Find the project web page here:
http://yaksoy.github.io/realisticEditing/

Realistic Saliency Guided Image Enhancement

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

[1] Simon Fraser University [2] Adobe Research

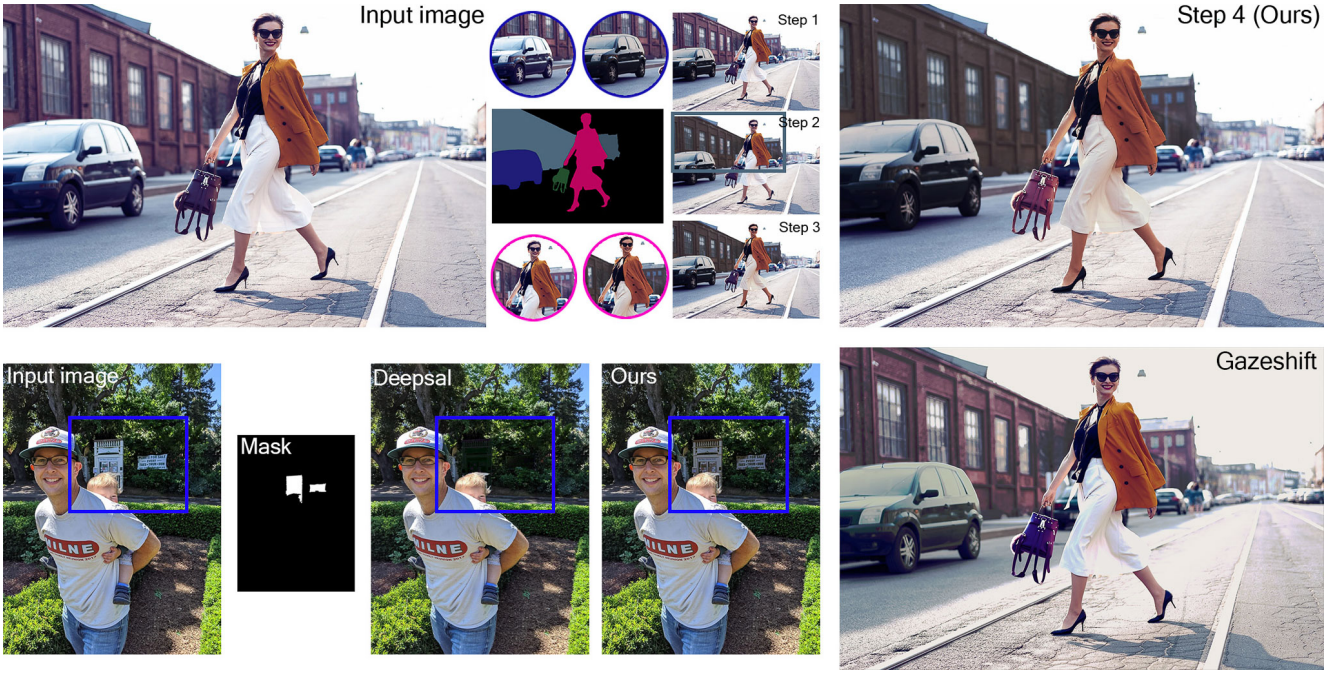


Figure 1. (top) We develop a saliency-based image enhancement method that can be applied to multiple regions in the image to de-emphasize objects (steps 1, 2) or enhance subjects (steps 3, 4). (bottom) Our novel realism loss allows us to apply realistic edits to a wide variety of objects while state-of-the-art methods [1, 17] may generate less realistic editing results.

Abstract

Common editing operations performed by professional photographers include the cleanup operations: de-emphasizing distracting elements and enhancing subjects. These edits are challenging, requiring a delicate balance between manipulating the viewer's attention while maintaining photo realism. While recent approaches can boast successful examples of attention attenuation or amplification, most of them also suffer from frequent unrealistic edits. We propose a realism loss for saliency-guided image enhancement to maintain high realism across varying image types, while attenuating distractors and amplifying objects of interest. Evaluations with professional photographers confirm that we achieve the dual objective of realism and effectiveness, and outperform the recent approaches on their own datasets, while requiring a smaller memory footprint and runtime. We thus offer a viable solution for automating image enhancement and photo cleanup operations.

1. Introduction

In everyday photography, the composition of a photo typically encompasses subjects on which the photographer intends to focus our attention, rather than other distracting things. When distracting things cannot be avoided, photographers routinely edit their photos to de-emphasize them. Conversely, when the subjects are not sufficiently visible, photographers routinely emphasize them. Among the most common emphasis and de-emphasis operations performed by professionals are the elementary ones: changing the saturation, exposure, or the color of each element. Although conceptually simple, these operations are challenging to apply because they must delicately balance the effects on the viewer attention with photo realism.

To automate this editing process, recent works use saliency models as a guide [1, 2, 4, 8, 16, 17]. These saliency models [3, 7, 10, 14, 19] aim to predict the regions in the

image that catch the viewer's attention, and saliency-guided image editing methods are optimized to increase or decrease the predicted saliency of a selected region. Optimizing solely based on the predicted saliency, however, often results in unrealistic edits, as illustrated in Fig. 1. This issue results from the instability of saliency models under the image editing operations, as saliency models are trained on unedited images. Unrealistic edits can have low predicted saliency even when they are highly noticeable to human observers, or vice versa. This was also noted by Aberman et al. [1], and is illustrated in Fig. 2.

Previous methods tried to enforce realism using adversarial setups [2, 4, 8, 17], GAN priors [1, 8], or cycle consistency [2] but with limited success (Fig. 1). Finding the exact point when an image edit stops looking realistic is challenging. Rather than focusing on the entire image, in this work, we propose a method for measuring the realism of a local edit. To train our network, we generate realistic image edits by subtle perturbations to exposure, saturation, color or white balance, as well as very unrealistic edits by applying extreme adjustments. Although our network is trained with only positive and negative examples at the extremes, we successfully learn a continuous measure of realism for a variety of editing operations as shown in Fig. 3.

We apply our realism metric to saliency-guided image editing by training the system to optimize the saliency of a selected region while being penalized for deviations from realism. We show that a combined loss allows us to enhance or suppress a selected region successfully while maintaining high realism. Our method can be also be applied to multiple regions in a photograph as shown in Fig. 1.

Evaluations with professional photographers and photo editors confirm our claim that we maintain high realism and succeed at redirecting attention in the edited photo. Further, our results are robust to different types of images including human faces, and are stable across different permutations of edit parameters. Taken together with our model size of 26Mb and run-time of 8ms, these results demonstrate that we have a more viable solution for broader use than the approaches that are available for these tasks to date.

2. Related Work

Various image enhancement methods have been introduced in the literature to amplify a region of interest or de-emphasise distracting regions, improve image aesthetics, and redirect the viewer's attention. This task has been referred to as *attention retargeting* [15] or *re-attentionizing* [18] as well. Earlier methods [5, 15, 20, 22, 23] incorporated prior knowledge of saliency cues (saturation, sharpness, color, gamut, etc.) to guide the editing process to achieve the desired change in saliency. But, relying solely on saliency cues both limits the diversity of generated edits, and creates unrealistic edits due to the lack of semantic

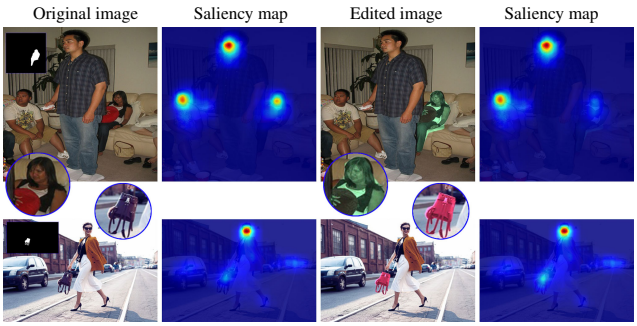


Figure 2. Predicted saliency maps [7] for the original images and edited versions, with extreme edits applied. Note that saliency models are typically trained with realistic images. This makes them susceptible to inaccurate predictions for unrealistic inputs, as the green woman in the top row estimated to have low saliency.

constraints. As our experiments show, OHR [15] tends to generate unrealistic color changes that are semantically incorrect, and WRS [23] is limited to contrast and saturation adjustments with limited effectiveness.

Recent works leverage saliency estimation networks [3, 7, 10, 14, 19] to optimize for a desired saliency map instead of relying on prior saliency cues. Saliency models are trained to output a heatmap that represents where human gaze would be concentrated in an image. These models are not trained to respond to the realism of the input image. Hence they might predict an inconsistent decrease or increase in the saliency of a region when unrealistic or semantically implausible edits are applied, which would be otherwise jarring to human viewers (Fig. 2). Using saliency as the only supervision can result in unrealistic images.

To prevent unrealistic edits, prior works enforce constraints on the allowable changes, use adversarial training [2, 4, 8, 17] or exploit learned priors from GAN-based models [1, 8]. For instance, Mechrez et al. [16] and Aberman et al. (Warping) [1] constrain the result to match the input content in order to maintain its appearance. Aberman et al. (CNN and Recolorization) [1] use a regularization term that limits the amount of change an image can undergo to maintain the realism. Mejjati et al. [17] designed a global parametric approach to limit the editing operations to a set of common photographic ones. Chen et al. [2] exploit cycle consistency to keep the output within the domain of the input image. Gatys et al. [4] use a texture loss alongside the VGG perceptual loss as a proxy for realism.

Lalonde et al. [11] argue that humans prefer color consistency within images, regardless of object semantics. They use color statistics to measure realism and use it to recolor the images to match the background in compositing task. Zhu et al. [26] train a network to discriminate between natural images and computer-generated composites and use it as a realism measure for compositing task. Realism is also a crucial factor in GANs, as highlighted by [9].

Table 1. Parameter value ranges used to generate real and fake training images for the realism estimation network.

	Exposure	Saturation	Color curve	White balancing	Number of edits
Real	$[0.85, 1.15]$	$[0.85, 1.15]$	$[0.85, 1.15]$	Not allowed	$[1, 3]$
Fake	$[0.5, 0.75] \cup [1.5, 2]$	$[0, 0.5] \cup [1.5 - 2]$	$[0.5, 2]$	$[0.9, 1]$	$[2, 4]$
Fake(human specific)	$[0.5, 0.75] \cup [1.25, 1.5]$	$[0.5, 0.75] \cup [1.25, 1.5]$	$[0.5, 2]$	Not allowed	$[2, 3]$

Figure 3. The efficacy of the realism estimation network is illustrated over a range of exposure and saturation adjustments. Left is $\Delta\mathcal{R}$ plotted (vertical axis) for example images (second column) when selected regions (third column) are edited. Right, the edited images are shown with the corresponding change in estimated realism (inset numbers), and the value of the editing parameter applied (underneath).

We present a new method for estimating the realism of a local edit. Combining our realism loss with saliency guidance, we show that we can successfully apply attention attenuation or amplification while keeping the final result realistic without requiring data annotated with realism or bulky GAN priors to estimate realism.

3. Realism Network

When editing specific regions in an image, it is challenging to maintain the overall realism of the photograph. How quickly realism starts to degrade depends on the contents and size of the image regions, the overall composition of the scene, as well as the type of edits being applied. This makes the problem of defining precisely when an image edit stops looking realistic particularly challenging.

In this work, we propose to train a realism network using only realistic and unrealistic examples at the extremes. We generate realistic edits by slightly perturbing image values, and unrealistic edits by applying aggressive edits. We show that, despite being trained on binary data, our network can estimate continuous realism scores that can adapt to different types of image regions and scenes. Our approach was inspired by the work of Zhu et al. [26], who similarly learn their realism from binary real and synthetic composites.

To generate *real* and *fake* samples, we exploit different parameter ranges for commonly used editing operations – exposure, saturation, color curve, and white balancing (formal definitions in the Supplementary Material). For instance, increasing the exposure of a region too much can result in an unrealistic image, while a subtle increase to saturation will not signficantly affect the realism. Based on experimentation, we determined the set of parameter ranges in Tab. 1 to apply to image regions to create our training data.

To generate a training example, we first select a random number of edits (between 1-4), then an order for the edit operations (e.g., exposure, saturation, color curve, white balancing), and values for each of the operations, sampled uniformly at random from the pre-specified ranges in Tab. 1. We apply these edits in the selected order to a region segment in an MS-COCO image [12]. Fake examples are generated by purposefully selecting extreme values. Real examples are generated by sampling subtle edits within narrower ranges. Because of the semantic importance of human faces and increased sensitivity to edits in facial regions, we enforce smaller parameter ranges when applying edits to faces. Fig. 4 shows several examples.

We use the Pix2Pix [6] network architecture followed by two MLP layers to estimate the realism score $\mathcal{R}$ of the input. For our samples in the training data, $\mathcal{R}$ is defined as 1 for real and 0 for fake samples. We also condition the output on the input region by feeding the region's mask M as input to the network. We use squared error [13] as the critic to compute the loss on the estimated value:

$$\mathcal{L}_{\text{disc}} = \frac{1}{2}\mathcal{R}(I_{fake}, M)^2 + \frac{1}{2}(\mathcal{R}(I_{real}, M) - 1)^2 \quad (1)$$

where I_{fake} and I_{real} are the generated fake and real samples. To measure the effect of the edit on the realism of the image, we compute the difference between the scores

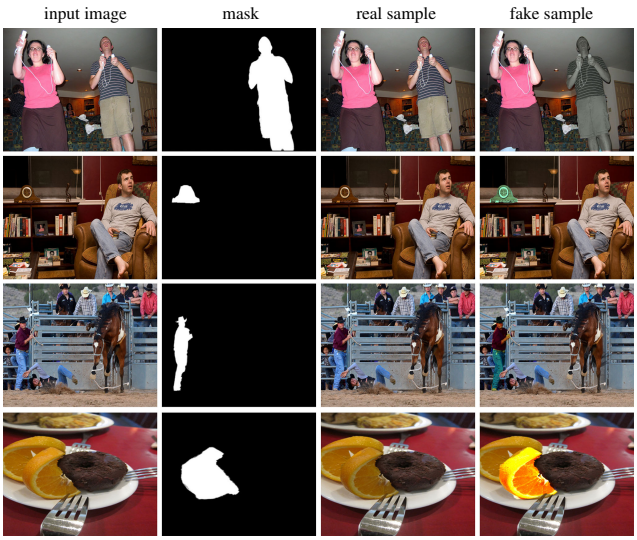


Figure 4. Examples of fake and real images that are used to train the realism estimation network. See Section 3 for more details.

estimated for the original image I and the edited image I':

$$\Delta\mathcal{R}(I', I, M) = \mathcal{R}(I', M) - \mathcal{R}(I, M), \tag{2}$$

where the edited region is defined by the mask M.

As Fig. 3 demonstrates $\Delta\mathcal{R}$ gives us continuous realism values for a range of edit parameters, despite the network being trained only on extreme cases. It also shows that the range of edits that are considered realistic by our network is not the same for each image and depends on the subject and editing operation. We show more examples of edits that are classified realistic or unrealistic by our network in Fig. 9 and the Supplementary Material.

4. Saliency Guided Image Enhancement

We develop a saliency-guided image editing pipeline that enforces our realism loss to generate realistic and effective object enhancement or distractor suppression results for a given mask. Our system can estimate a set of editing parameters for any permutation of 4 editing operators: exposure, saturation, color curve, and white balancing.

In constructing our system, we borrow many ideas from the existing saliency-guided image editing literature, and focus our design improvements on improving the realism of the results, especially by including our proposed realism loss. Since these edit operations are non-linear, different orderings of edits changes the end results. As a result, we condition the regressed parameters on the permutation of the edit operations by feeding the permutation as an input to the network. More details on the architecture of the network and the embedding used to encode the permutation is included in the Supplementary Material.

Saliency Loss A pretrained saliency model [7] (SalNet) is used as a proxy for the viewer attention that would be captured by image regions before and after applying the edits, to supervise the image editing process.

We measure the change in the saliency of the region of interest as the expected value of its relative change within the masked region:

$$S(I, I', M) = \mathbb{E}_M \left[\frac{SalNet(I) - SalNet(I')}{SalNet(I)} \right] \tag{3}$$

where $\mathbb{E}$ denotes the expectation and M is the region mask.

As Fig. 2 shows, the predicted saliency heatmaps can change drastically when applied to unrealistic edits. As a result, relying on conventional metrics (e.g., absolute and relative differences by [1, 17], Binary cross entropy by [2] and KL-divergence by [4]) to measure the change in saliency can cause large rewards or penalties during optimization. Infinitely large rewards for an unreal edit reduces the effectiveness of the realism term in the final loss function. To tackle this issue we define our saliency loss function as:

$$\mathcal{L}_{\text{sal}} = \exp\left(w_{\text{sal}} S(I, I', M)\right) \tag{4}$$

When saliency moves in the desired direction, the exponential squashes the loss, converging to the minimum and reducing the penalty quickly, acting as a soft margin. This converging behaviour prevents large rewards that can be generated by unrealistic edits during training. The exponential term imposes larger penalties when saliency moves in the wrong direction, providing robustness against outliers and faster convergence. w_{sal} controls the absolute value of the loss to balance the weight of saliency loss in our final loss, which we set to 5 and -1 for amplification and attenuation, respectively.

Realism Loss The realism loss is defined as:

$$\mathcal{L}_{\text{realism}} = ReLU(-\Delta\mathcal{R}(I', I, M) - b_r) \tag{5}$$

This loss is designed to penalize unrealistic edits, while giving no rewards for edits that improve the estimated realism score of the input. This prevents the network from being penalized by images that receive a low realism score even before any edits are applied. ReLU and offset b_r provide a margin that allows a slight decrease in realism without a penalty which we set to 0.1 in our experiments.

We train two separate networks for each. The final network objective is the product of the two loss functions:

$$\mathcal{L} = (1 + \mathcal{L}_{\text{realism}}) \times \mathcal{L}_{\text{sal}}. \tag{6}$$

In this formulation, the realism score acts as a weight for the penalty imposed on the change in the saliency. This allows us to balance the realism and saliency objectives.

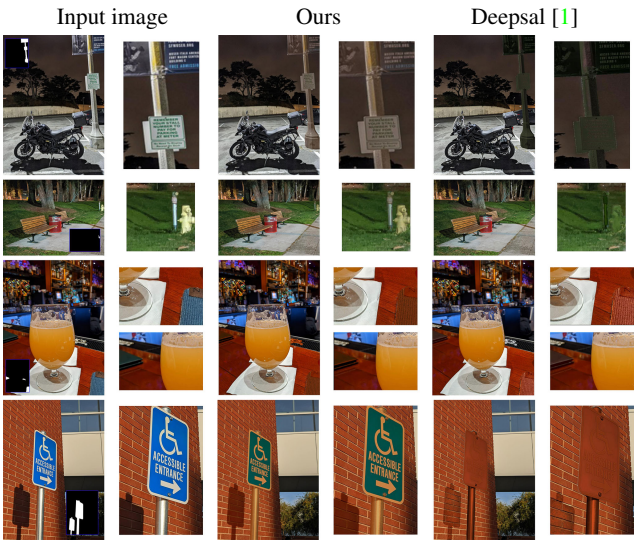


Figure 5. Saliency attenuation compared to Deepsal [1] on the images provided by the authors on their project webpage. Our method is able to effectively attenuate the saliency of the target region without applying an unrealistic camouflage.

We use an EfficientNet-lite3 [21] backbone and cascaded MLP layers as decoders to estimate parameters for each of the edit operations. A detailed explanation of the architecture specifics, datasets and training is provided in Supplementary Material.

5. Experiments and Results

We compare our method against state of the art saliency based image editing approaches – Deepsal [1], Gazeshift [17] and MEC [16]. MEC provides results on their dataset alongside pre-computed results of WRS [23] and OHR [15] on the same dataset. We use this dataset to compare against WRS and OHR as well as MEC. [1]

The EfficientNet [21] backbone used in our architecture is known for its small size. Our results are thus generated significantly faster than the other state-of-the-art (SOTA) methods with bulkier architectures and slower per-image optimizations. Based on speed measures reported in [17] Table 1c, MEC takes more than a day, OHR needs 30 seconds and Gazeshift takes 8 seconds to process each image, while our model requires only 8ms per image.

We present both qualitative and quantitative results. Since our method takes the permutation of the edits as input during inference time we select the permutation at random for the presented results unless mentioned otherwise.

[1] Deepsal, WRS and MEC do not provide an open-source implementation. Hence, we relied on the results included on their project pages. Also, Deepsal authors kindly provided us with results on Adobe Stock dataset for their "convolutional network" variation.

5.1. Qualitative Comparison

Figs. 5, 6, and 7 illustrate our results compared to the SOTA. They show our method performs different edits based on the contents of the image. It can apply more significant color changes that camouflage the distractor (2nd and 4th rows of Fig. 6, 3rd row of Fig. 5) or very subtle edits for human faces (1st row of Fig. 6). The intensity and characteristics of the applied edits depends on semantics.

The use of adversarial loss in Gazeshift [17] and the regularization used in Deepsal [1] constrain the edits their methods apply without taking realism explicitly into account. As results show, they often apply unrealistic edits (e.g., the camouflaged signs in Fig. 5 or the unattural skin tone and the color artifacts in Fig. 6) or very subtle edits with lower effectiveness.

MEC [16] reuses the color patterns and textures available in the image to update the target region. On the other hand different regions and textures can correspond to different semantics. Consequently, as illustrated in Fig. 7a this method can apply incompatible color and texture values to produce unrealistic edits (green crocodile eye, orange traffic sign) or ineffective enhancements (brown bird). Fig. 7b provides a comparison on their *distractor suppression* image set. Our method performs comparable in terms of effectiveness and generates realistic results consistently.

OHR [15] tries to maximize the color distinction between the masked region and the rest of the image for the image enhancement task. Without explicit realism modeling, it tends to generate unrealistic colors (e.g., blue crocodile, bird, and horse in Fig. 7a). While incorrect colors increase the saliency of these regions, they do so at the cost of realism. For similar reasons, this method is ineffective suppressing distractors (Fig. 7b).

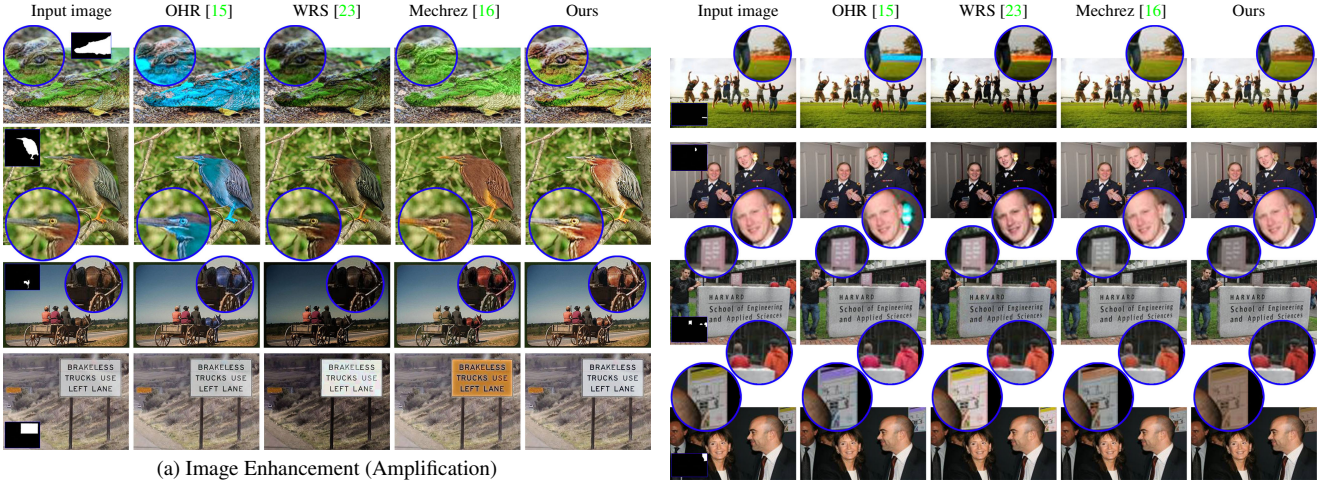
WRS [23] does not generate unrealistic images, but also makes edits that are hardly noticeable, and less effective at enhancing or suppressing the target regions. We believe this is due to the purposely limited range of allowed edit parameters (luminance, saturation and sharpness).

5.2. What Do Photographers Think?

To include the perspective of professional photographers in comparing our results to others, we ran a user study. We report our results using three choices of parameter orderings: choosing the one that achieves the *Best Saliency*, the one that generates the *Best Realism* (according to our realism estimation network), and a permutation of parameters selected at *Random* as used for the qualitative figures.

User Study We recruited 8 professionals from UpWork, all of whom have multiple years of experience with photography, as well as using Photoshop to edit photos. We used the Appen platform for hosting our rating tasks.

Figure 6. Saliency modulation compared to GazeShift [17] and Deepsal [1] on Adobe Stock images from [17].



(a) Image Enhancement (Amplification)

(b) Distractor Suppression (Attenuation)

Figure 7. Saliency modulation compared to MEC [16], WRS [23] and OHR [15] on the Mechrez dataset [16].

Our study participants were presented with a panel of 3 images: the original image, mask, and an edited result from one of methods evaluated. They were asked to *"rate each image based on 2 criteria"* - effectiveness and realism, with the following definitions provided for the *attenuate* version of the task: *"The images were edited to make certain objects and regions less distracting. An image edit is effective if the masked objects/regions have indeed become less distracting. An image edit is realistic if the photo does not look edited."* For the *amplify* version of the task, the wording for effectiveness was modified to: *"The images were edited to make certain objects and regions pop-out (more attention-capturing, or salient). An image edit is effective if the masked objects/regions have indeed become more attention-capturing."* Images were randomly shuffled in each task, so the photographers rated the images and methods independently of each other.

Results In Tab. 2 we compare our approach to GazeShift and Deepsal on the 30 Adobe Stock images from [17]. We find that our approach achieves significantly higher scores for both effectiveness and realism compared to GazeShift in

Table 2. Photographer ratings (on a 1 to 10 scale, higher is better) of effectiveness (i.e., achieve objective of attenuation or amplification of saliency) and realism (i.e., photo looks natural) on the dataset of 30 Adobe stock images. Numbers are the mean score across 8 photographers, with standard deviation in parentheses.

	Saliency Attenuation		Saliency Amplification	
Method	Effectiveness ↑	Realism ↑	Effectiveness ↑	Realism ↑
GazeShift [17]	4.78 (2.89)	5.93 (3.13)	7.36 (2.37)	7.07 (2.76)
DeepSal [1]	4.04 (2.90)	8.49 (2.72)	-	-
Ours - Best Realism	6.56 (2.73)	6.78 (2.70)	7.39 (2.17)	8.31 (1.89)
Ours - Random	6.36 (2.79)	6.34 (2.88)	7.36 (2.21)	8.27 (1.94)
Ours - Best Saliency	6.64 (2.79)	6.31 (2.70)	7.50 (2.08)	8.15 (2.10)

Table 3. Photographer ratings as in Tab. 2 on (a) Mechrez [16] dataset and (b) the 14 images from DeepSal project webpage [1]

	Saliency Attenuation	
Method	Effectiveness ↑	Realism ↑
Deepsal [1]	7.08 (2.84)	5.82 (3.43)
Ours - Random	6.83 (2.52)	7.41 (2.70)

(a)

	Saliency Amplification	
Method	Effectiveness ↑	Realism ↑
MEC [16]	7.06 (2.68)	7.31 (2.93)
WRS [23]	5.41 (3.22)	7.97 (2.70)
OHR [15]	7.04 (3.04)	5.18 (3.76)
Ours - Random	6.24 (2.9)	8.88 (1.74)

(b)

the attenuation task. This matches our qualitative observations that GazeShift is not successful at the task of attenuating distractor. GazeShift specializes in amplifying saliency in image regions, and we achieve similar performance on this task, while also maintaining significantly higher realism levels. In addition, results show a poor effectiveness score for Deepsal as a result of subtle edits in Fig. 6. Subtle edits mean the realism score remains high since the results are almost identical to the original images.

Since Deepsal was ineffective on Adobe Stock images, to provide a fair comparison we also compare to Deepsal on 14 images they provided on their project page in Tab. 3a. We achieve significantly higher realism scores while being similarly effective at reducing the saliency of the distractors. This matches our qualitative observations that Deepsal edits can be quite extreme and not always photo realistic.

Tab. 3b shows user study results on Mechrez dataset [16].[2] We used 77 images from the dataset to perform the user study. Results confirm that our results are superior in the realism while we achieve a comparable effectiveness compared MEC. WRS's low effectiveness yields a high realism score as its results are almost identical to the input; while the unrealistic color changes by OHR result in low realism and effectiveness scores.

5.3. Ablation Study

We trained a variation of our method in which instead of a fixed realism score estimation model we used a discriminator as adversarial loss. We trained the discriminator as part of an adversarial training approach, similar to related work [2, 4, 17]. We used the input image as the real sample and the generated image as the fake sample during training. Fig. 8 illustrates sample results with this training strategy. Since the discriminator is trained to penalize "any edits" applied in the previous steps of training it encourages the network to apply subtle edits and hence a drop in effectiveness of the method. On the other hand, due to the lack of explicit realism training, the edits are unrealistic while the effectiveness is reasonable. Ratings reported in Tab. 4 also confirm our findings.

[2]Dataset has only 10 images for attenuation task, which is inadequate for a meaningful user study. Hence we only provide amplification results.

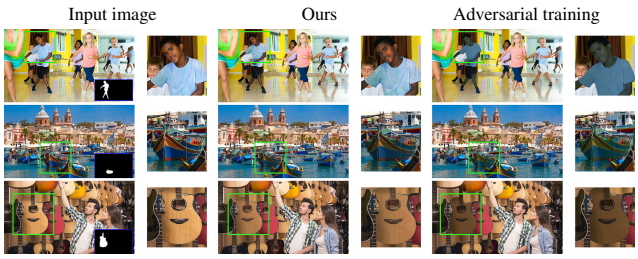


Figure 8. When the model trained via adversarial training produces results that are effective at reducing saliency, the resulting images are not realistic according to our user study.

Table 4. Photographer ratings as in Tab. 2 comparing our main method to a variation with adversarial training instead of our fixed realism network.

	Saliency Attenuation	
Method	Effectiveness ↑	Realism ↑
Adversarial Training	5.06 (2.84)	7.36 (3.07)
Ours - Random	6.36 (2.79)	6.34 (2.88)

5.4. Diversity and Optimality of Estimated Edits

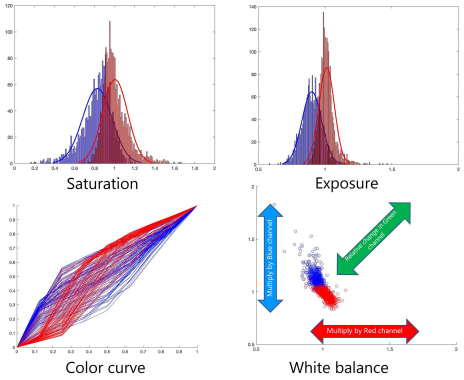
Fig. 9b illustrates the distribution of edit parameters estimated by our parameter estimation network for different images on ADE20K [24, 25] dataset. It shows that edit parameters are different for each image and is based on its content. Also, it shows that the range of estimated edits is not the same as the ranges used in Tab. 1 for real samples.

To evaluate if the estimated edits are close to optimal with respect to realism, we provide Fig. 9a. In the figure we show a realism heatmap obtained by adding a small additive constant to the estimated edit parameter of *saturation* and *exposure*. Heatmaps shows the estimated edit parameters (center of the heatmap) are in the optimal realism region. Changing the edit parameters in each direction reduces the realism of the end result.

5.5. Generalization to Multiple Image Regions

Since our model only modifies the region of interest, and performs a forward pass efficiently, we can run it on multiple regions and multiple masks by generating edit parameters for each region, in an iterative manner. Examples are provided in Figs. 1 and 10. We used the same approach with Gazeshift [17], which edits the whole image by estimating two sets of edit parameters, one for the region of interest (foreground) and one for the background. This formulation of Gazeshift makes iterative editing impractical, since there would be contradictory objectives between the

(a) A heatmap visualizes 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.



(b) The diversity of estimated parameters on ADE20K [24, 25] dataset. The x-axis is the range of each parameter. The attenuation task is blue, and amplification is labeled red.

Figure 9. Visualizing diversity and optimality of edit pararmeters estimated by our method

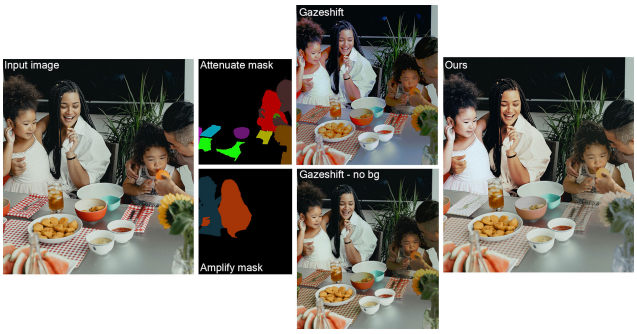


Figure 10. Given an input image and masks to attenuate and amplify(left), Gazeshift when used iteratively on each object suffers from color artifacts (center top, faces, bowl and watermelons). Ours produces a notably more realistic and effective result (right). Contradictory objective of edits applied to background and foreground, Gazeshift fails to generalize to multiple regions and omitting the background edits (center bottom) reduces the effectiveness of the edits. Image credit: @tysonbrand

iterations (what is foreground in one iteration becomes a background in the next iteration). For a more practical comparison, we omit background edits when running Gazeshift. Figure 10 shows that Gazeshift performance suffers on an iterative saliency enhancement task, but our method is able to generalize to multiple regions robustly.

5.6. Limitations

The global edits (applying the same edits to every pixel inside a mask), used in both our method and Gazeshift [17] require an accurate mask of the target region. As shown in Fig. 11 mask imperfections can cause unsmooth transitions around the boundaries. In these cases, pixel-wise optimization approaches likes Deepsal [1] and MEC [16] do not suffer from heavy artifacts due to mask imperfections.

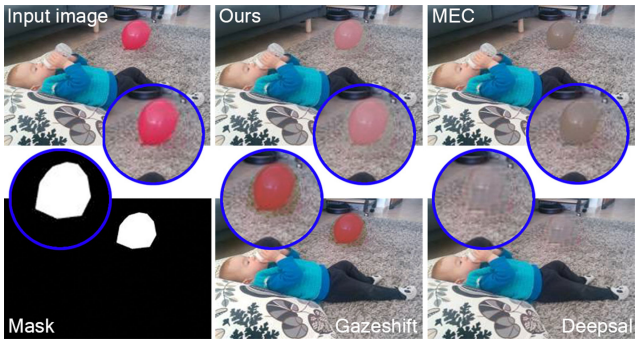


Figure 11. The effect of non-smooth mask boundaries. Left, an input image has a mask with a sharp edge. Center, our method and Gazeshift [17] produce strong boundary artifacts around the mask region (see inset). Right, MEC [16] and Deepsal [1] do not exhibit this problem because they operate in a pixel-wise manner.

6. Conclusion and Future work

We describe a method to edit images using conventional image editing operators to attenuate or amplify the attention captured by a target region while preserving image realism. Realism is achieved by introducing an explicit, and separate realism network that is pre-trained to distinguish edited images. This strategy to achieve realism is distinct from prevailing approaches, including adversarial training schemes, as it introduces an additional form of weak supervision—manually specified ranges of parameter values that correspond to realistic and unrealistic ("fake") edits. Training with this realism critic makes it possible to estimate saliency modulating image edits that are significantly more realistic and robust. Together with our millisecond-level inference time, our approach offers a practical and deployable application of saliency guided image editing.

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. 1, 2, 4, 5, 6, 7, 8

[2] Yen-Chung Chen, Keng-Jui Chang, Yi-Hsuan Tsai, Yu-Chiang Frank Wang, and Wei-Chen Chiu. Guide your eyes: Learning image manipulation under saliency guidance. In *Proc. BMVC*, 2019. 1, 2, 4, 7

[3] Camilo Fosco, Anelise Newman, Pat Sukhum, Yun Bin Zhang, Nanxuan Zhao, Aude Oliva, and Zoya Bylinskii. How much time do you have? modeling multi-duration saliency. In *Proc. CVPR*, 2020. 1, 2

[4] Leon A. Gatys, Matthias Kümmerer, Thomas S. A. Wallis, and Matthias Bethge. Guiding human gaze with convolutional neural networks. `arXiv:2109.01980 [cs.CV]`, 2017. 1, 2, 4, 7

[5] Aiko Hagiwara, Akihiro Sugimoto, and Kazuhiko Kawamoto. Saliency-based image editing for guiding visual attention. In *Proc. PETMEI*, 2011. 2

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proc. CVPR*, 2017. 3

[7] Sen Jia and Neil D.B. Bruce. EML-NET: An Expandable Multi-layer NETwork for saliency prediction. *Image Vis. Comput.*, 95:103887, 2020. 1, 2, 4

[8] Lai Jiang, Mai Xu, Xiaofei Wang, and Leonid Sigal. Saliency-guided image translation. In *Proc. CVPR*, 2021. 1, 2

[9] Alexia Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard gan. In *Proc. ICLR*, 2019. 2

[10] Matthias Kummerer, Thomas SA Wallis, Leon A Gatys, and Matthias Bethge. Understanding low-and high-level contributions to fixation prediction. In *Proc. CVPR*, 2017. 1, 2

[11] Jean-Francois Lalonde and Alexei A Efros. Using color compatibility for assessing image realism. In *Proc. ICCV*, 2007. 2

[12] 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. 3

[13] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proc. ICCV*, 2017. 3

[14] R. Margolin, A. Tal, and L. Zelnik-Manor. What makes a patch distinct? In *Proc. CVPR*, 2013. 1, 2

[15] Victor A. Mateescu and Ivan V. Bajić. Attention retargeting by color manipulation in images. In *Proc. PIPV*, 2014. 2, 5, 6, 7

[16] Roey Mechrez, Eli Shechtman, and Lihi Zelnik-Manor. Saliency driven image manipulation. *Mach. Vis. Appl.*, 30(2):189–202, 2019. 1, 2, 5, 6, 7, 8

[17] 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. 1, 2, 4, 5, 6, 7, 8

[18] Tam Nguyen, Bingbing ni, Hairong Liu, Wei Xia, Jiebo Luo, Mohan Kankanhalli, and Shuicheng Yan. Image re-attentionizing. *IEEE Trans. Multimed.*, 15(8):1910–1919, 2013. 2

[19] Junting Pan, Cristian Canton, Kevin McGuinness, Noel E. O'Connor, Jordi Torres, Elisa Sayrol, and Xavier and Giro-i Nieto. Salgan: Visual saliency prediction with generative adversarial networks. `arXiv:1701.01081 [cs.CV]`, 2017. 1, 2

[20] Sara L. Su, Frédo Durand, and Maneesh Agrawala. De-emphasis of distracting image regions using texture power maps. In *Proc. APGV*, 2005. 2

[21] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proc. ICML*, 2019. 5

[22] Javier Vazquez-Corral and Marcelo Bertalmío. Gamut mapping for visual attention retargeting. In *Proc. CIC*, 2017. 2

[23] Lai-Kuan Wong and Kok-Lim Low. Saliency retargeting: An approach to enhance image aesthetics. In *Proc. WACV*, 2011. 2, 5, 6, 7

[24] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proc. CVPR*, 2017. 7, 8

[25] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *Int. J. Comput. Vision*, 127(3):302–321, 2019. 7, 8

[26] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. Learning a discriminative model for the perception of realism in composite images. In *Proc. ICCV*, 2015. 2, 3