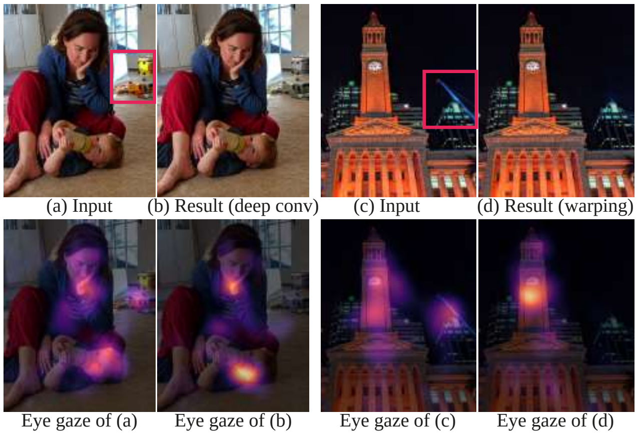
Deep Saliency Prior for Reducing Visual Distraction - Supplementary Material

Anonymous CVPR submission

Paper ID 5058


(a) Input (b) Result (deep conv) (c) Input (d) Result (warping)

Eye gaze of (a) Eye gaze of (b) Eye gaze of (c) Eye gaze of (d)

Figure 1. Examples of real eye-gaze saliency maps measured in our perceptual study, involving 20 subjects and 31 images. Top row: each pair shows an original image (left) with a region of interest (red border) and our result (right). Bottom row: the corresponding average eye-gaze maps across participants in the study.

Correction: Please note that there is an error in Figure 8 in the submitted paper. In the top row, (b) should be the input and (a) should be the result. The corrected figure is shown here in Figure 1.

1. Image regularization as a baseline

In Figure 2 of the main paper, we've demonstrated how a naive and direct optimization on the region-of-interest's pixels may lead to an out-of-distribution (adversarial attack) result. Here, we show that adding a regularization term to equation (1) in the paper is insufficient neither. To apply the regularization, we expand the main loss function to be

$$\mathcal{L}_{\text{sal}}\left(\tilde{\mathbf{I}}\right) + \beta\mathcal{L}_{\text{sim}}\left(\tilde{\mathbf{I}}\right) + \gamma\mathcal{L}_{\text{reg}}\left(\tilde{\mathbf{I}}\right),$$

where $\mathcal{L}_{\text{reg}}\left(\tilde{\mathbf{I}}\right)$ is a regularization loss on the pixels of $\tilde{\mathbf{I}}$, e.g., image total variation

$$\mathcal{L}_{\text{reg}}\left(\tilde{\mathbf{I}}\right) = M \circ (\|\nabla_x \tilde{\mathbf{I}}\|_1 + \|\nabla_y \tilde{\mathbf{I}}\|_1),$$

where ∇_x and ∇_y represent the gradients with respect to the horizontal/vertical axes x and y, respectively.

Figure 2 shows a couple of results by a direct optimization on the region-of-interest's pixels for the loss function above with different values of the hyper-parameter γ. We can see that simple regularization is insufficient for avoiding the out-of-distribution examples, or for generating realistic images.

2. Implementation Details

First, recall that our loss functions is

$$\mathcal{L}_{\text{sal}}\left(\tilde{\mathbf{I}}\right) + \beta\mathcal{L}_{\text{sim}}\left(\tilde{\mathbf{I}}\right) + \gamma\Gamma(\theta), \tag{1}$$

where $\tilde{\mathbf{I}} = O_\theta(\mathbf{I})$, and $\mathcal{L}_{\text{sal}}$ is the saliency loss term, and $\mathcal{L}_{\text{sim}}$ is the similarity loss term.

2.1. Saliency loss

In (1), $\mathcal{L}_{\text{sal}}$ is defined as

$$\mathcal{L}_{\text{sal}}\left(\tilde{\mathbf{I}}\right) = \left\|\mathbf{M} \circ \left(S(\tilde{\mathbf{I}}) - \mathbf{T}\right)\right\|^2.$$

The target map $\mathbf{T}$ can be an arbitrary map, where the saliency can be controlled. For example, Figure 3 shows an example where the saliency level can be controlled by a single scalar with $\mathbf{T} = \alpha S(\mathbf{I})$, for various values of α that are indicated in the figure.

However, all other examples in the paper are generated by setting $\mathbf{T} \equiv 0$ to minimize the saliency within the region of interest (or $\mathbf{T} \equiv 1$ to maximize the saliency, for the

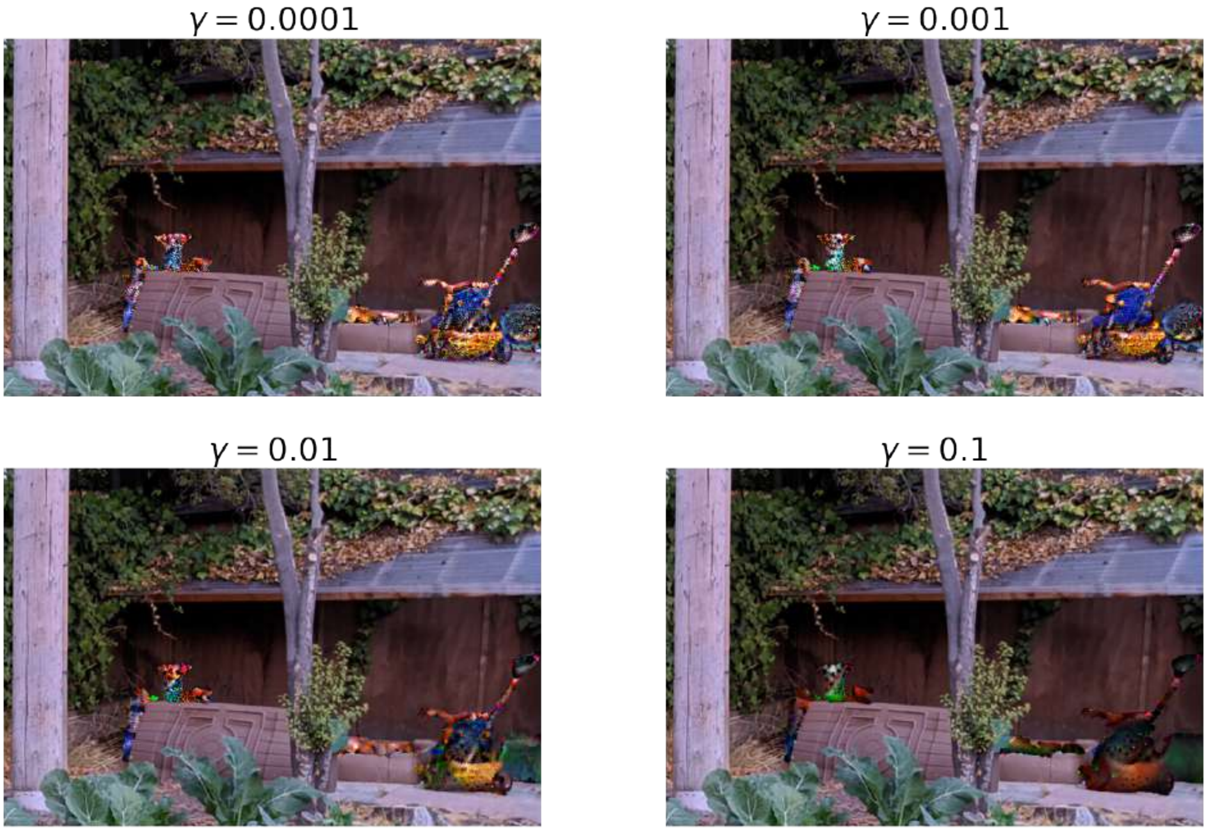


Figure 2. A naive baseline that applies a direct optimization to the region's pixels together with a total variation regularization yields out-of-distribution, unnatural results. γ is the regularization term weight described in equation (1) in the original paper.

GAN operator). When $\mathbf{T} \equiv 0$ or 1, $\mathcal{L}_{\text{sal}}\left(\tilde{\mathbf{I}}\right)$ is equivalent to $\left\|\mathbf{M} \circ S(\tilde{\mathbf{I}})\right\|^2$ for saliency decreasing or $-\left\|\mathbf{M} \circ S(\tilde{\mathbf{I}})\right\|^2$ for saliency increasing, respectively.

2.2. Similarity loss

In equation (1), the similarity term in the loss is defined as

$$\mathcal{L}_{\text{sim}}\left(\tilde{\mathbf{I}}\right) = \left\|(1 - \mathbf{M}) \circ \left(\tilde{\mathbf{I}} - \mathbf{I}\right)\right\|^2.$$

Another way to preserve similarity of pixels outside of the mask is using a hard constraint such as $(1 - \mathbf{M}) \circ \left(\tilde{\mathbf{I}} - \mathbf{I}\right) = 0$. We can enforce this constraint by copying pixels outside of mask $\mathbf{M}$ in $\mathbf{I}$ to $\tilde{\mathbf{I}}$ in each iteration step. While "Learning convolutional networks" and GAN operator adopt $\mathcal{L}_{\text{sim}}$ as equation (1), our recolor and warp operators adopt this hard constraint approach. Such a hard constraint enables us to crop the image around the distracting region when the region of interest does not have enough saliency in the original image. Namely, we can crop the image to a smaller one that still contains the region of interest - a step which enables to gain high-saliency, which can be removed more effectively. Since pixels outside of the mask will be exactly the same, the crop will not create any artifacts around the cropping boundary. This is helpful when the region of interest is too small or not very salient in the original, large image, that contains more attention grabbing regions.

2.3. Hyper parameters

- For the "recolor" operator, the hyper parameters include weight parameter $\gamma = 0.01$, learning rate = 10, number of iterations $= 800$.
- For the "warp" operator,, the hyper parameters include weight parameter $\gamma = 0.1$, learning rate = 1, number of iterations $= 500$.
- For the "deep conv" operator, the hyper parameters include weight parameters $\beta = 100$, $\gamma = 0$, learning rate = 0.1, number of iterations $= 1000$.
- For "GAN" operator, we set $\gamma = 0$, $\beta = 0.03$, learning rate $= 0.01$, number of iterations $= 200$.

2.4. GAN operator

In order to apply our GAN operator to real images, the image should be first embedded in latent codes, e.g., with a GAN inversion technique. Existing works [8, 16] show that

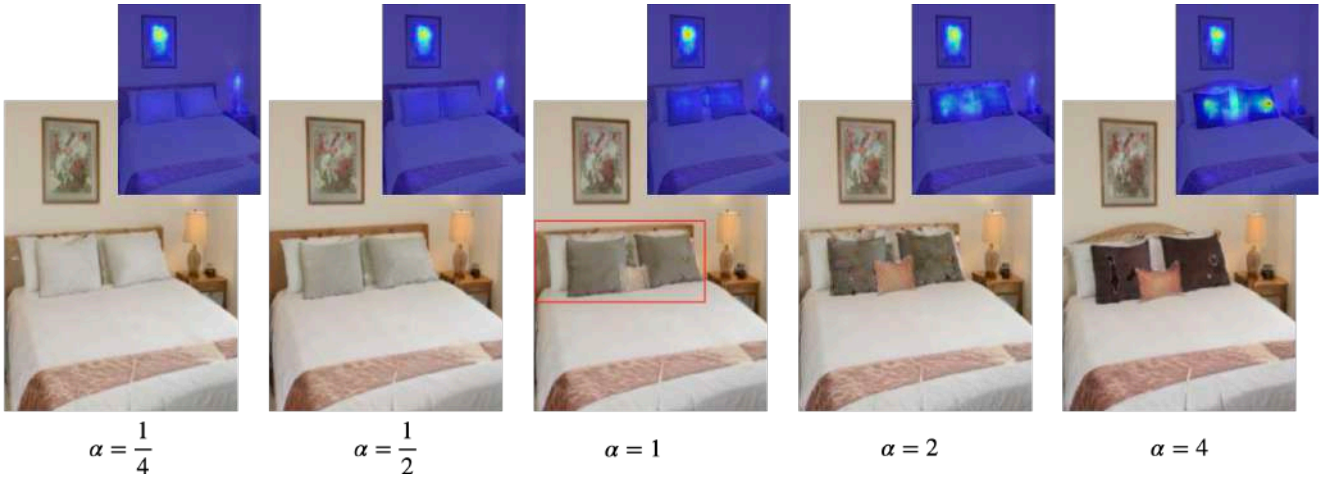


Figure 3. Varying the target saliency level. Given an image (middle) and a region of interest (marked by a red box) our approach can control the saliency level in the specified region of interest. The target saliency is α times the original saliency (within the mask), i.e, $\mathbf{T} = \alpha S(\mathbf{I})$.

in-domain images (i.e., images similar to the GAN training set) can be reconstructed effectively within different subspaces of StyleGAN. For example, Figure 5 shows some examples of reconstructed images with GAN inversion techniques [16]. As can be seen, the technique achieves reasonable reconstruction quality on those in-domain examples, ,however, reconstruction of some facial, fine grained details which are crucial for identity preservation need to be improved. Moreover, GAN inversion for out-of-domain images is even a more challenging task. Recently, some works [5,11] show promising results in reconstructing out-of-domain real images too. Exploring GAN inversion techniques is beyond the scope of this work, and we assume that a latent code corresponding to the input image is given, so our method focus on the modification of the latent codes to modify the content (and saliency) of the output image.

Moreover, in the main paper, we show results where the embedding of W space is edited by our saliency driven approach. For styleGAN, other embedding space like W^+ space, or style space can also be edited [1, 11, 15]. In Figure 4, we show one example for which our GAN operator edits different embedding spaces. Papers like [15] further discover the physical meaning of each dimension of style subspace, and achieve image editing via modify one or few particular subset of style vector only. Exploring those directions can be our future works.

2.5. Saliency model

For the experiments in this paper we use the saliency prediction model EML-Net of [7], with minor modifications. More specifically, we use NasNet [17] only, without DenseNet [6] in the EML-Net architecture for simplicity, since adding DenseNet will only slightly increase the accuracy (as reported in Table 1 in [7]).

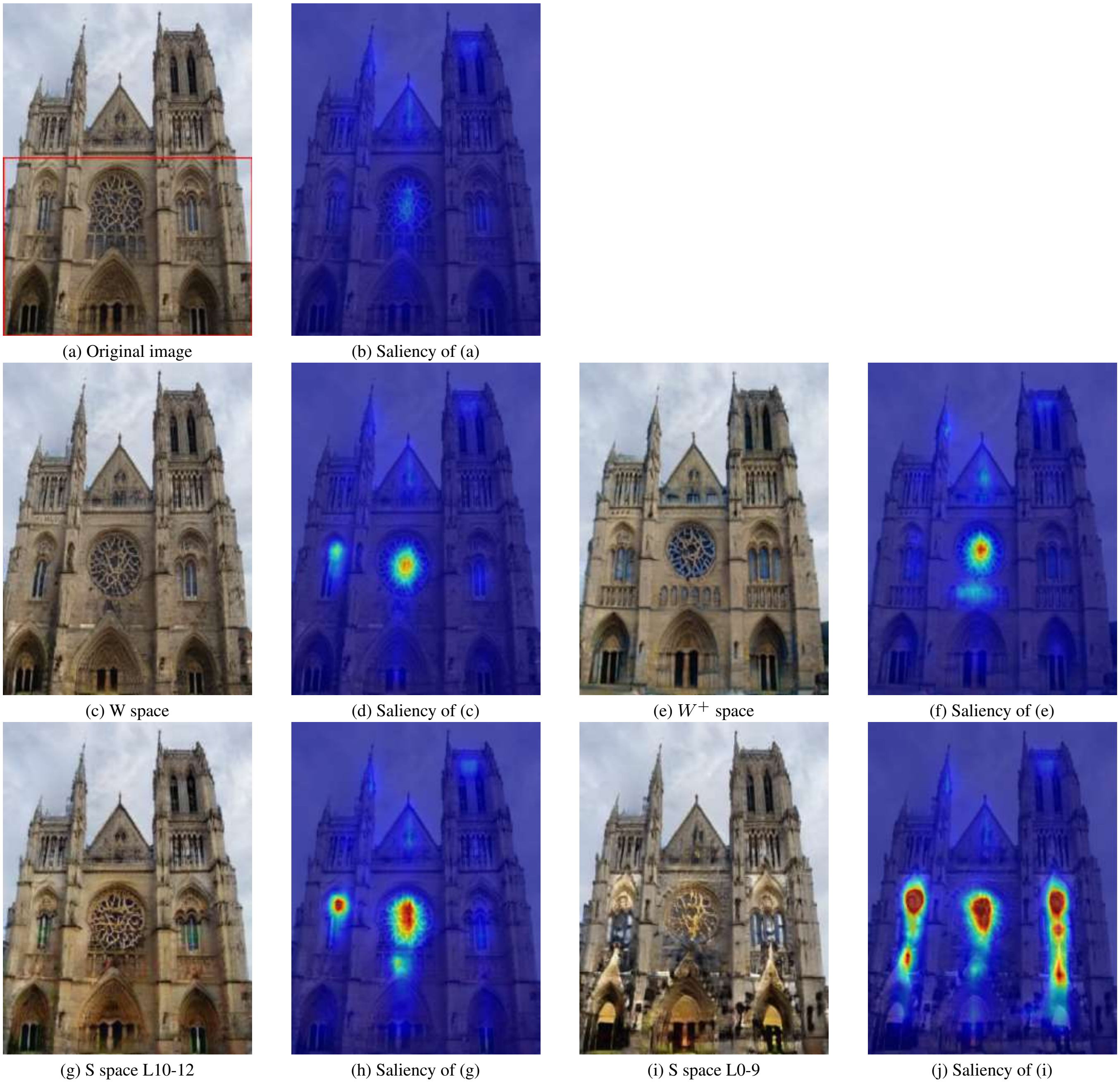
EML-Net [7] is extensively evaluated and shown to be one of state-of-the-art models [3, 7, 10]. However, most previous evaluations are conducted on real images only. To evaluate the accuracy of the saliency model on ***edited*** images, we compared the predicted maps to the ground-truth fixations on original and edited images using standard metrics: AUC-Judd, NSS, SIM and KLD [4]. As shown in Table 1, the accuracy on edited images is slightly lower than original ones. It happens since the model is trained on natural images. Although our proposed operators force the edited images to be close to a natural images distribution, small deviation (e.g, 82.5% vs 82.9% for AUC-Judd) may still exist. For reference, we also provide the accuracy metrics of a baseline center bias saliency model as a sanity check.

	AUC-Judd ↑	**NSS** ↑	**SIM** ↑	**KLD** ↓
original (p)	82.9%	1.275	60.7%	0.745
manipulated (p)	82.5%	1.167	57.9%	0.854
original (c)	68.7%	0.714	50.0%	2.208
manipulated (c)	68.4%	0.714	49.8%	2.228

Table 1. original (p): predicted saliency on original images; manipulated (p): predicted saliency on manipulated images; original (c): center bias baseline saliency on original images; manipulated (c): center bias baseline saliency on manipulated images.

3. More results/analysis on eye-gaze user study

For the three gaze metrics in Table 1 (b) in the main paper, we also ran a paired samples T-Test (original and edited images as control and observation, respectively) , to demonstrate the statistical significance for the changes on each of the three gaze metrics. As can be seen in Table 2, the p-value of each metric is < 0.003, implying statistical significance of gaze saliency/duration reduction and first



(a) Original image

(b) Saliency of (a)

(c) W space

(d) Saliency of (c)

(e) W^+ space

(f) Saliency of (e)

(g) S space L10-12

(h) Saliency of (g)

(i) S space L0-9

(j) Saliency of (i)

Figure 4. Manipulate different embedding space of styleGAN with our GAN operator to increase saliency. For (c), (e), (g) and (i), editing happens in W, W^+, layer 10-12 in style space, layer 0-9 in style space respectively.

	Duration (ms)	First gaze (ms)	Gaze saliency
signifance	p=6 × 10^{-5}	p=1.2 × 10^{-3}	p=2.2 × 10^{-5}

Table 2. Paired Samples T-Test of manipulated images vs original ones, withthree gaze features for decreasing saliency cases.

gaze time increase.

Besides the experiment to evaluate decreasing saliency as reported in the paper, we also run experiments of 13 images from our GAN operator for increasing the saliency. Example images and their gaze saliency maps for increasing saliency cases can be found in the supplementary html.

The average gaze duration is increased from 985.09 ms to 1548.23 ms, and average first gaze time is decreased from 1555.28 ms to 1210.44 ms. We have conducted the

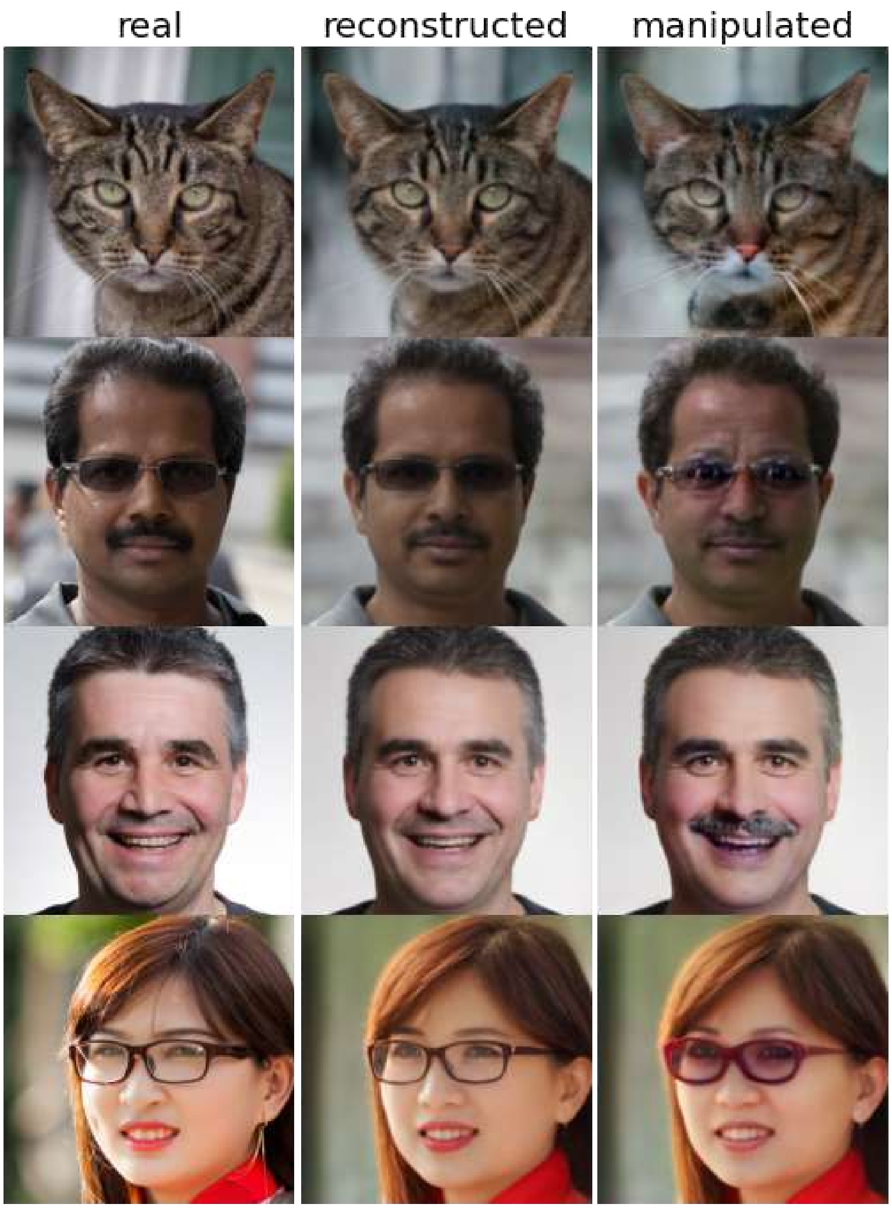


Figure 5. Reconstruction of real images by GAN inversion. The reconstructed images are then manipulated by our proposed method with the GAN operator.

Paired Samples T-Test for the three gaze features on these 13 examples, as in Table 3. We can see that we have achieved the statistical significance with these very small number of examples (except for first gaze time, which is a noisier feature essentially).

4. Implementation details for experiments

4.1. Details of eye-gaze user study

We developed a user study app on Android phones to measure users' gaze/attention on the original and manipulate images, based on the mobile eye tracking techniques introduced in [14]. The app collects calibration data by asking participants to look at moving dots on the screen, to enable high eye tracking accuracy comparable to state-of-the-art

Gaze saliency	Duration	First gaze
p=0.00055	p=0.0022	p=0.073

Table 3. Paired Samples T-Test of manipulated images vs original ones, with three gaze features for increasing saliency cases.

eye trackers with special hardware [14]. Example screen shots of the user study app are shown in Figure 6. The study was conducted on internal participants with explicit and fully informed consent, and with the option for participants to opt-out of the study or delete the data anytime during or after the study.

Generate gaze saliency The gaze saliency map is computed following the procedure in [9]. More specifically, we first convolve each gaze point with a Gaussian kernel, then take sums of all Gaussian kernels for the gaze points on the image, and normalize by the number of participants. The kernel width is chosen $1/30$ of the minimum of image width and height.

First gaze time First gaze time is defined as the time that gaze visits the region of interest for > 50ms. Note that a threshold is needed so that only gaze visit on purpose to the regions within the mask will count, ruling out accidental gaze visit. If there is no gaze within mask, first gaze time will be set as 5s, the duration one image is shown to the user.

4.2. Details of the survey user study

As mentioned in the paper, we conducted a survey user study that contains two main questions. The goal of the first was to evaluate how natural our results seem, and the second to understand what kind of effects users prefer for the task of saliency reduction. The users were asked to look at various images with a marked region of interest, together with two outputs, ours (various effects) and "look-here!", and were asked: “The following two results attempt to draw LESS attention to the region marked in red on the original image. Which one do you like better?”. Figure 7 shows a couple of screenshots from the study.

5. Examples for more operators

The proposed method is a general framework which can support other operators besides the four ones discussed in the paper, as long as the operator can constraint the solution space within the valid space of the saliency prediction model.

Another simple yet effective operator that can be used is a spectral decomposition and re-composition of the region of interest that modifies the saliency within the region. For example, we can project the image into different frequency bands and recompute the weights of each band in the reconstruction stage, such that the overall saliency in the region of interest is modified. We use the multi-layer Laplacian decomposition introduced in [12], where original image $\mathbf{I}$ is decomposed to as $\mathbf{I} = \mathbf{I}_s + \sum_{k=1,...,K} \mathbf{I}_{r_k}$, where $\mathbf{I}_{r_k} = (\mathbf{1} - L)L^{k-1}\mathbf{I}$ is the k-th residual component, while $I_s = L^K\mathbf{I}$ is the smooth component, and L can be any smoothing or low frequency filters like bilateral filter [13], Gaussian filter, Nonlocal Means filter [2], etc. $\mathbf{1}$ is the identity matrix.

Here, O_θ is a recomposition operator, as

$$\tilde{\mathbf{I}} = \mathbf{I}_s + \sum_{k=1,...,m} \mathbf{I}_{r_k} + \sum_{k=m+1,...,K} \mathbf{I}_{r_k} * W_k, \qquad (2)$$

where each weighting map W_k is upsampled from a low-resolution spatial grid θ_k, for $k = m+1, ..., K$. In other words, we will fix the smooth component and first m residual components, and only modify the remaining residual components, where m is a parameter that can be tuned depending how subtle we demand the effects to be. If we apply the above method to a RGB image directly, unnatural colors may be obtained sometimes. To avoid the issue, we can either apply the same W_k on all three R, G, B channels, or convert the image to L-ab format, and apply the decomposition and recomposition on the L-layer only. This operator can be combined with recoloring operator too, i.e, applying recomposition operator in L channel, and recoloring operator in ab channel.

Figure 8 shows a few examples that use the above spectral recomposition to increase saliency, where the operator tends to enhance detail/tone within the region of interest. Even though increasing saliency is not the major focus of this paper, we want to emphasize that the proposed framework is general enough to support different kinds of attention guiding effects with appropriate operators.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 3

[2] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005. 6

[3] Zoya Bylinskii, Tilke Judd, Ali Borji, Laurent Itti, Frédo Durand, Aude Oliva, and Antonio Torralba. Mit saliency benchmark. *http://saliency.mit.edu*, 2012. 3

[4] Zoya Bylinskii, Tilke Judd, Aude Oliva, Antonio Torralba, and Frédo Durand. What do different evaluation metrics tell us about saliency models? *IEEE transactions on pattern analysis and machine intelligence*, 41(3):740–757, 2018. 3

[5] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code gan prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3012–3021, 2020. 3

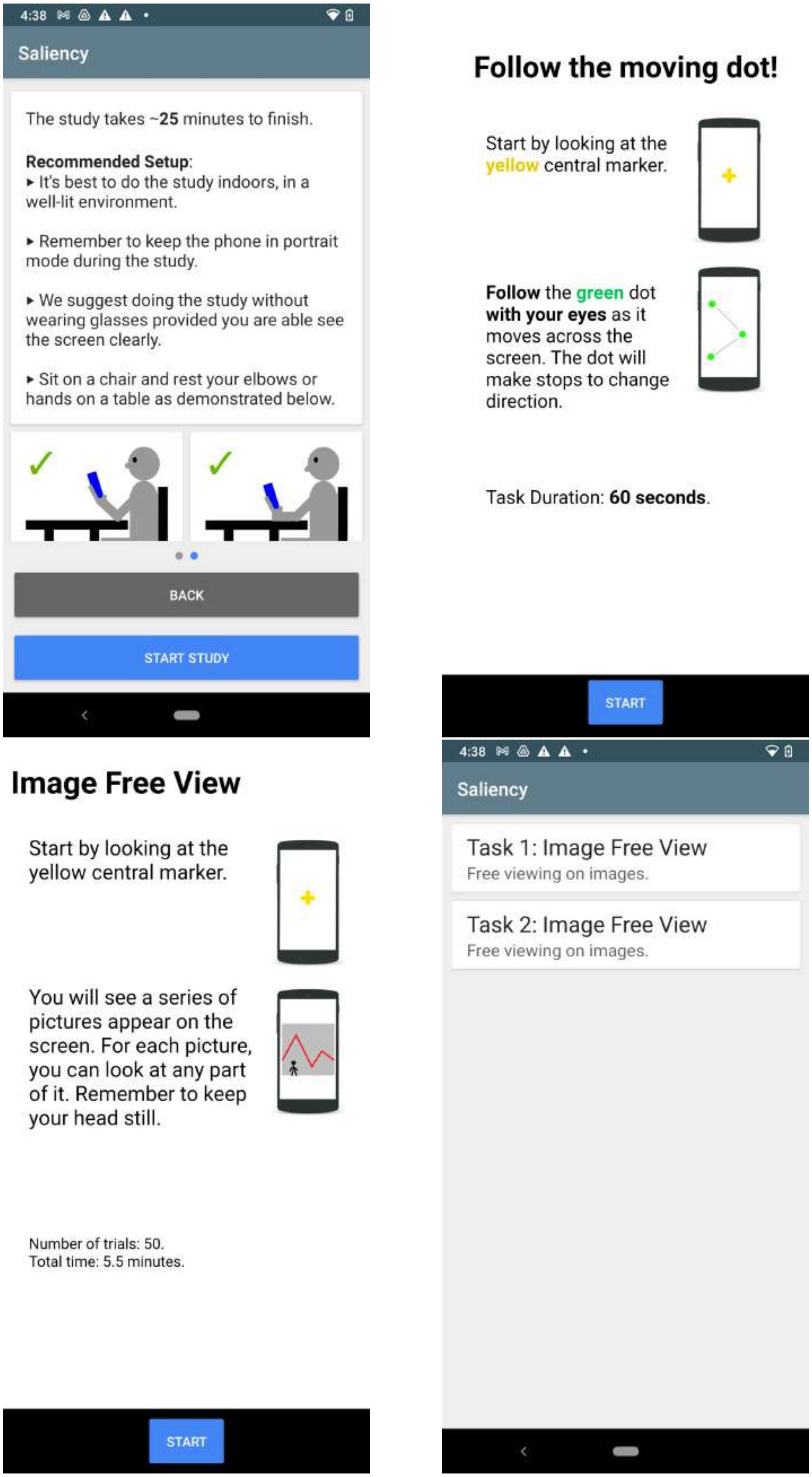
[6] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 3

[7] Sen Jia and Neil DB Bruce. Eml-net: An expandable multi-layer network for saliency prediction. *Image and Vision Computing*, 95:103887, 2020. 3

[8] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 2

[9] Olivier Le Meur and Thierry Baccino. Methods for comparing scanpaths and saliency maps: strengths and weaknesses. *Behavior research methods*, 45(1):251–266, 2013. 6

[10] Navyasri Reddy, Samyak Jain, Pradeep Yarlagadda, and Vineet Gandhi. Tidying deep saliency prediction architectures. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10241–10247. IEEE, 2020. 3



The study takes ~**25** minutes to finish.

Recommended Setup:
▸ It's best to do the study indoors, in a well-lit environment.

▸ Remember to keep the phone in portrait mode during the study.

▸ We suggest doing the study without wearing glasses provided you are able see the screen clearly.

▸ Sit on a chair and rest your elbows or hands on a table as demonstrated below.

Follow the moving dot!

Start by looking at the yellow central marker.

Follow the green dot **with your eyes** as it moves across the screen. The dot will make stops to change direction.

Task Duration: **60 seconds**.

Image Free View

Start by looking at the yellow central marker.

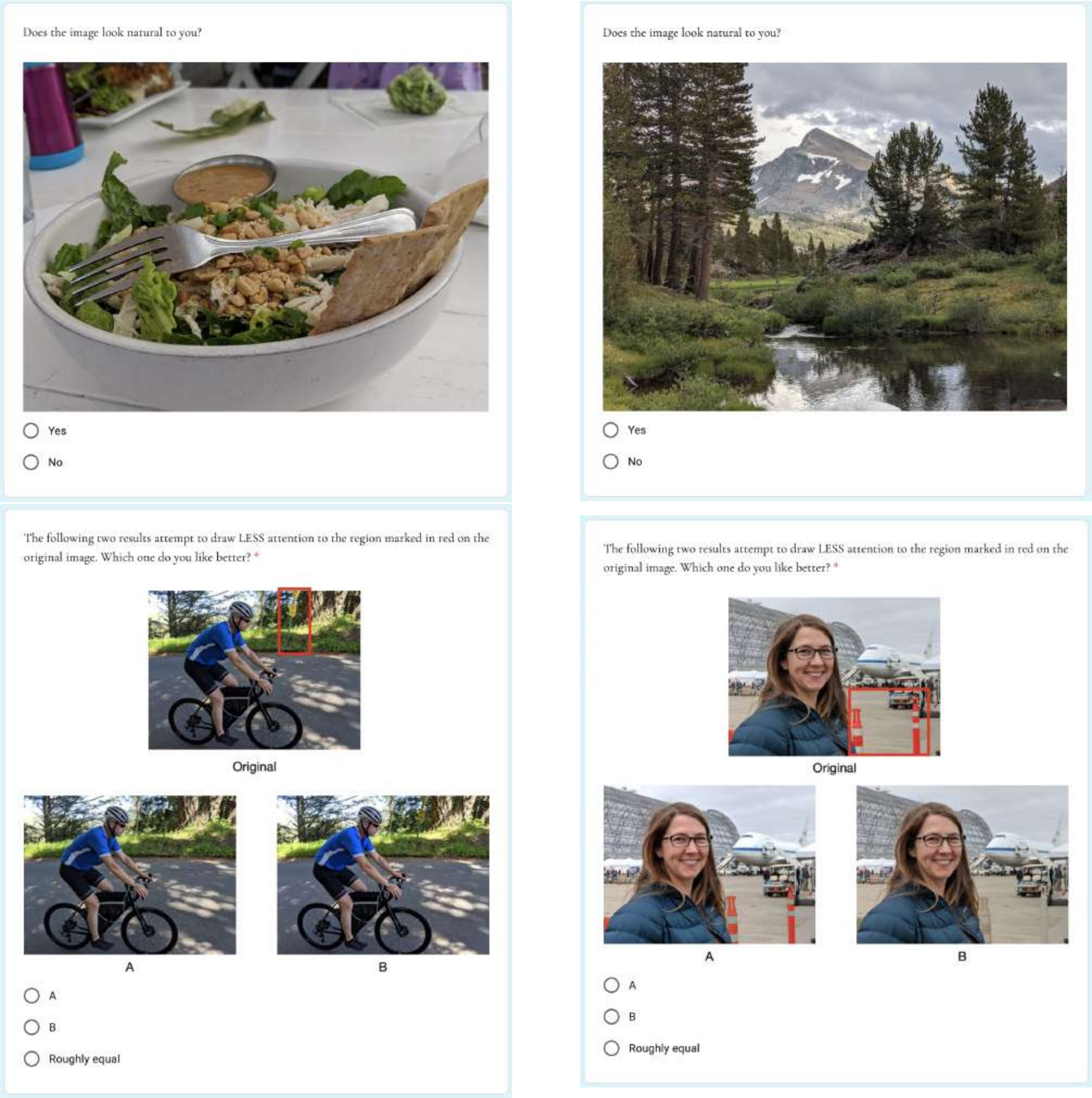
You will see a series of pictures appear on the screen. For each picture, you can look at any part of it. Remember to keep your head still.

Number of trials: 50.
Total time: 5.5 minutes.

Task 1: Image Free View
Free viewing on images.

Task 2: Image Free View
Free viewing on images.

Figure 6. Example screenshots from the eye-gaze user study app.



Does the image look natural to you?

○ Yes

○ No

Does the image look natural to you?

○ Yes

○ No

The following two results attempt to draw LESS attention to the region marked in red on the original image. Which one do you like better? *

○ A

○ B

○ Roughly equal

The following two results attempt to draw LESS attention to the region marked in red on the original image. Which one do you like better? *

○ A

○ B

○ Roughly equal

Figure 7. Example screenshots from our survey.

[11] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2287–2296, 2021. 3

[12] Hossein Talebi and Peyman Milanfar. Fast multilayer laplacian enhancement. *IEEE Transactions on Computational Imaging*, 2(4):496–509, 2016. 6

[13] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pages 839–846. IEEE, 1998. 6

[14] Nachiappan Valliappan, Na Dai, Ethan Steinberg, Junfeng He, Kantwon Rogers, Venky Ramachandran, Pingmei Xu, Mina Shojaeizadeh, Li Guo, Kai Kohlhoff, et al. Accelerating eye movement research via accurate and affordable smartphone eye tracking. *Nature communications*, 11(1):1–12, 2020. 5, 6

[15] Zongze Wu, Dani Lischinski, and Eli Shechtman. Stylespace analysis: Disentangled controls for stylegan image generation. In *Proceedings of the IEEE/CVF Conference on Computer*

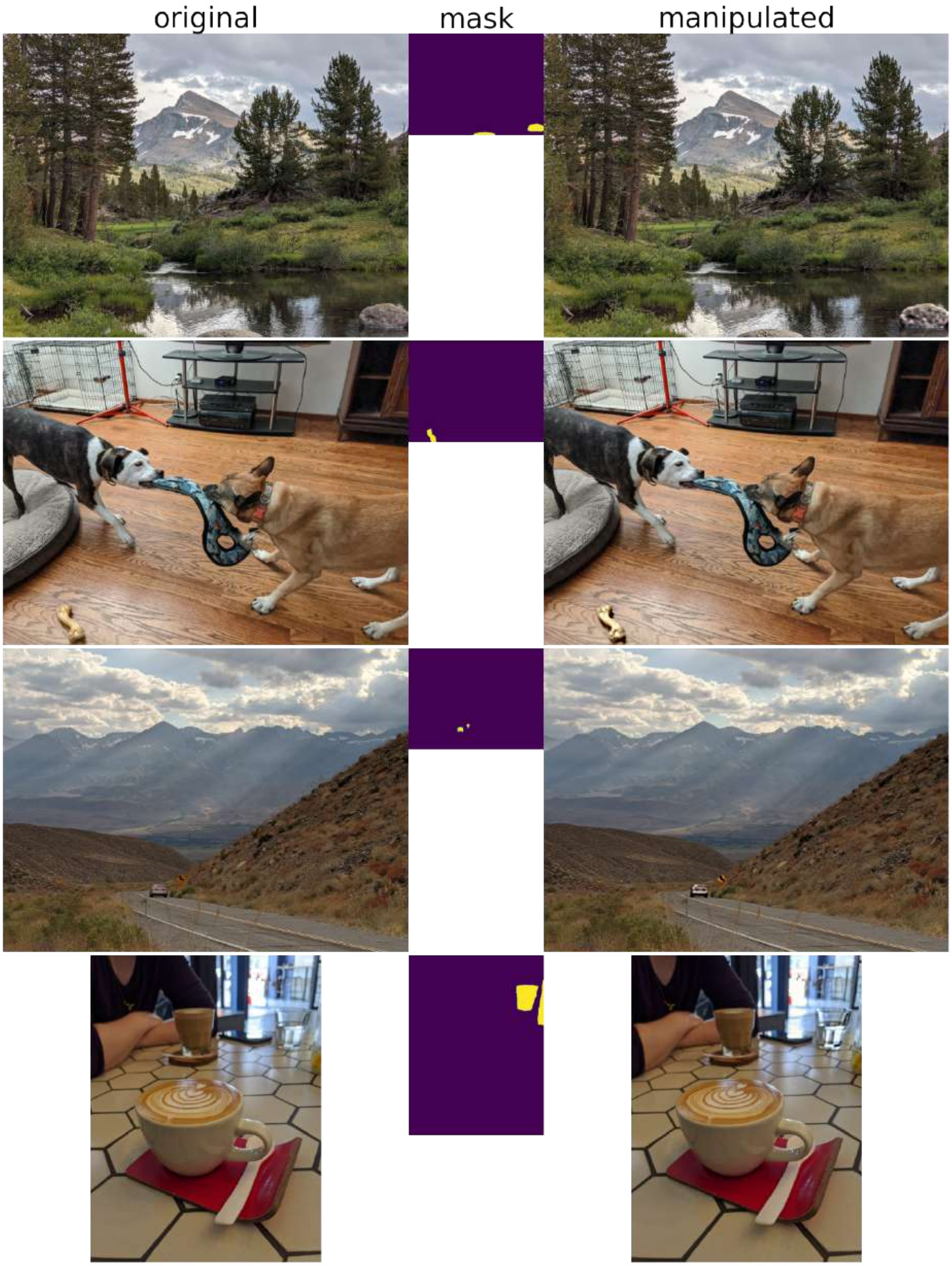


Figure 8. Increased saliency with spectral decomposition.

Vision and Pattern Recognition, pages 12863–12872, 2021. 3

[16] Jonas Wulff and Antonio Torralba. Improving inversion and generation diversity in stylegan using a gaussianized latent space. *arXiv preprint arXiv:2009.06529*, 2020. 2, 3

[17] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018. 3