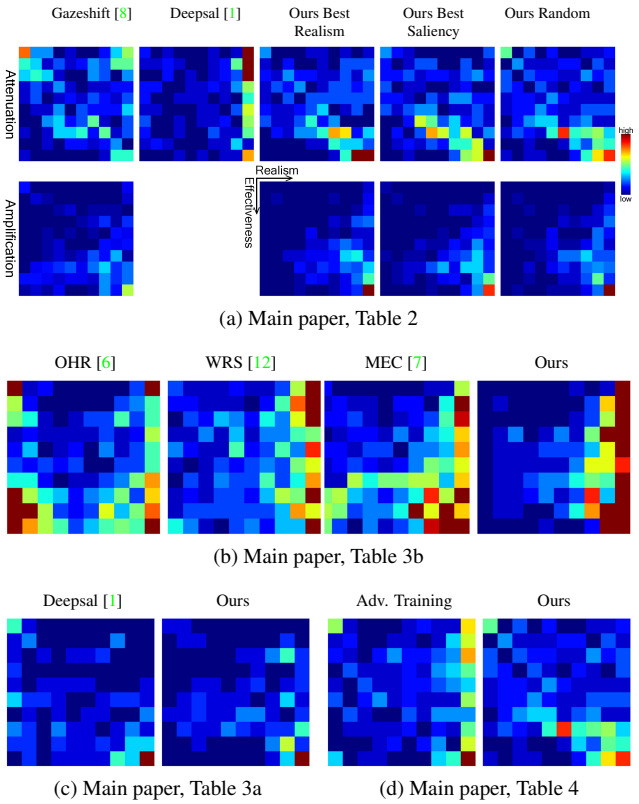
C.2. User Study

Professional photographers The photographers in our study had 5-20+ years of experience with photo retouching, studio, and freelance photography (portrait, event, stock, product, magazine). They live and work in different countries. More details on each of the photographers can be found in Table 2.

Heatmap visualization During our user study we asked professional photographers to score the images with respect to realism and effectiveness. One important point when evaluating the user study results is that both metrics should be considered on the same time. A method that tend to apply very subtle edits will achieve a high realism score. A preferred method is the one that gains high realism scores while being effective. To complement the mean and variance scores provided in the main paper, Table 2, 3, and 4 we provide a 2-D heatmap visualizing the realism vs effectiveness in Figure 2. The activation corresponds to the number of samples in the user study with the corresponding effectiveness and realism values. Heatmaps show that our method always achieves a high realism and effectiveness together (high activation on bottom right) while the state of the art methods are either not realistic enough (high activation toward the left side of the heatmap) or are not effectively changing the saliency(high activation on top right part of the heatmap).

C.3. Effect of Different Order of Edits

Figure 3 illustrates the results generated by our method for the same image, but given different orderings of parameters. Our method is able generate robust, while subtly different results, given different parameter orders. Our user study results (main paper, Table 2) also provides comparison between different strategies of selecting a permutation. Our Photographer ratings also indicates a randomly selected permutation of parameters achieves comparable performance to a permutation of parameters that is more carefully chosen. For this reason, we plot the results of a randomly selected permutation of parameters for all the comparison figures in this document and in the main paper.



(a) Main paper, Table 2

(b) Main paper, Table 3b

(c) Main paper, Table 3a

(d) Main paper, Table 4

Figure 2. User study results visualized as heatmaps as complement of the mean and variance values reported in the main paper Table 2, 3, and 4. The activation values corresponds to the number of samples in the user study with the specific realism(y-axis) and effectiveness(x-axis) values (from 1 to 10). Our method always achieves a high realism and effectiveness at the same time (high activation on bottom right) while the state of the art methods are either not realistic enough (high activation toward the left) or are not effectively changing the saliency (high activation on top right).

C.4. No-Reference Image Quality Metrics

We utilized no-reference image quality assessment algorithms of Natural Image Quality Evaluator (NIQE) [10] and Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) [9], to numerically evaluate image naturalness and realism of ours and comparing methods. NIQE mea-

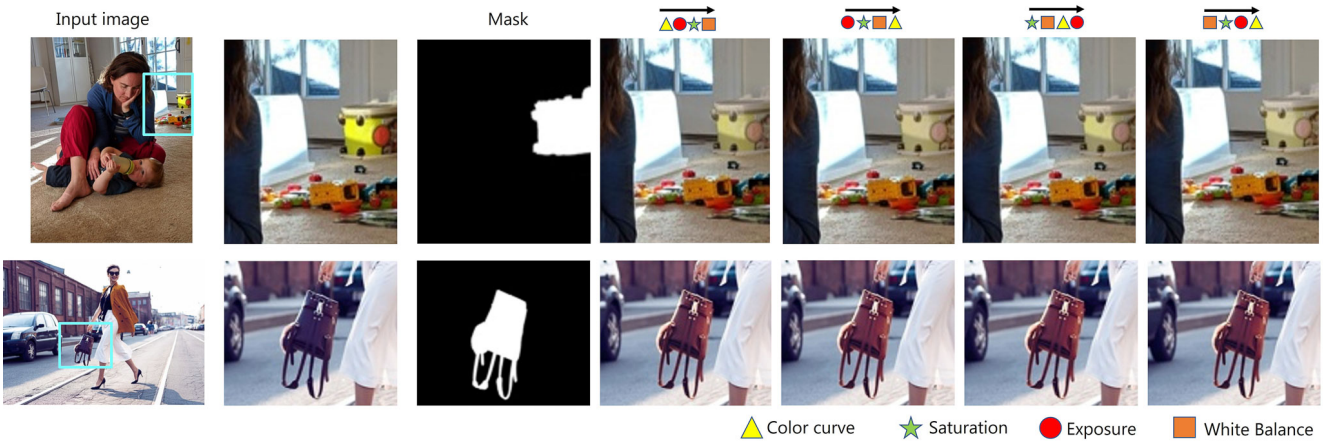


Figure 3. Editing results using different orderings of editing operations as input to our Parameter network yields the same editing results. The sequence of symbols above each image indicates the ordering, read left to right. The operations are noted below: color curves, saturation, exposure, and white balance.

sures the deviations from statistical regularities observed in natural images without requiring any exposure to distorted images or training examples with corresponding human opinion scores. BRISQUE is a distortion-generic algorithm that employs natural scene statistics of locally normalized luminance coefficients to quantify possible losses of 'naturalness' in an image due to the presence of distortions. Both metrics generate a score that reflects the quality and naturalness of an image. Table 3 provides the evaluations in parallel to Table 2 and 3b of the main paper using NIQE and BRISQUE metrics to quantify the naturalness of our results.

C.5. Edit Optimality

Figure 4 provides extra examples to show the optimality of the estimated parameters by our method (same as main paper, Figure 9a). To test the optimality of our estimated edits we keep the estimated parameters for *white balance* and *color curve* operations fixed and change the *saturation* and *exposure* by ± 0.1 in each step and visualize the realism score of the resulting image as a heatmap (center of the heatmap corresponds to our estimated parameters and in each direction we add or subtract the estimated value by 0.1). Heatmaps visualized in Figure 4 show that our estimated parameters are the optimal edits in regard with the realism scores. Images also illustrate the type of edits our realism estimation network considers realistic or unrealistic.

C.6. Extra Qualitative Results

We present results of our method alongside the estimated saliency heatmaps before and after applying the edits in Figure 5. Figure shows that the estimated edits change the attention in the desired direction effectively.

Figures 6, 7, 8, 9, 10 and 11 provide our results in comparison with state-of-the-art methods– Gazeshift [8] and Deepsal [1] as an extension to the comparisons included in the main paper.

Figure 6 and 7 (images from Deepsal supplementary webpage) show that Deepsal frequently applies intense color changes without considering the semantics (e.g., gray camouflaged traffics cones in the first row and decorations in the last row of Fig. 6). This can cause unrealistic edits. Our method effectively suppresses the distractor while keeping high levels photo-realism. Figures 8 and 9 indicate that Deepsal fails to apply effective edits on Adobe Stock dataset images.

Comparing our results and Gazeshift for attenuating the saliency of the masked region in Figures 6, 7, 8 and 9 indicates that Gazeshift is less effective compared to our method while we also maintain a higher level of realism (e.g. green sheep in the sixth row of Fig. 9). Since Gazeshift specializes in increasing the saliency results in Figure 10 and Figure 11 show they succeed to amplify the saliency of the masked region. However, our method applies more effective edits while maintaining the realism. We believe the explicit realism loss helps our method to apply more intense edits when the object semantics allows, while the adversarial loss used in Gazeshift limits the span of the edits their method applies. This limiting effect of adversarial loss was also studied in our ablation study (main paper, Section 5.3).

Table 3. No-Reference Image quality metrics NIQE [10] and BRISQUE [9] parallel to Table 2 and 3b of the main paper. All pairwise comparisons are statistically significant ($p < 0.01$ according to Welch T-tests).

	Saliency Attenuation		Saliency Amplification	
Method	NIQE ↓	BRISQUE ↓	NIQE ↓	BRISQUE ↓
GazeShift [8]	2.76(0.88)	23.8(10.4)	2.78(0.80)	24.2(10.0)
DeepSal [1]	2.75(0.75)	24.8(10.1)	-	-
Ours - Random	2.41(0.84)	25.4(9.9)	2.41(0.84)	25.3(9.8)

(a) Adobe Stock dataset [8].

	Saliency Amplification	
Method	NIQE ↓	BRISQUE ↓
MEC [7]	3.76(0.96)	30.1(8.6)
WRS [12]	3.78(0.95)	31.0(8.8)
OHR [6]	3.78(0.96)	30.3(8.7)
Ours - Random	3.75(0.98)	30.0(8.8)

(b) Mechrez dataset [7].

References

[1] Kfir Aberman, Junfeng He, Yossi Gandelsman, Inbar Mosseri, David E. Jacobs, Kai Kohlhoff, Yael Pritch, and Michael Rubinstein. Deep saliency prior for reducing visual distraction. In *Proc. CVPR*, 2021. 3, 4, 5, 8, 9, 10

[2] Marco Forte and François Pitié. F, b, alpha matting. `arXiv:2003.07711 [cs.CV]`, 2020. 2

[3] Yuanming Hu, Hao He, Chenxi Xu, Baoyuan Wang, and Stephen Lin. Exposure: A white-box photo post-processing framework. *ACM Trans. Graph.*, 37(2):26, 2018. 1

[4] Sen Jia and Neil D.B. Bruce. EML-NET: An Expandable Multi-layer NETwork for saliency prediction. *Image Vis. Comput.*, 95:103887, 2020. 2, 7

[5] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proc. ECCV*, 2014. 2

[6] Victor A. Mateescu and Ivan V. Bajić. Attention retargeting by color manipulation in images. In *Proc. PIPV*, 2014. 3, 5

[7] Roey Mechrez, Eli Shechtman, and Lihi Zelnik-Manor. Saliency driven image manipulation. *Mach. Vis. Appl.*, 30(2):189–202, 2019. 3, 5

[8] Youssef Alami Mejjati, Celso F. Gomez, Kwang In Kim, Eli Shechtman, and Zoya Bylinskii. Look here! a parametric learning based approach to redirect visual attention. In *Proc. ECCV*, 2020. 2, 3, 4, 5, 8, 9, 10, 11, 12

[9] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.*, 21(12):4695–4708, 2012. 3, 5

[10] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Process. Lett.*, 20(3):209–212, 2012. 3, 5

[11] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proc. ICML*, 2019. 1

[12] Lai-Kuan Wong and Kok-Lim Low. Saliency retargeting: An approach to enhance image aesthetics. In *Proc. WACV*, 2011. 3, 5

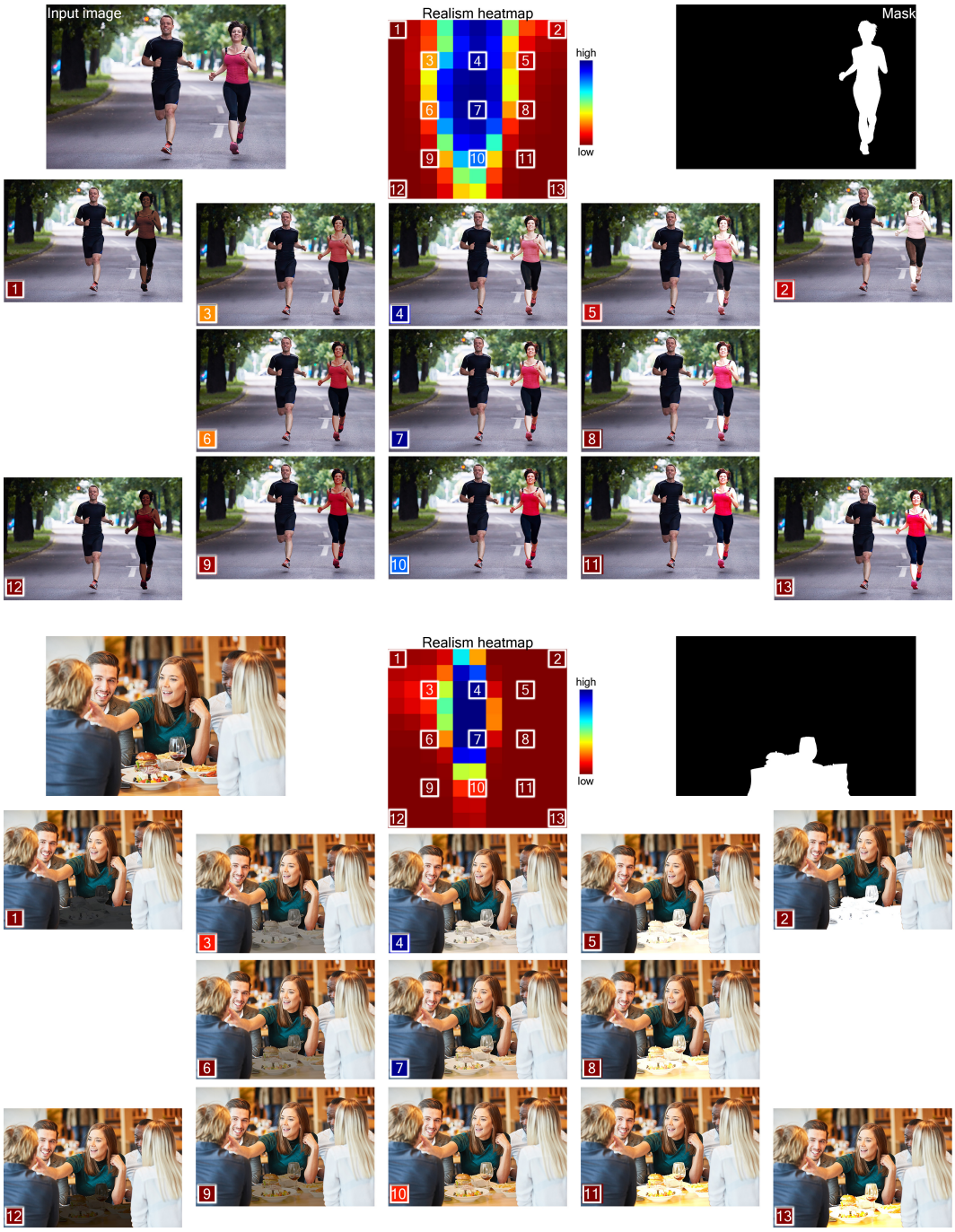


Figure 4. Heatmaps visualize the realism score achieved when we change the estimated saturation (x-axis) and exposure (y-axis). Our estimated values (center of the heatmap) achieve the optimal realism while changing the parameters in any direction reduces the realism. Sample edited images and their corresponding location in the heatmap are also visualized.

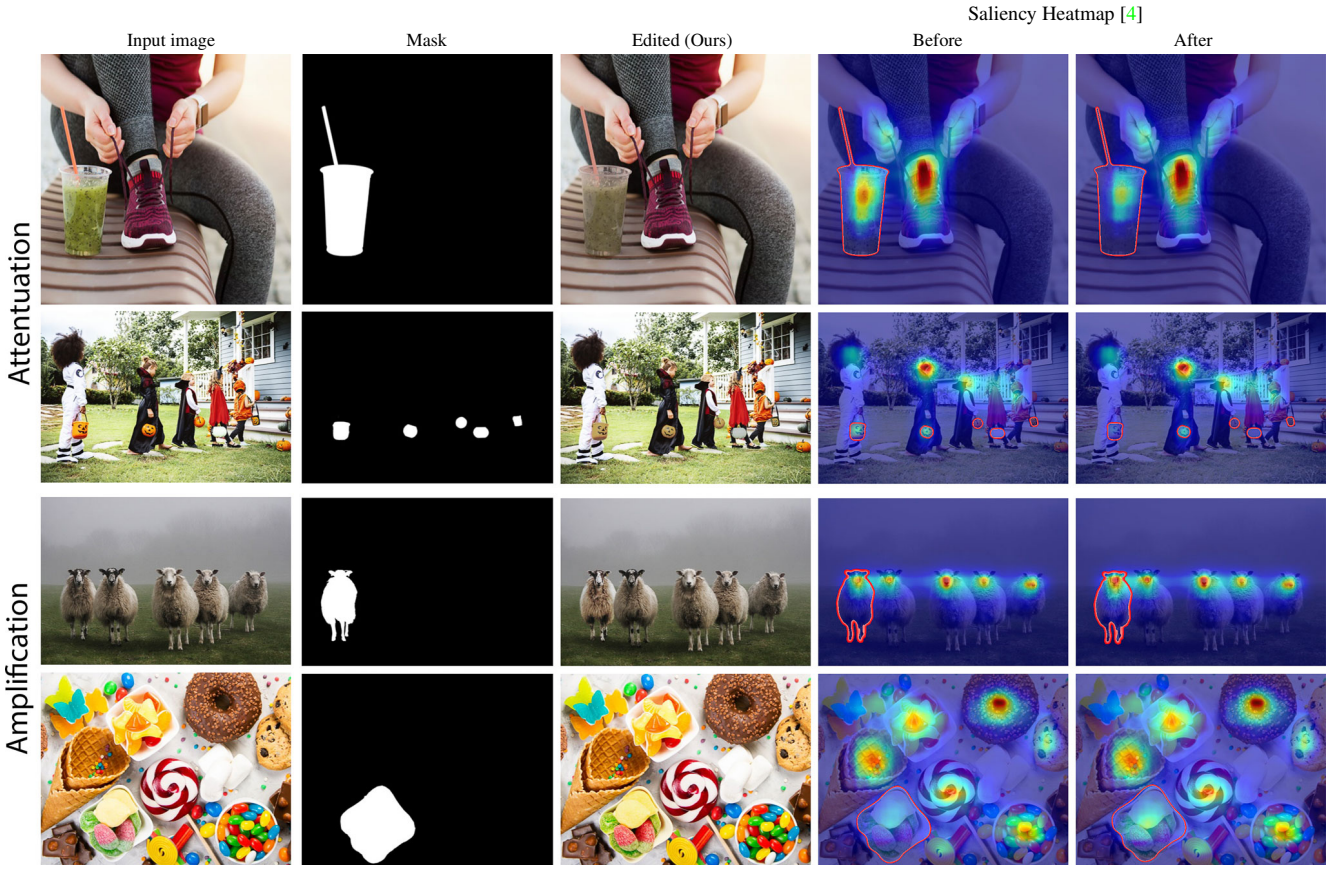


Figure 5. Predicted edits for the saliency attenuation and amplification tasks consistently improve the saliency maps of [4] in the desired direction. Saliency maps of the input and edited images are also shown; edited regions are emphasized with the red borders. The saliency maps are consistently attenuated (top two images) and amplified (bottom two images) with the predicted edits.

Figure 6. Saliency attenuation compared to Deepsal [1] and Gazeshift [8] on the images provided by Deepsal authors on their project webpage.

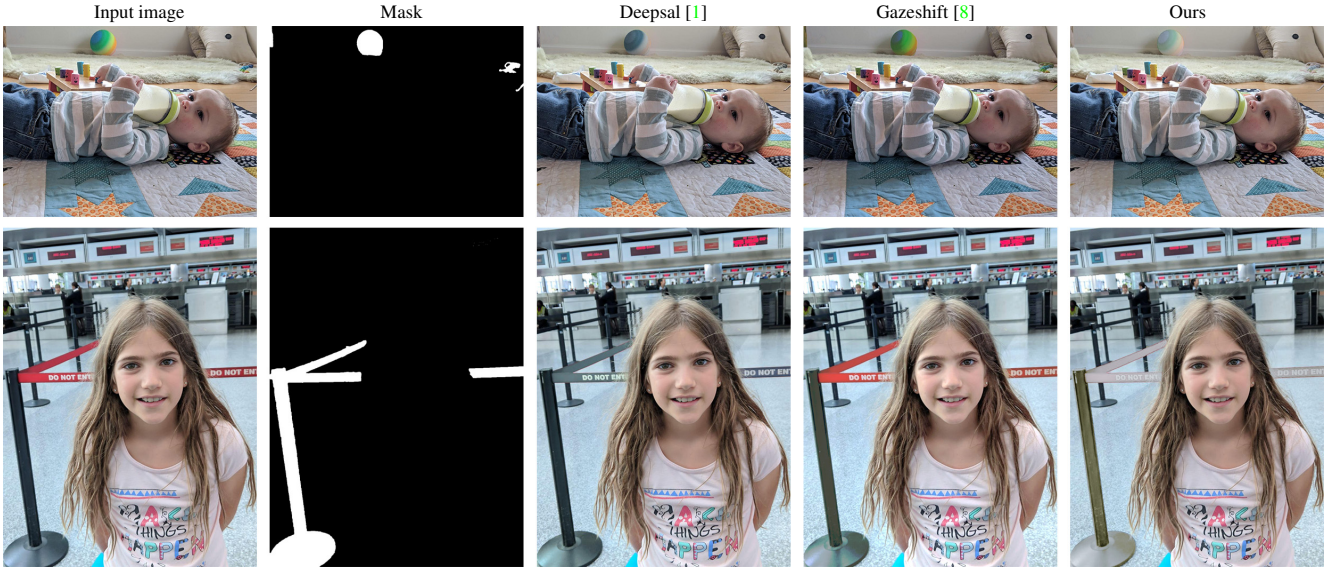


Figure 7. Figure 6 continued.

Figure 8. Saliency attenuation compared to Gazeshift [8] and Deepsal [1] on Adobe Stock images from Gazeshift.

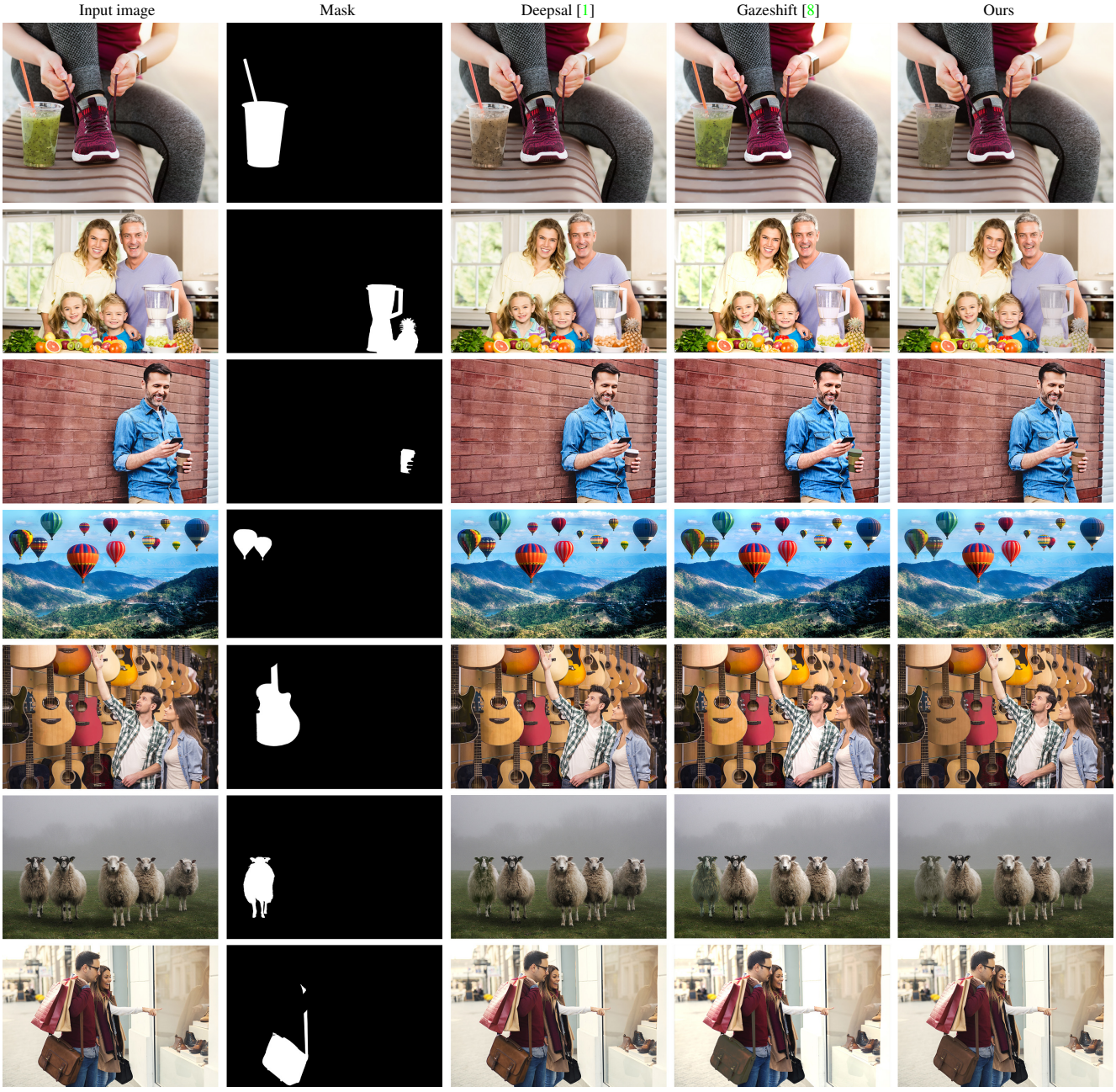


Figure 9. Figure 8 continued.

Figure 10. Saliency amplification compared to Gazeshift [8] on their Adobe Stock images.

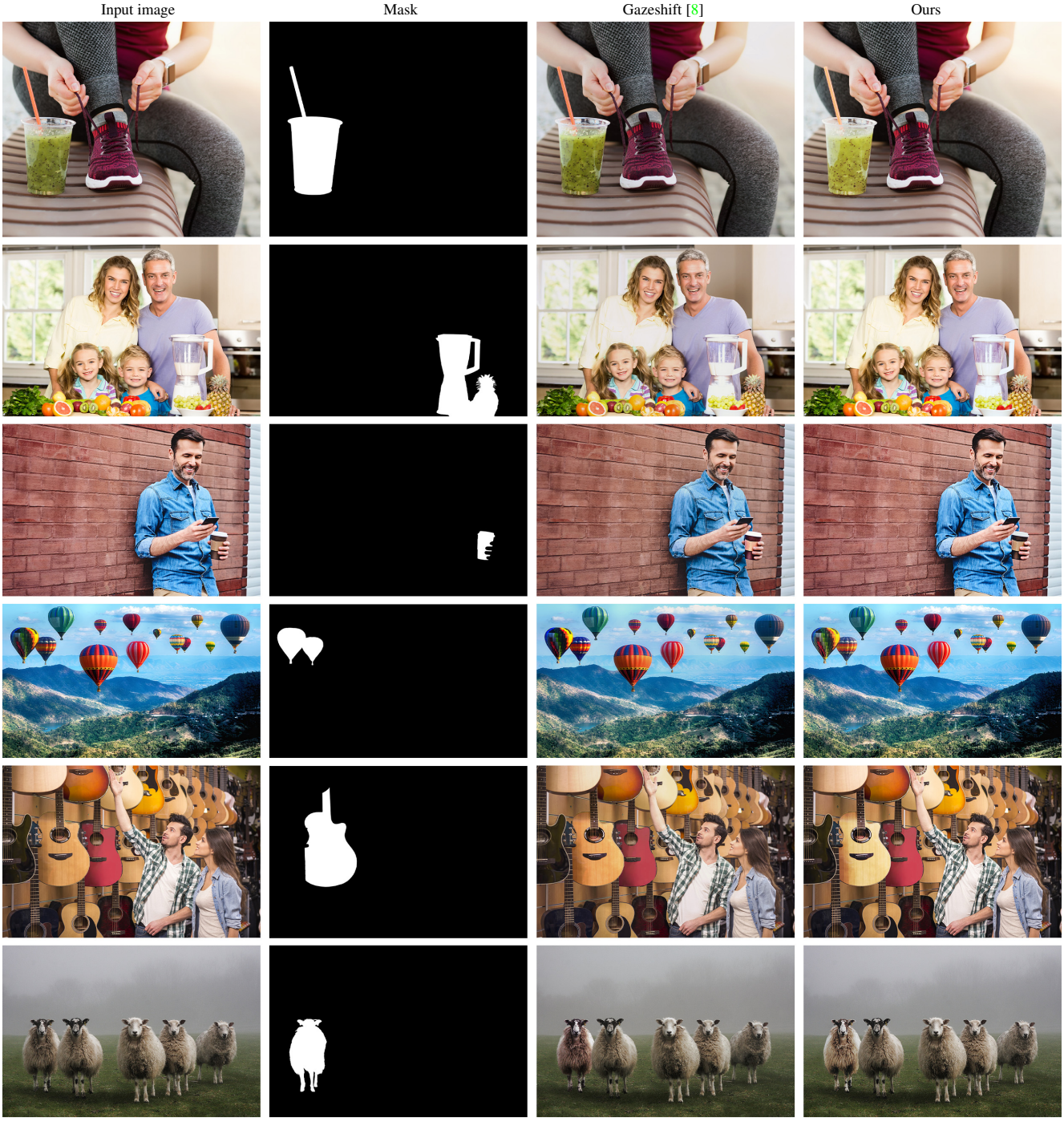


Figure 11. Figure 10 continued.