Deep Edge-Aware Interactive Colorization against Color-Bleeding Effects

Eungyeup Kim[*1] Sanghyeon Lee[*1] Jeonghoon Park[*1] Somi Choi[1]
Choonghyun Seo[2] Jaegul Choo[1]

[1]KAIST, [2]NAVER WEBTOON Corp.

[1]{eykim94, shlee6825, jeonghoon_park, smchoi257, jchoo}@kaist.ac.kr, [2]choonghyun.seo@webtoonscorp.com

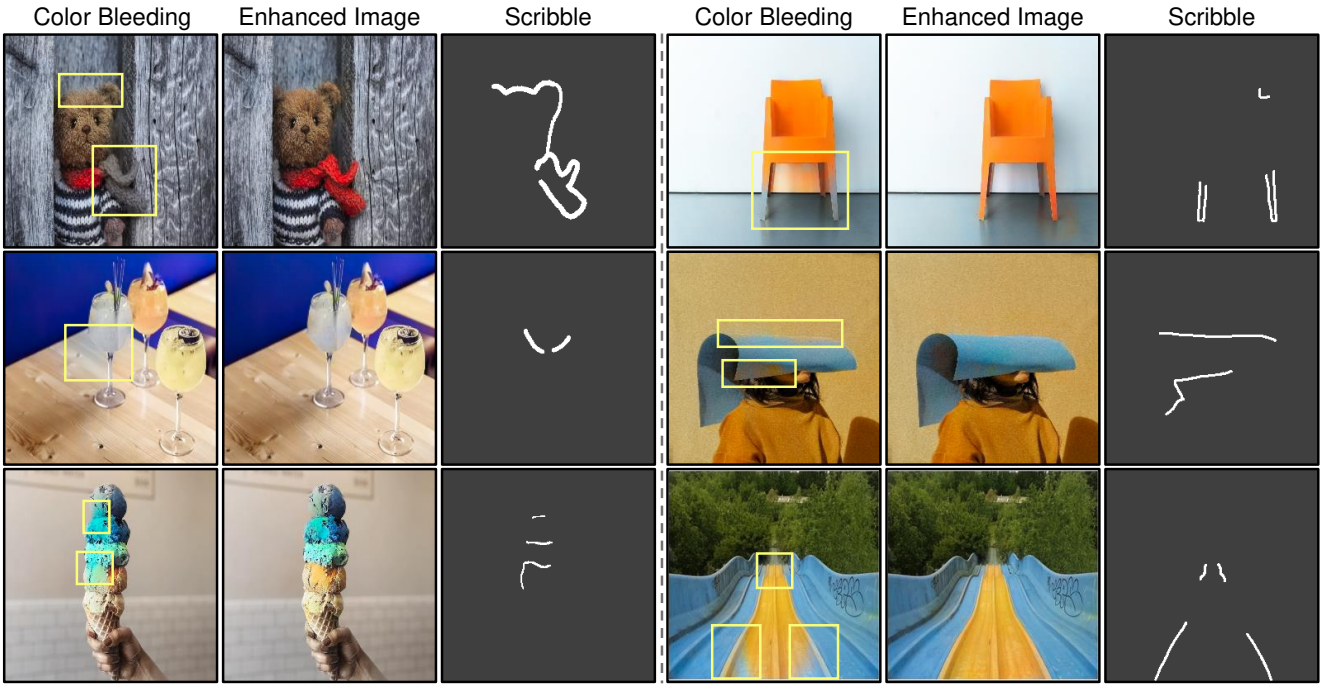

Figure 1: Qualitative results of edge enhancement by using our proposed method. These images are collected from https://unsplash.com. The first and the fourth columns show samples containing color-bleeding artifacts. Yellow boxes represent color-bleeding regions. The second and the fifth columns have edge-enhanced samples by our proposed method. The scribbles utilized for edge enhancement are shown in the third and the sixth columns. Please see our project webpage for the demo video of our method.

Abstract

Deep neural networks for automatic image colorization often suffer from the color-bleeding artifact, a problematic color spreading near the boundaries between adjacent objects. Such color-bleeding artifacts debase the reality of generated outputs, limiting the applicability of colorization models in practice. Although previous approaches have attempted to address this problem in an automatic manner, they tend to work only in limited cases where a high contrast of gray-scale values are given in an input image. Alternatively, leveraging user interactions would be a promising approach for solving this color-breeding artifacts. In this paper, we propose a novel edge-enhancing network for the regions of interest via simple user scribbles indicating where to enhance. In addition, our method requires a minimal amount of effort from users for their satisfactory enhancement. Experimental results demonstrate that our interactive edge-enhancing approach effectively improves the color-bleeding artifacts compared to the existing baselines across various datasets.

1. Introduction

In recent years, deep image colorization methods [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11] have achieved a great performance on the generation of a realistic colorized image given

* indicates equal contribution

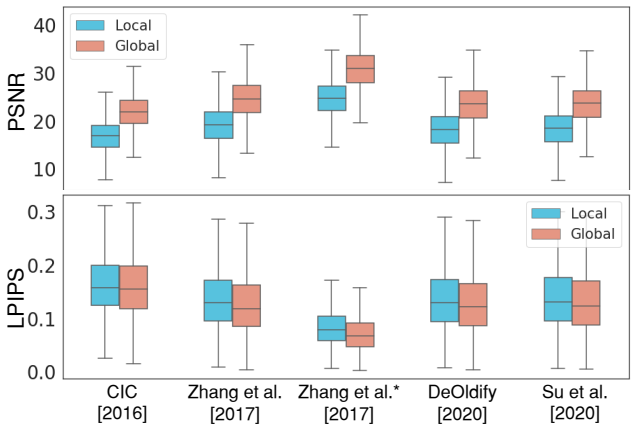

Figure 2: Comparison of local (around edges) and global PSNR and LPIPS scores of baseline models. Evaluations are conducted over ImageNet [17] ctest10k dataset, and the edges are extracted from the ground-truth, using Canny edge extractor [18]. Throughout this paper, the symbol $*$ denotes the conditional model, which takes the color hints as additional input.

a gray-scale or a sketch image. However, these methods often contain the *color-bleeding* artifact, a problematic color spreading across the adjacent objects. As shown in Fig. 1, the color-bleeding artifacts degrade the colorization quality particularly along the edges (red), compared to that in the whole image (blue). For the quantitative analysis, we also compare the quality of the colorized outputs between the existing methods [3, 12, 6, 13, 5] along the edges and the entire region in Fig. 2. The result demonstrates that existing colorization methods, including Zhang *et al*. [6] and Su *et al*. [5] which are widely used, suffer from the quality degradation particularly along the edges. Therefore, we believe there is still room for further improvement in the colorization task by resolving such color-bleeding problem.

Some approaches have addressed this issue by applying a sharpening filter on an image to colorize [14, 15, 16], or leveraging additional tasks, such as semantic segmentation, to enhance the boundaries of the semantic objects [5, 13]. However, their improvements are limited to edges appearing with strong gray-scale contrast or along the objects corresponding to the predefined categories. Color bleeding often occurs between the different objects that share similar gray-scale intensities and also along the edges that appear inside the objects, such as a zigzag pattern of a color pencil in Fig. 5. Therefore, tackling these bleeding edges at any desired locations still remains challenging, even in the recently proposed colorization methods. Moreover, evaluation of the color bleeding regions can be highly subjective depending on the users, as the plausible boundaries of the multi-modal colorized objects can differ by point of view.

Therefore, we propose a novel interactive edge enhancement framework that takes a direct user interaction annotating a color-bleeding edge. Unlike the previous approaches, our framework guarantees the reliable edge enhancement in any desired regions by utilizing user interactions. In addition, our interactive approach only requires users the minimum efforts for edge enhancement. We first apply a simple add-on edge-enhancing network, which takes both scribbles and an intermediate activation map of the colorization network as inputs. This network encodes an edge-corrective representation for its input activation map, particularly in the regions annotated by the scribbles, and adds it into the original activation map by a residual connection. Given this refined representations for the bleeding edges, the following layers of the colorization network can generate the edge-enhanced colorization output.

Experimental results demonstrate that our method has a remarkable performance over the baselines on diverse benchmark datasets, ImageNet [17], COCO-Stuff [19] and Place205 [20]. Moreover, we introduce a new evaluation metric for measuring how reliably the colorization methods obey the color boundaries. Also, we confirm that our approach takes the reasonable amount of time and efforts through the user-study, representing its potentials in practical applications . Furthermore, we explore the applicability of our approach in the task of sketch colorization as well, by validating our method on Yumi's Cells [21] and Danbooru [22] datasets.

2. Related Work

2.1. Unconditional and Conditional Colorization

Deep learning-based colorization methods [1, 3, 2, 16, 23, 24, 4, 5] have proposed fully automatic colorization approaches without any additional conditions. These unconditional models predict the most plausible colors for the given input image, even without any laborious color annotations provided by a user. By leveraging conditions given by the user, the recent colorization methods have accomplished multi-modal colorization. One of the widely used conditions is a reference image [11, 8, 9]. However, a reference image containing visually different contents from the gray-scale often induces implausible results. On the other hand, a color palette or a scribble hint given by a user directly designates the user's preference on both color and region [25, 6, 7]. However, as we observed in Fig. 2, both unconditional and conditional colorization often fail to preserve the color edge, resulting in generating color-bleeding artifacts along the boundary regions.

2.2. Edge-Aware Colorization

Classical approaches [14, 15] address the bleeding artifacts in an optimization problem. Huang *et al*. [14] develop an edge detection algorithm for improving edges in-

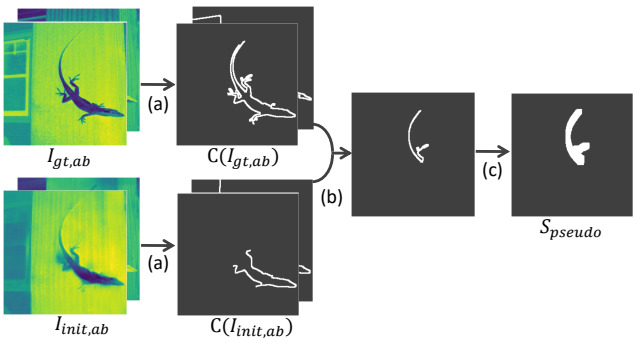

Figure 3: An overview of the pseudo-scribble generation. Two channels of $I_{\text{gt},ab}$ and $I_{\text{init},ab}$ represent an a and b channel, respectively.

formation during colorization. Yin *et al*. [15] propose a sharpening filter applied over a colorized image, alleviating the bleeding artifacts via optimization. Similarly, Zhao *et al*. [16] propose a joint bilateral filter which considers the adjacent color values for sharpening the edges. However, as these approaches mainly rely on the edges of input image, they still fail on the boundaries between the objects that share the similar gray-scale intensities. In contrast, our approach involves a direct interaction, which refines the boundary representation in any region regardless of its edge pixel values in the input image.

Recently, Su *et al*. [5] and Zhao *et al*. [13] leverage the semantic segmentation and object detection, respectively, when training a colorization model. These tasks enforce the network to learn the semantic objects, which may help to recognize the boundary of such objects to colorize. However, since both semantic segmentation and object detection recognize the objects defined by particular classes, the methods may still suffer from the color bleeding along the objects that are not classified by any categories. For example, in Fig. 5, the color-bleeding artifacts across the patterns inside the balloon would not be fully addressed by these methods, as such patterns are not classified into a certain class. In our approach, we leverage the scribbles that are independent of object types, allowing more general edge enhancement against these methods.

3. Proposed Method

3.1. Overall Workflow

This section provides a detailed description of our proposed method, as described in Fig. 4. As an interactive approach, we design an edge-enhancing network E to take scribbles, which annotates the color-bleeding edges, as additional inputs. These scribbles, which we term pseudo-scribbles, are automatically generated to approximate the real-world user hints (Section 3.2). The network E also takes the intermediate activation maps of the colorization network and refines them along the edges annotated by the pseudo-scribbles. Afterward, these refined representations pass through the following layers to obtain an edge-enhanced colorized output in the end (Section 3.3). To train our model, we introduce an edge-enhancing loss, which enforces the model to recover the clear edges close to those of the ground-truth image. Also, we propose both feature-regularization loss and consistency loss to prevent undesirable color distortion which debases the overall quality of edge-enhanced outputs (Section 3.4).

3.2. Pseudo-Scribble

Our approach requires a user-driven hint to be trained in an interactive manner, but collecting real-world user annotations needs a lot of time and human resources, which is prohibitive. Instead, we automatically generate the pseudo-scribbles S_{pseudo}, which emulate the real user scribbles, for each training image. The overall procedure of generating the S_{pseudo} is presented in Fig. 3. First, we obtain the color-bleeding outputs I_{init} from a pre-trained colorization model. Afterwards, we apply the Canny edge detector [18] $C(\cdot)$, a widely used edge detecting algorithm, onto the *ab* color channels of a ground-truth image I_{gt} and a I_{init}, respectively (Fig. 3 **(a)**). Then, we can obtain the binary maps $C(I_{\text{gt},ab})$ and $C(I_{\text{init},ab})$ which represent the edges. By selecting one of the edges that appears in $C(I_{\text{gt},ab})$, but not in $C(I_{\text{init},ab})$, we can have a single edge where the baseline fails to preserve boundary as clearly as I_{gt} does (Fig. 3 **(b)**). Note that while a gecko's tail is shown to be selected in **(b)**, other scribbles can be chosen in the training as well, such as its paw. Afterward, to better approximate a real user's hint, the S_{pseudo} are formed to be thick and coarse enough to 1) be easy to draw and 2) contain the bleeding boundary. To this end, we apply a width transformation $w(\cdot)$ that randomly modifies the width of the selected edge between 1 and 11 pixels (Fig. 3 **(c)**).

3.3. Edge-Enhancing Network

We apply an edge-enhancing network E to refine the intermediate representations of a colorization network, correcting the erroneously spread colors across the boundaries. This network encodes the corrective representations from both scribbles and the intermediate features as inputs and adds them to the original features with the residual connection. Suppose that we want to modify an activation map A_i, where $A = (A_1, A_2, ...A_l)$ is the set of intermediate activation maps from l different encoder layers of the colorization model (Fig. 4 (a)). To obtain the scribbles, we generate a S_{pseudo} by the procedure described in Section 3.2 and downscale it to match the spatial resolution of the activation map A_i from the i-th layer (Fig. 4 (b)). Then, S_{pseudo} and A_i are concatenated to provide an input for edge-enhancing network E. Given the concatenated tensor, we can obtain an

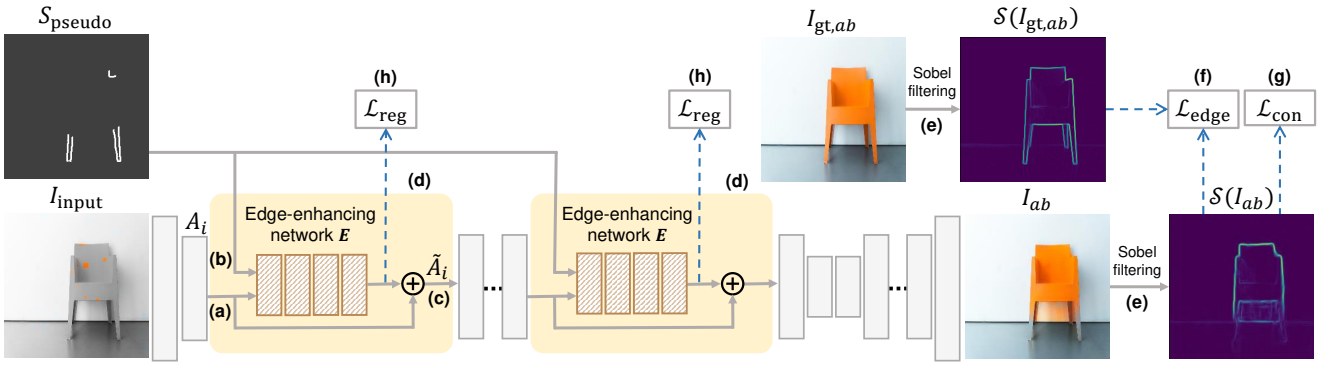

Figure 4: **An overview of our proposed method.** First, base colorization model (gray networks) colorize gray-scale image $\boldsymbol{I}_{\text{input}}$. After, multiple add-on edge-enhancing network $\boldsymbol{E}$ (yellow boxes) take the user-driven scribble $\boldsymbol{S}_{\text{pseudo}}$ and refine the corresponding activation maps from the base model. We apply $\boldsymbol{L}_{\text{edge}}$ between the edges of $\boldsymbol{I}_{ab}$ and $\boldsymbol{I}_{\text{gt},ab}$.

activation map representing the correction of A_i to alleviate color-bleeding artifacts. Therefore, by applying a residual connection with A_i, (Fig. 4 (c)) refined activation map $\tilde{A}_i$ is calculated as

$$\tilde{A}_i = A_i + E\left([S_{\text{pseudo}}, A_i]\right), \tag{1}$$

where E is the proposed edge-enhancing network and $[\cdot,\cdot]$ a concatenation.

We apply the edge-enhancing networks to the encoder because the edge-enhancing performance is empirically better when applying our network E to the encoder than the decoder. Detailed comparisons on the qualitative results of the network E applied in the encoder and decoder layer are provided in Section D of the supplementary material. To encourage edge refinement in both low- and high-level representations, we apply multiple edge-enhancing networks in both shallow and deep layers of the encoder (Fig. 4 (d)).

3.4. Objective Functions

Edge-Enhancing Loss. Inspired by the gradient difference loss (GDL) [26] which sharpens the video prediction, we propose an edge-enhancing loss $\mathcal{L}_{\text{edge}}$ for enhancing the edges in a target region. This loss $\mathcal{L}_{\text{edge}}$ enforces an edge-enhancing network E to generate the refined activation maps that enhance the edges of I_{ab} to be close to those of $I_{\text{gt},ab}$ (Fig. 4 (f)). To obtain the edge map, we utilize the Sobel filter, a differentiable edge extracting filter, onto the *CIE ab* channels of images, obtaining both horizontal and vertical derivative approximations of color intensities (Fig. 4 (e)).

The resulting color gradient is formally written as

$$\mathcal{S}(I) = \sqrt{(G_x * I)^2 + (G_y * I)^2},$$
$$G_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}, \; G_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}, \tag{2}$$

where G_x and G_y are horizontal and vertical Sobel filters which convolve with the given image, and $\mathcal{S}$ returns the gradient magnitude of them. Our proposed edge-enhancing loss can be written as

$$\mathcal{L}_{\text{edge}} = \mathbb{E}_{x,y\in\mathbb{P}}\left[||S(x,y) - S_{\text{gt}}(x,y)||_2^2\right], \quad S = \mathcal{S}(I_{ab}),\ S_{\text{gt}} = \mathcal{S}(I_{\text{gt},ab}), \tag{3}$$

where $\mathbb{P}$ denotes a set of coordinates (x, y) whose values are non-zero in a binary mask M. M only activates a set of pixels within certain distance from the target edge, *i.e.*, S_{pseudo}.

Feature-Regularization Loss. We wish for our proposed method to improve the edges while maintaining the original performance of the colorization network. Therefore, we introduce a feature-regularization loss $\mathcal{L}_{\text{reg}}$ (Fig. 4 (h)) to the output of the edge-enhancing network E. This encourages our network to learn optimal edge enhancement while avoiding the excessive perturbations in the network activation maps. This loss is formulated as

$$\mathcal{L}_{\text{reg}_i} = \|E\left([S_{\text{pseudo}}, A_i]\right)\|_2^2, \tag{4}$$

where i denotes an index of a layer to be revised and $[\cdot,\cdot]$ a concatenation.

Consistency Loss. While our proposed network enhances the gradient of colors around the given edges, it can unintentionally induce color distortions in the undesirable regions, *i.e.*, outside of the target edges. Therefore, we design an additional constraint, named consistency loss $\mathcal{L}_{\text{con}}$ (Fig. 4 (g)), to prevent these unnecessary changes. This loss further optimizes our network E to learn the refinements only in the desired regions. $\mathcal{L}_{\text{con}}$ penalizes the unfavorable changes of our enhanced output I_{ab} from the initial colorized output $I_{\text{init},ab}$, only in the regions where we wish for the colors to remain. As the binary mask M mentioned above indicates the regions for the colors to be changed by edge enhancement, we apply this loss on the pixels whose values of M

Kernel Size	Method	ImageNet ctest [17]		COCO-Stuff [19]		Place205 [20]	
		LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑
K=7	CIC [3]	0.248	13.281	0.247	13.368	0.254	13.577
	DeOldify [12]	0.250	13.234	0.251	13.059	0.227	14.258
	Zhang *et al*. [6]	0.246	13.248	0.206	14.755	0.219	14.815
	+Ours	**0.217**	**13.919**	**0.192**	**15.037**	**0.211**	**15.104**
	Zhang *et al*. [6]*	0.208	14.966	0.158	17.456	0.171	17.530
	+Ours*	**0.177**	**16.041**	**0.143**	**17.953**	**0.161**	**17.906**
	Su *et al*. [5]*	0.185	16.393	0.187	15.971	0.194	17.032
	+Ours*	**0.177**	**16.507**	**0.176**	**16.188**	**0.187**	**17.098**
K=Full	CIC [3]	0.172	21.001	0.164	21.456	0.153	21.873
	DeOldify [12]	0.159	21.433	0.149	21.985	0.156	21.933
	Zhang *et al*. [6]	0.148	21.981	0.135	22.729	0.138	**22.846**
	+Ours	**0.147**	**22.026**	**0.134**	22.729	0.138	22.845
	Zhang *et al*. [6]*	0.086	27.202	0.080	27.681	0.087	27.697
	+Ours *	**0.085**	**27.559**	**0.078**	**27.955**	0.087	**27.935**
	Su *et al*. [5]*	0.091	26.211	0.089	26.050	0.090	27.414
	+Ours*	0.091	**26.291**	**0.088**	**26.233**	**0.089**	**27.486**

Table 1: Quantitative comparison with the baselines on 1,000 images in the ImageNet [17], COCO-Stuff [19] and Place205 [20] validation set. Quantitative results in the local region show that our method effectively enhances the images.

are zero. This is enabled by multiplying $1 - M$ on each channel. This loss can be formulated as

$$\mathcal{L}_{\text{con}} = \mathbb{E}_{x,y \notin \mathbb{P}} \left[\|S(x,y) - S_{\text{init}}(x,y)\|_2^2 \right], \quad S = \mathcal{S}(I_{ab}),\ S_{\text{init}} = \mathcal{S}(I_{\text{init},ab}), \tag{5}$$

where $\mathbb{P}$ denotes a set of coordinates (x, y) whose values are non-zero in a binary mask M.

In summary, the overall objective function for training the edge-enhancing network is defined as

$$\mathcal{L}_{\text{total}} = \lambda_{\text{edge}}\mathcal{L}_{\text{edge}} + \sum_{i=1}^{l} \lambda_{\text{reg}_i}\mathcal{L}_{\text{reg}_i} + \lambda_{\text{con}}\mathcal{L}_{\text{con}}, \tag{6}$$

where λ_{edge}, λ_{reg} and λ_{con} are hyperparameters, and l is the number of layers with the edge-enhancing network.

3.5. Implementation Details

In our experiments, we use the colorization networks introduced in Zhang *et al*. [6] and Su *et al*. [5] as our backbone colorization models. Zhang *et al*. first proposes an interactive colorization approach that takes color hints, achieving state-of-the-art performance over the existing conditional methods. Su *et al*. achieves superior performance in an unconditional setting by leveraging an object detection module for an instance-level colorization. We empirically confirm that applying our framework to the baselines taking explicit color hints results in better optimized edge-enhancing networks, compared to training on unconditional ones. Therefore, similar to Zhang *et al*., we re-implement Su *et al*. to take local color hints as additional inputs, which is not available in the original paper. Further details are provided in Section I of the supplementary material.

Method	Cluster Discrepancy Ratio↑		
	ImageNet ctest [17]	COCO-Stuff [19]	Place205 [20]
CIC [3]	0.383	0.401	0.381
DeOldify [12]	0.437	0.445	0.441
Zhang *et al*. [6]	0.385	0.391	0.377
+Ours	**0.502**	**0.521**	**0.473**
Zhang *et al*. [6]*	0.418	0.421	0.402
+Ours *	**0.543**	**0.547**	**0.508**
Su *et al*. [5]*	0.336	0.325	0.336
+Ours *	**0.394**	**0.398**	**0.371**

Table 2: Quantitative results using cluster discrepancy ratio measured within the kernel size of 7 along the edges. The score ranges from 0 to 1.

4. Experiments

Baselines. We compare our proposed model with various colorization methods, including both unconditional and conditional ones. Unconditional baselines include CIC [3], DeOldify [12], and Zhang *et al*. without color hints. For conditional baselines, we utilize Zhang *et al*. and Su *et al*. with local hints, as mentioned in Section 3.5.

Dataset. The experiments are conducted with dataset

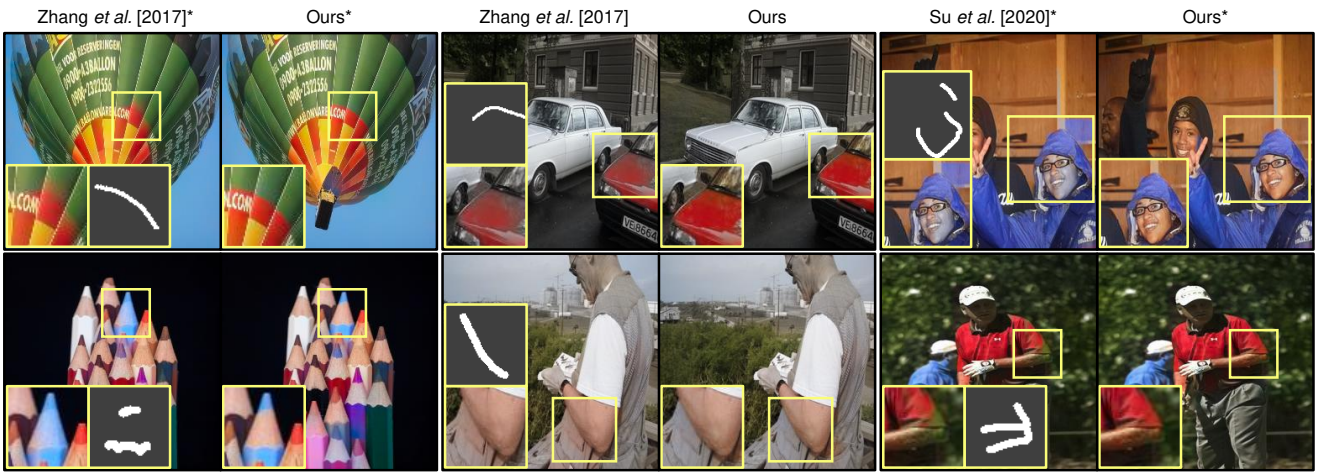

Figure 5: Qualitative examples of edge enhancement in gray-scale colorization. For each box, the left contains an original colorized image with artifacts, its magnified view, and given user scribble.

including *ImageNet* [17], *COCO-Stuff* [19] and *Place205* [20], which are generally used in colorization tasks.

Evaluation Measure. For evaluation, we assess the performance of our proposed method and other prior models by utilizing two measures, peak signal-to-ratio (PSNR) and learned perceptual image patch similarity (LPIPS) [27]. In addition, we newly propose a metric named *cluster discrepancy ratio* (CDR), which is designed to measure the degree of color-bleeding effects.

4.1. Cluster Discrepancy Ratio

Although PSNR and LPIPS are generally used for evaluating the colorization performance in a rich literature [3, 6, 5, 4, 2, 9, 11, 16], they are essentially based on the color difference between a generated image and a ground-truth one. Therefore, a colorized image that contains well-preserved edges but different colors from the ground-truth may be underrated by these two metrics. Note that this image would appear even more realistic compared to a bleeding image with similar colors. This specific failure case is described with an example and its scores in Section C of the supplementary material.

To compensate for this concern, we propose a novel evaluation metric that measures the discrepancy of color clusters grouped by the super-pixels defined by a simple linear iterative clustering method [28]. The super-pixels have their cluster assignments C based on color similarity. Inspired by this, we can perform binary classification on whether two adjacent pixels with different cluster assignments in the ground-truth still have different color values in the colorized outputs, especially along the edges. Specifically, for the pixel x^i_{gt} in a set of coordinates along the boundary E of the ground-truth $I_{\text{gt},ab}$, we identify whether the adjacent pixel x^j_{gt} within kernel size have different cluster assignments from that of x^i_{gt}. For those who have different cluster assignments from $C_{x^i_{\text{gt}}}$, the cluster index of x^i_{gt}, we define a set $\Omega(i)$ that consists of their coordinates. Then, in the generated outputs I_{ab}, we count the number of adjacent pixels x^j that 1) belong to $\Omega(i)$ and 2) have the same cluster assignment as C_{x^i}. The number indicates how many pixels are from different clusters in the $I_{\text{gt},ab}$, but share the same colors in the I_{ab}, which corresponds to the color-bleeding artifacts. Therefore, a super-pixel-based cluster discrepancy ratio can be written as

$$R\left(I_0, I_{\text{gt}}\right) = \frac{1}{|E|}\sum_{i\in E}\left(1 - \frac{1}{|\Omega(i)|}\sum_{k\in\Omega(i)} \mathbb{1}_{C_{x^i}=C_{x^{i+k}}}\right),$$
$$\Omega(i) = \left\{j : C_{x^i_{\text{gt}}} \neq C_{x^{i+j}_{\text{gt}}}, j \in S\right\}, \tag{7}$$

where E denotes a set of coordinates for the pixels of edges, $C_{x_{\text{gt}}}$ and C_x a cluster assignment given to the super-pixels of the $I_{\text{gt},ab}$ and I_{ab}. All possible shifts S within the kernel size K are described as

$$S \coloneqq \left[-\left\lfloor\frac{K}{2}\right\rfloor, ..., \left\lfloor\frac{K}{2}\right\rfloor\right] \times \left[-\left\lfloor\frac{K}{2}\right\rfloor, ..., \left\lfloor\frac{K}{2}\right\rfloor\right]. \tag{8}$$

4.2. Qualitative Results

In Fig. 1, we visualize the images having color-bleeding artifacts from the conditional colorization model of Zhang *et al.* [6] and their enhanced results using our proposed method. This demonstrates that our approach robustly corrects the bleeding boundaries even when roughly drawn scribbles are given. In addition, multiple edges can be enhanced in a single feed-forward when their corresponding scribbles are given at once. Fig. 5 provides additional qualitative examples of edge enhancement in our approach applied to both the conditional and unconditional model of

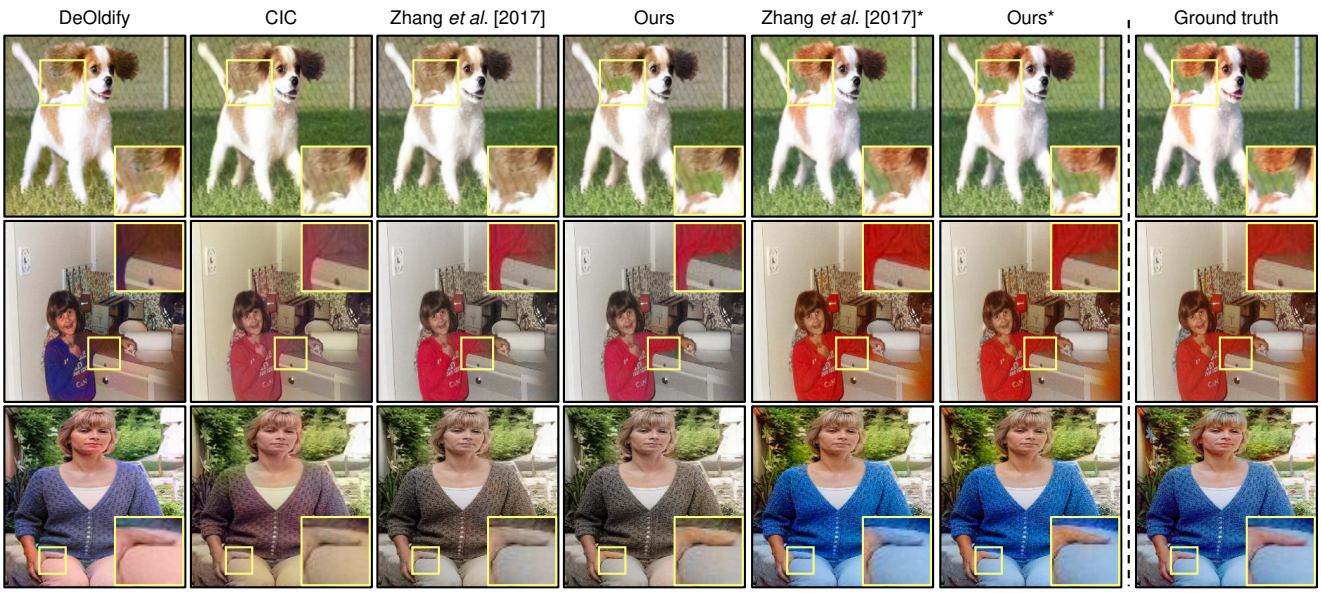

Figure 6: Qualitative comparisons between ours and baseline models. The results from the baselines include the color-bleeding artifacts in the same region, while our method successfully improves such bleedings and makes the results realistic.

Zhang *et al*., and conditional of Su *et al*. [5] In Fig. 6, colorized images have the color-bleeding region for the baseline models. Comparing with other approaches, our method refines the coarse region. We present additional qualitative results of our approach with these two baselines in Figs.8 and 9 of the supplementary material.

4.3. Quantitative Comparison

As the improvement in the colorization outputs induced by our approach arises in a local region along the edges, we present the results of PSNR and LPIPS measured in these particular locations, as well as in a global region. To conduct a local evaluation, we randomly sample $1,000$ S_{pseudo} from the test samples of each dataset, which contain the regions of color-bleeding, as explained in Section 3.2. Afterward, we report the local scores measured within the kernel size of 7 along the edge pixels of those S_{pseudo}. For global evaluation, we report the scores calculated on the entire region, specifying a kernel size as *Full*. To accommodate the scribbles similar to the real hints in our evaluation, we provide S_{pseudo} with random widths ranging from 1 to 5 pixels for edge enhancement. Table 1 shows that our model effectively advances the colorization performance of two baselines, outperforming the existing methods. As bleeding effects are mostly mitigated in the local regions, large improvements are observed when the evaluation regions are localized along the edges. In addition, Table 2 presents the CDR measured for both our method and the baselines. Our method achieves a higher score against other baselines, demonstrating its superiority in colorizing the adjacent instances with different colors.

In Table 3, we ablate $\mathcal{L}_{\text{con}}$, $\mathcal{L}_{\text{reg}}$, and width augmentation technique to analyze their effectiveness quantitatively. When we ablate $\mathcal{L}_{\text{con}}$, overall performance on PSNR, LPIPS, and CDR is slightly degraded, which is mainly due to the color distortion in the wrong regions. Our method without $\mathcal{L}_{\text{reg}}$ results in degraded scores of PSNR and LPIPS while obtaining the best score in CDR. Since $\mathcal{L}_{\text{reg}}$ suppresses the excessive perturbations in the refined feature maps of our edge-enhancing network, removing this loss causes an excessive edge enhancement (*e.g.*, saturated colors along the edge) as well as undesirable color distortions in the entire region. As we ablate the width augmentation for the S_{pseudo} in the training, our approach achieves the best score in CDR, while PSNR and LPIPS score become even worse than Zhang *et al*. This implies that our edge-enhancing network tends to excessively enhance the color boundaries when unseen thick scribble is provided in the test time, ruining the overall colorization quality. Therefore, width augmentation plays a critical role in learning a robust color enhancing, invariant to the width of a given scribble. We support this claim with the qualitative results of the models with these objective functions ablated in Section B of the supplementary material. In summary, our proposed method with every proposed objective function and augmented S_{pseudo} achieves an optimal performance of edge enhancement.

4.4. Verification for Labor-Efficient Interaction

We believe that our approach provides fast and reliable interactions, which help users to correct the color bleeding in an intuitive manner. To verify its usefulness with regard to labor efficiency, we conduct a user study with 13 novice participants using a user interface that we provide. Each

Figure 7: Qualitative results of edge enhancement in sketch colorization. We trained and evaluated our method on two datasets, *Yumi's Cells* [21] (first row) and *Danbooru* [22] (second row). For each row, the columns denoted as "Zhang *et al.* " include the colorized outputs with color-bleeding artifacts, and the columns of "ours" represent the edge-enhanced results.

participant is given five randomly selected colorized images that contain the color-bleeding artifacts. They are instructed to enhance the images by identifying color-bleeding areas and drawing scribbles until they obtain satisfactory results. On average, finding color-bleeding areas and drawing scribbles take 13.20 ± 8.46 and 4.41 ± 3.08 seconds per image, and each participant draws 2.9 ± 1.2 scribbles to enhance an image. The resultant improvement of PSNR is from 23.7 to 26.2 and LPIPS from 0.14 to 0.07, indicating that participants provide meaningful edges for enhancement. These results demonstrate that our method have potentials to be applied in practical applications. We provide additional analysis on the robustness of our method across different users in Section A of the supplementary material.

5. Experiments on Sketch Colorization

In this section, we further explore the potentials of our method in sketch colorization as well. Compared to the gray-scale image, the sketch image contains a set of thin lines that explicitly define the semantic boundaries between objects. However, as shown in the Fig. 7, color-bleeding artifacts across these lines are easily observed, which indicates that the model still fails to preserve the boundary even when they contain edges in the input image. The qualitative results in the columns denoted as "ours" in Fig. 7 demonstrate that our method performs robust edge preservation in the sketch colorization as well. We train and evaluate our method and the baselines on comic domain dataset including *Yumi's Cells* [21] and *Danbooru* [22]. The implementation details about sketch colorization are described in Section I of the supplementary material. To further demonstrate the effectiveness of our method on sketch colorization, we present quantitative and qualitative results in Section H of the supplementary material.

Method	ImageNet ctest [17]		
	LPIPS↓	PSNR↑	CDR↑
Zhang *et al.* [6]*	0.208	14.966	0.418
+Ours w/o $\mathcal{L}_{\text{con}}$ *	0.183	15.799	0.472
+Ours w/o $\mathcal{L}_{\text{reg}}$ *	0.185	15.624	0.605
+Ours w/o aug *	0.259	13.372	**0.647**
+Ours (Full) *	**0.177**	**16.041**	0.512

Table 3: Ablation study on the proposed modules. All results are measured within kernel size of 7 along with the single scribble.

6. Conclusion

In this paper, we propose a novel and simple approach to effectively alleviate the color-bleeding artifacts which significantly degrades the quality of colorized outputs. Our method improves the bleeding edges by refining the intermediate features of the colorization network in the desired regions via user-interactive scribbles as additional inputs. Extensive experiments demonstrate its outstanding performance and reasonable labor efficiency, manifesting its potentials in practical applications.

Acknowledgements This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2019-0-00075, Artificial Intelligence Graduate School Program(KAIST)). This work was also supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2021-0-01778, Development of human image synthesis and discrimination technology below the perceptual threshold) We thank all researchers at NAVER WEBTOON Corp.

References

[1] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *Proc. the ACM Transactions on Graphics (ToG)*, 35, 2016. 1, 2

[2] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Proc. of the European Conference on Computer Vision (ECCV)*, 2016. 1, 2, 6

[3] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proc. of the European Conference on Computer Vision (ECCV)*, 2016. 1, 2, 5, 6

[4] Seungjoo Yoo, Hyojin Bahng, Sunghyo Chung, Junsoo Lee, Jaehyuk Chang, and Jaegul Choo. Coloring with limited data: Few-shot colorization via memory augmented networks. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2019. 1, 2, 6

[5] Jheng-Wei Su, Hung-Kuo Chu, and Jia-Bin Huang. Instance-aware image colorization. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2020. 1, 2, 3, 5, 6, 7

[6] Richard Zhang, Jun-Yan Zhu, Phillip Isola, Xinyang Geng, Angela S. Lin, Tianhe Yu, and Alexei A. Efros. Real-time user-guided image colorization with learned deep priors. *Proc. the ACM Transactions on Graphics (ToG)*, 36, 2017. 1, 2, 5, 6, 8

[7] Lvmin Zhang, Chengze Li, Tien-Tsin Wong, Yi Ji, and Chunping Liu. Two-stage sketch colorization. *Proc. the ACM Transactions on Graphics (ToG)*, 37:261:1–261:14, 2018. 1, 2

[8] Mingming He, Dongdong Chen, Jing Liao, Pedro V Sander, and Lu Yuan. Deep exemplar-based colorization. *Proc. the ACM Transactions on Graphics (ToG)*, 37:47, 2018. 1, 2

[9] Bo Zhang, Mingming He, Jing Liao, Pedro V Sander, Lu Yuan, Amine Bermak, and Dong Chen. Deep exemplar-based video colorization. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2019. 1, 2, 6

[10] Y. Xiao, P. Zhou, Y. Zheng, and C. Leung. Interactive deep colorization using simultaneous global and local inputs. In *The IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1887–1891, 2019. 1

[11] J. Lee, E. Kim, Y. Lee, D. Kim, J. Chang, and J. Choo. Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2020. 1, 2, 6

[12] Jason Antic. deoldify. https://github.com/jantic/DeOldify, 2020. [Online; accessed 07-11-2020]. 2, 5

[13] Jiaojiao Zhao, Jungong Han, Ling Shao, and Cees Snoek. Pixelated semantic colorization. *International Journal of Computer Vision*, 128, 04 2020. 2, 3

[14] Yi-Chin Huang, Yi-Shin Tung, Jun-Cheng Chen, Sung-Wen Wang, and Ja-Ling Wu. An adaptive edge detection based colorization algorithm and its applications. In *Proc. of the ACM International Conference on Multimedia*, 2005. 2

[15] Hui Yin, Yuanhao Gong, and Guoping Qiu. Side window filtering. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2019. 2, 3

[16] Jiaojiao Zhao, L. Liu, Cees G. M. Snoek, J. Han, and L. Shao. Pixel-level semantics guided image colorization. *ArXiv*, abs/1808.01597, 2018. 2, 3, 6

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2009. 2, 5, 6, 8

[18] J. Canny. A computational approach to edge detection. *The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPMAI)*, 8:679–698, 1986. 2, 3

[19] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2018. 2, 5, 6

[20] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPMAI)*, 40:1452–1464, 2018. 2, 5, 6

[21] NaverWebtoon. Yumi's cells. https://comic.naver.com/webtoon/list.nhn?titleId=651673, 2019. [Online; accessed 22-11-2019]. 2, 8

[22] Aaron Gokaslan Gwern Branwen. Danbooru2017: A large-scale crowdsourced and tagged anime illustration dataset. https://www.gwern.net/Danbooru2017, 2018. [Online; accessed 22-03-2018]. 2, 8

[23] Man M. Ho, L. Zhang, Alexander Raake, and J. Zhou. Semantic-driven colorization. *ArXiv*, abs/2006.07587, 2020. 2

[24] Patricia Vitoria, Lara Raad, and Coloma Ballester. Chromagan: Adversarial picture colorization with semantic class distribution. In *The IEEE Winter Conference on Applications of Computer Vision*, 2020. 2

[25] Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. *Proc. the ACM Transactions on Graphics (ToG)*, 23:689–694, 2004. 2

[26] Michaël Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. In *Proc. the International Conference on Learning Representations (ICLR)*, 2016. 4

[27] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2018. 6

[28] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *The IEEE Transactions on Pattern*

Analysis and Machine Intelligence (TPMAI), 34:2274–2282, 2012. 6