Deep Edge-Aware Interactive Colorization against Color-Bleeding Effects

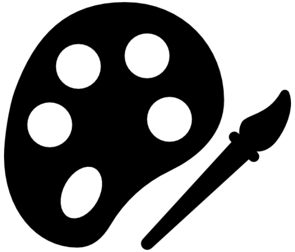
Eungyeup Kim*[1] Sanghyeon Lee*[1] Jeonghoon Park*[1] Somi Choi[1]
Choonghyeon Seo[2] Jaegul Choo[1]

KAIST AI[1] Naver Webtoon Corp.[2]

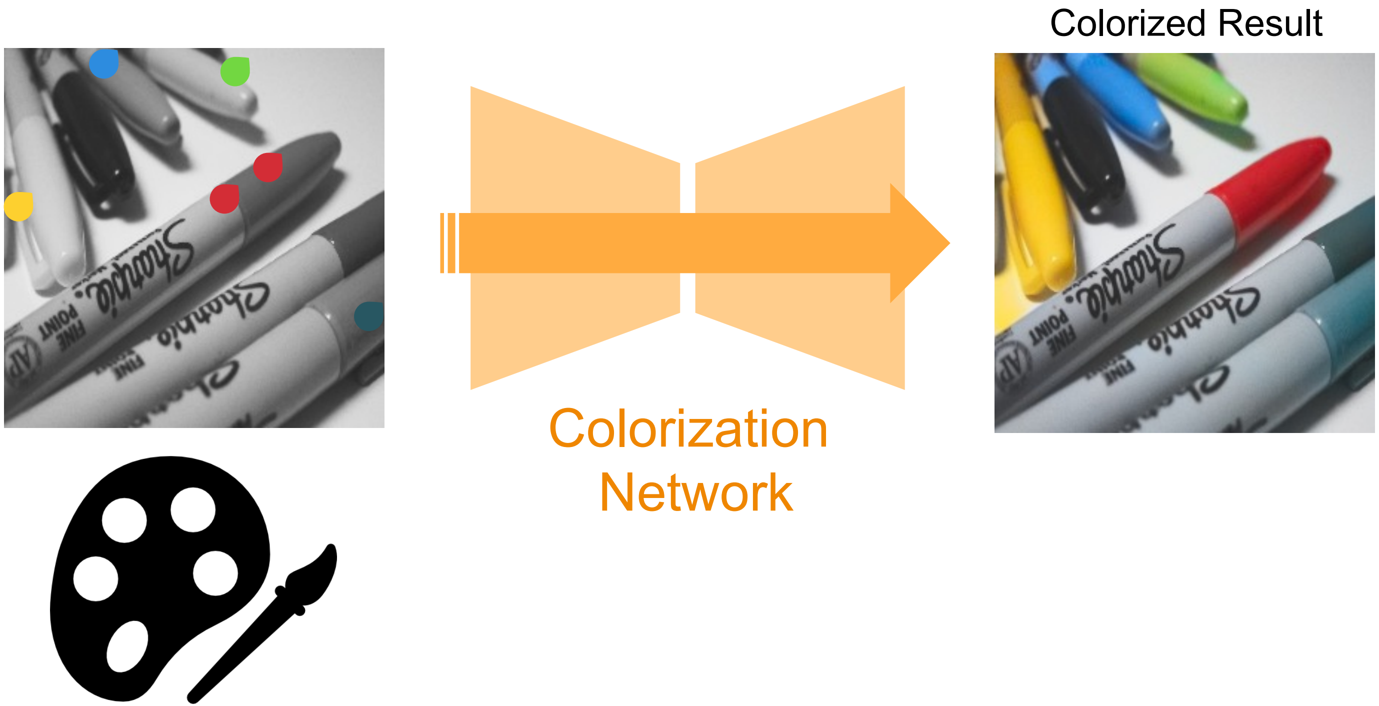
ICCV 2021 Oral Presentation

Deep Image Colorization : Overview

Deep Image Colorization : Overview



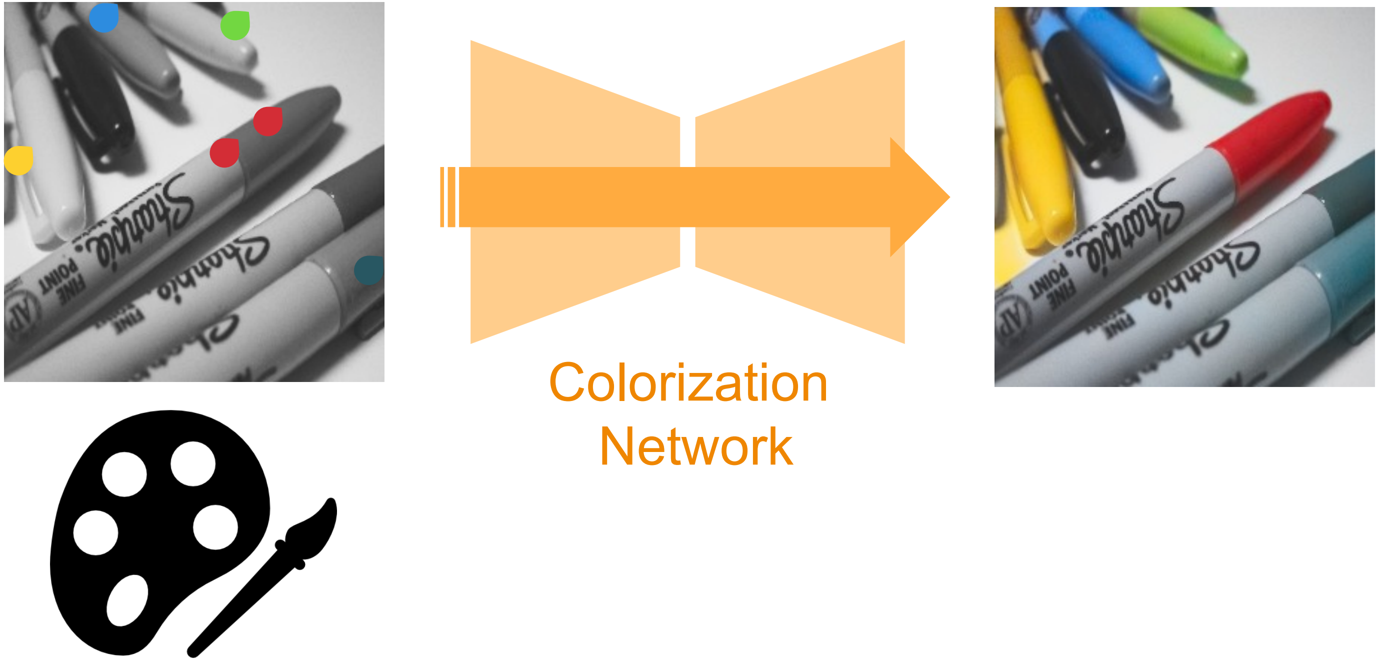
Deep Image Colorization : Overview



Plausible colorization output, but ***not perfect*** yet

Deep Image Colorization : Color-Bleeding Artifacts

Colorized Result

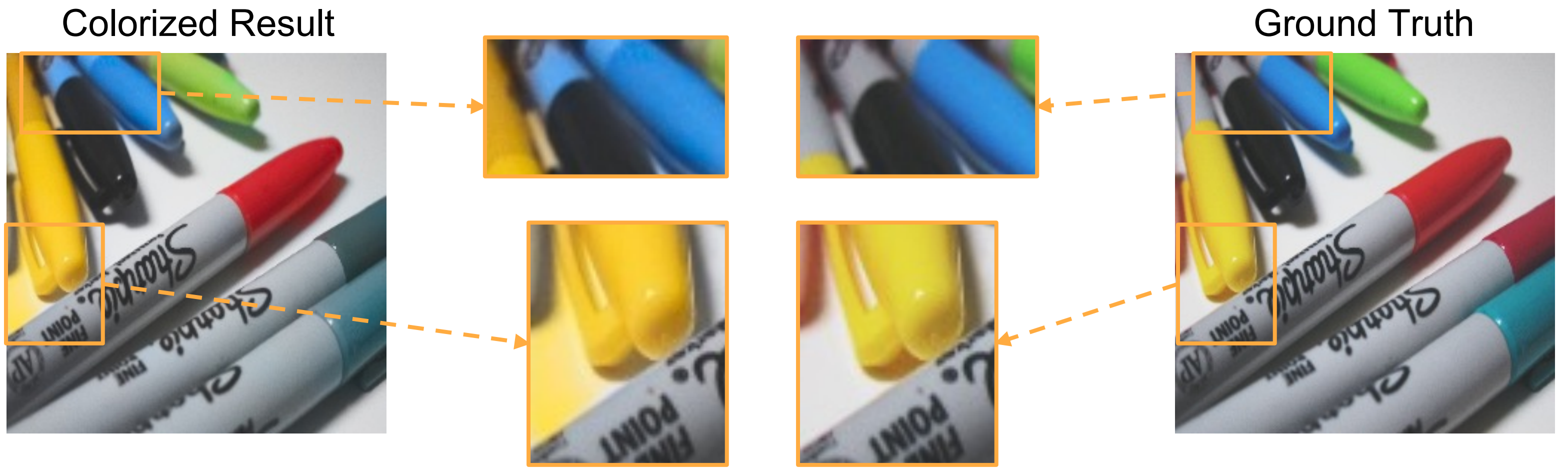


Deep Image Colorization : Color-Bleeding Artifacts

Colorized Result

Ground Truth

Deep Image Colorization : Color-Bleeding Artifacts



Incorrect color spreading across the
object boundaries makes the result unrealistic.

Color-Bleeding Artifacts in Existing Approaches

Zhang *et al.*, SIGGRAPH’17

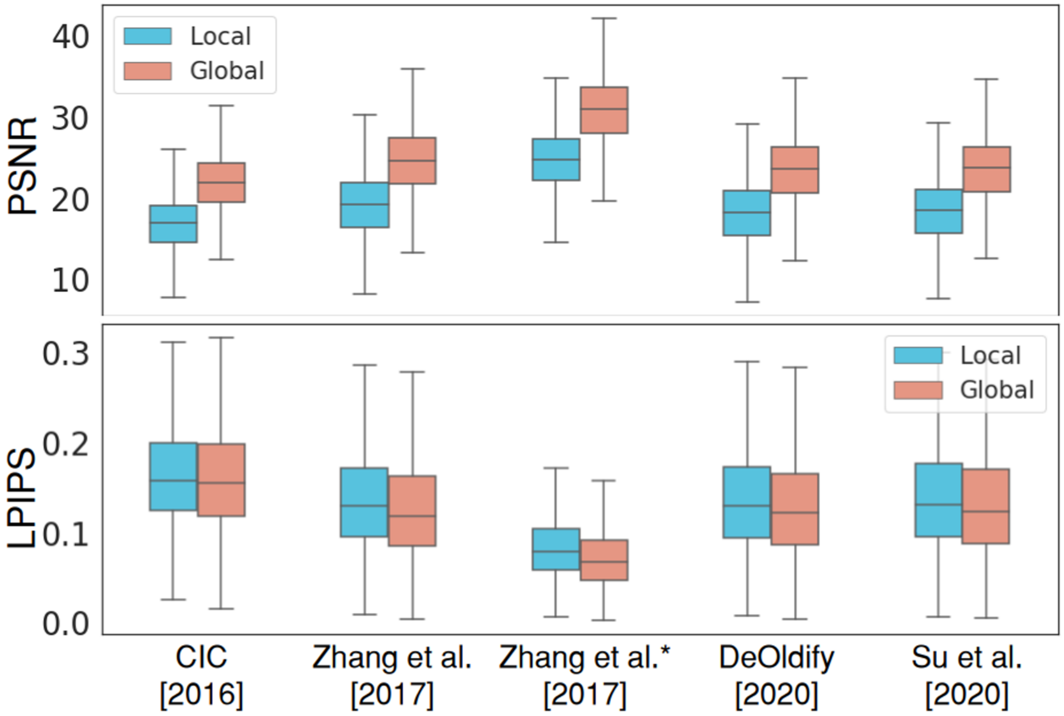
Su *et al.*, CVPR’20

Zhang *et al.*, ECCV’16

Lee *et al.*, CVPR’20

: color-bleeding regions

Color-Bleeding Artifacts in Existing Approaches



Colorization quality ***along edges*** are significantly ***lower*** than that of other regions.

Our Main Contribution

1. We propose a ***human-in-the-loop*** approach which can resolve the color-bleeding artifacts via a simple ***add-on module***, which refines the edge-relevant representations of the back-bone model.

2. With only a reasonable amount of user effort, our approach achieves the ***SOTA*** results when applied to widely used baselines in both gray-scale and sketch colorization.

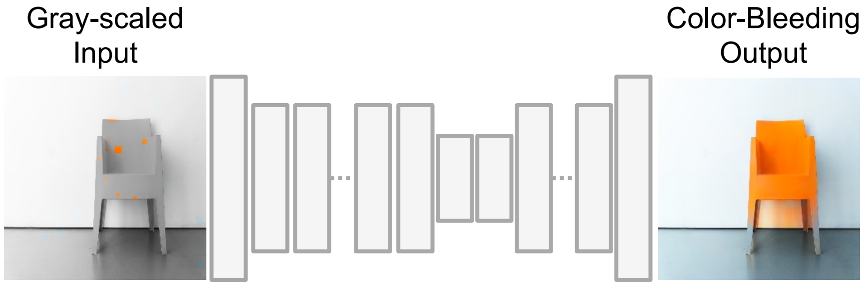
3. To compensate for the blind spot in PSNR and LPIPS, we propose a ***cluster discrepancy ratio*** which measures how precisely the model colorizes along the given edges.

Let’s See How Our Method Works!

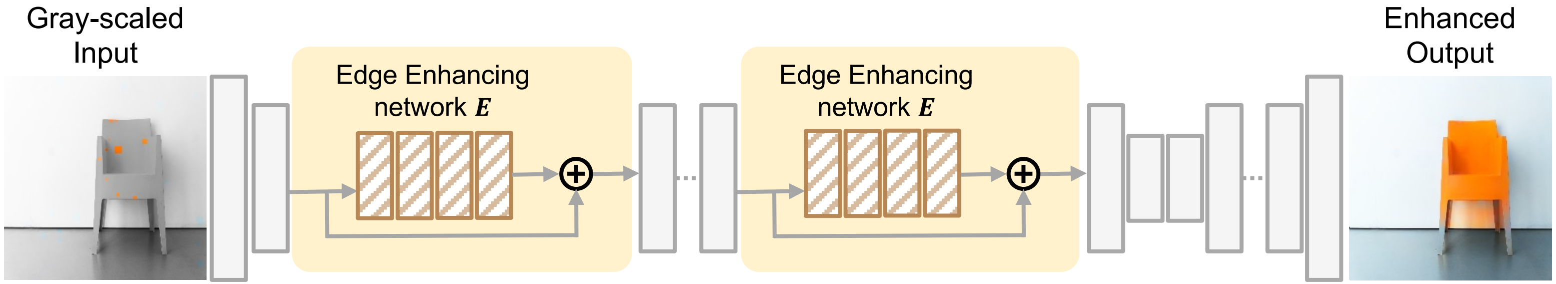
Colorized results

Edge-enhanced results

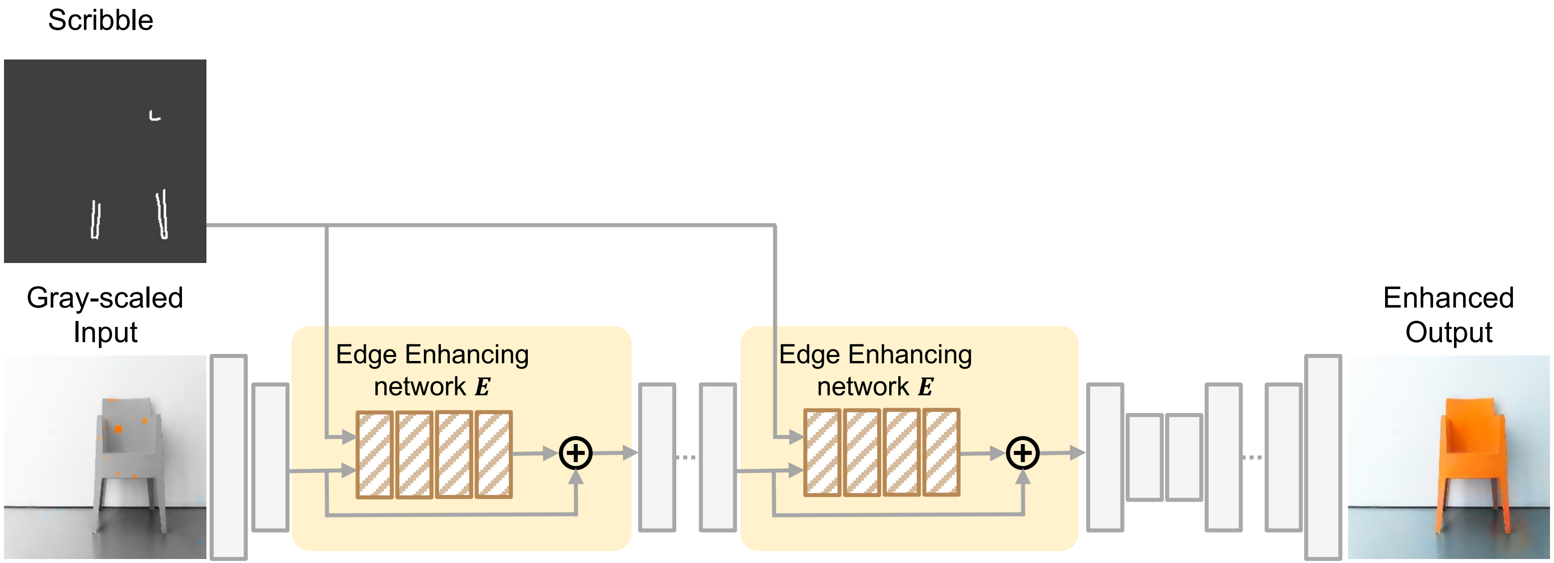
Proposed Method: Overall Workflow



Proposed Method: Overall Workflow

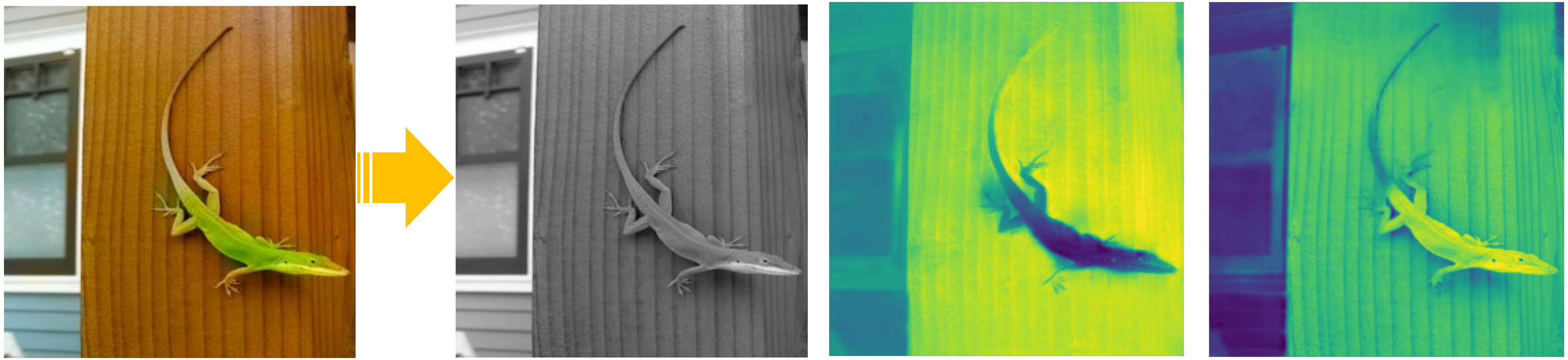


Proposed Method: Overall Workflow

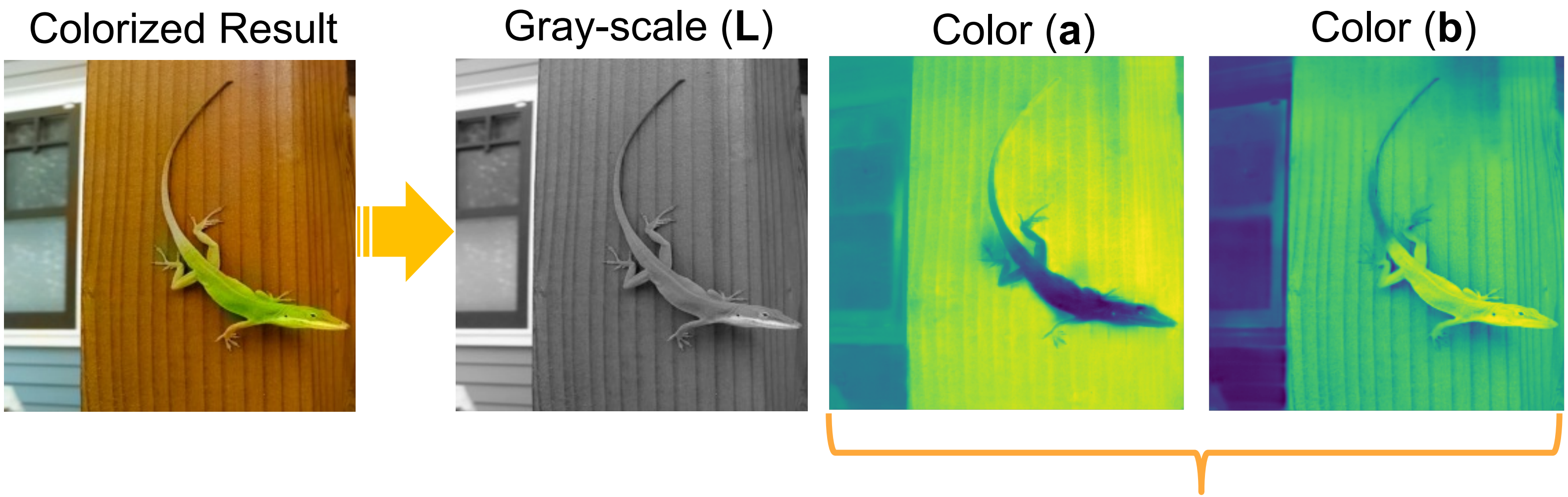


Generation of Pseudo Scribbles

Colorized Result



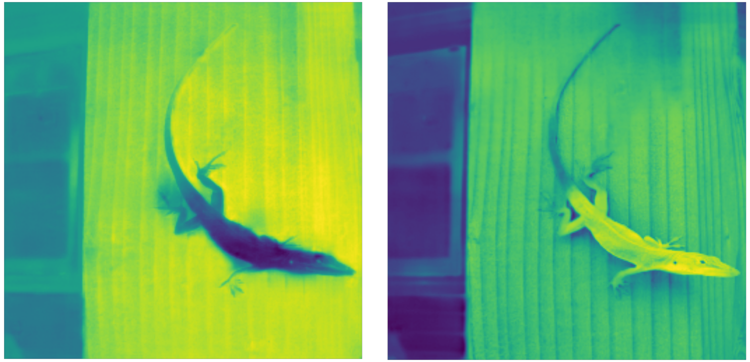
Generation of Pseudo Scribbles



Color space where the lizard colors are determined!

Generation of Pseudo Scribbles

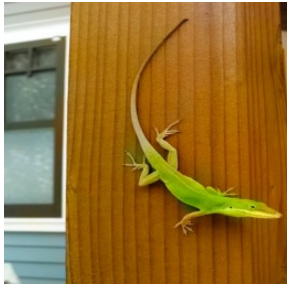
Colorized Result

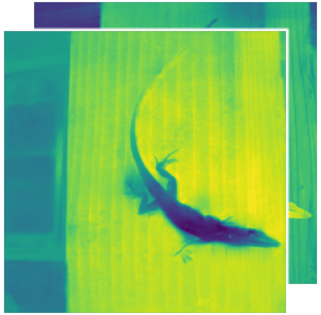


Color space where the lizard colors are determined!

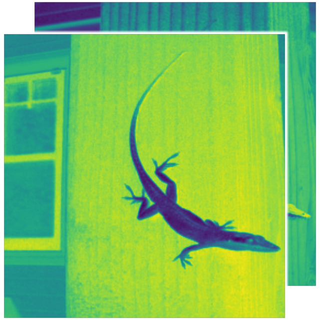
Generation of Pseudo Scribbles

Colorized Result

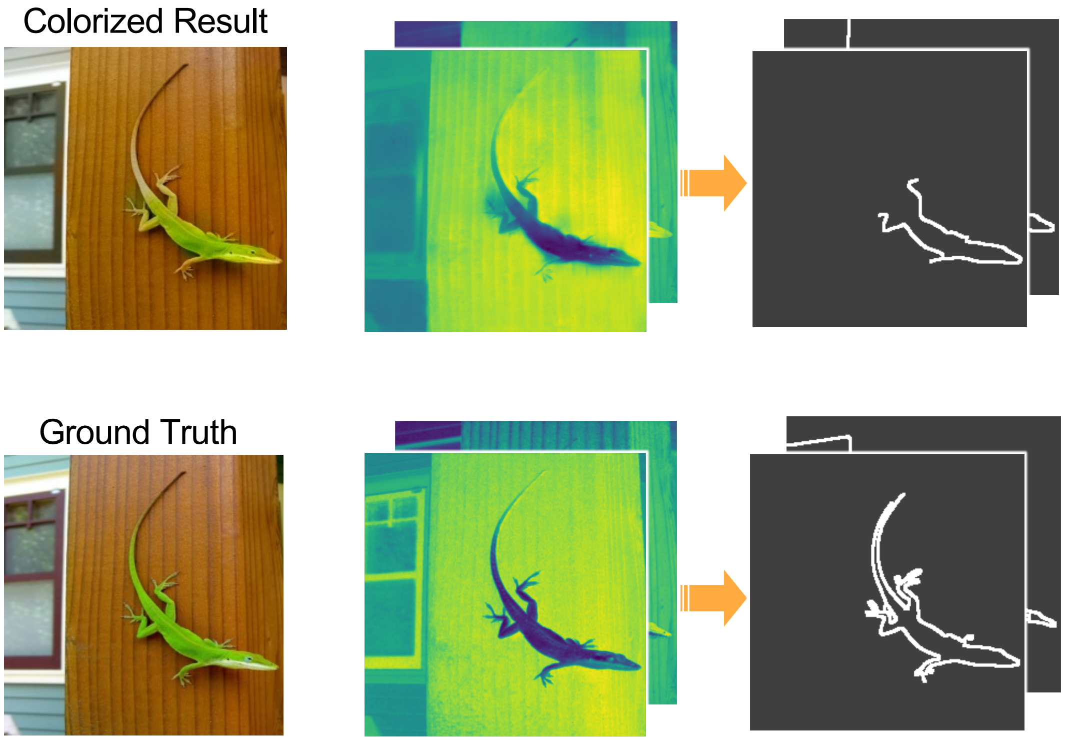




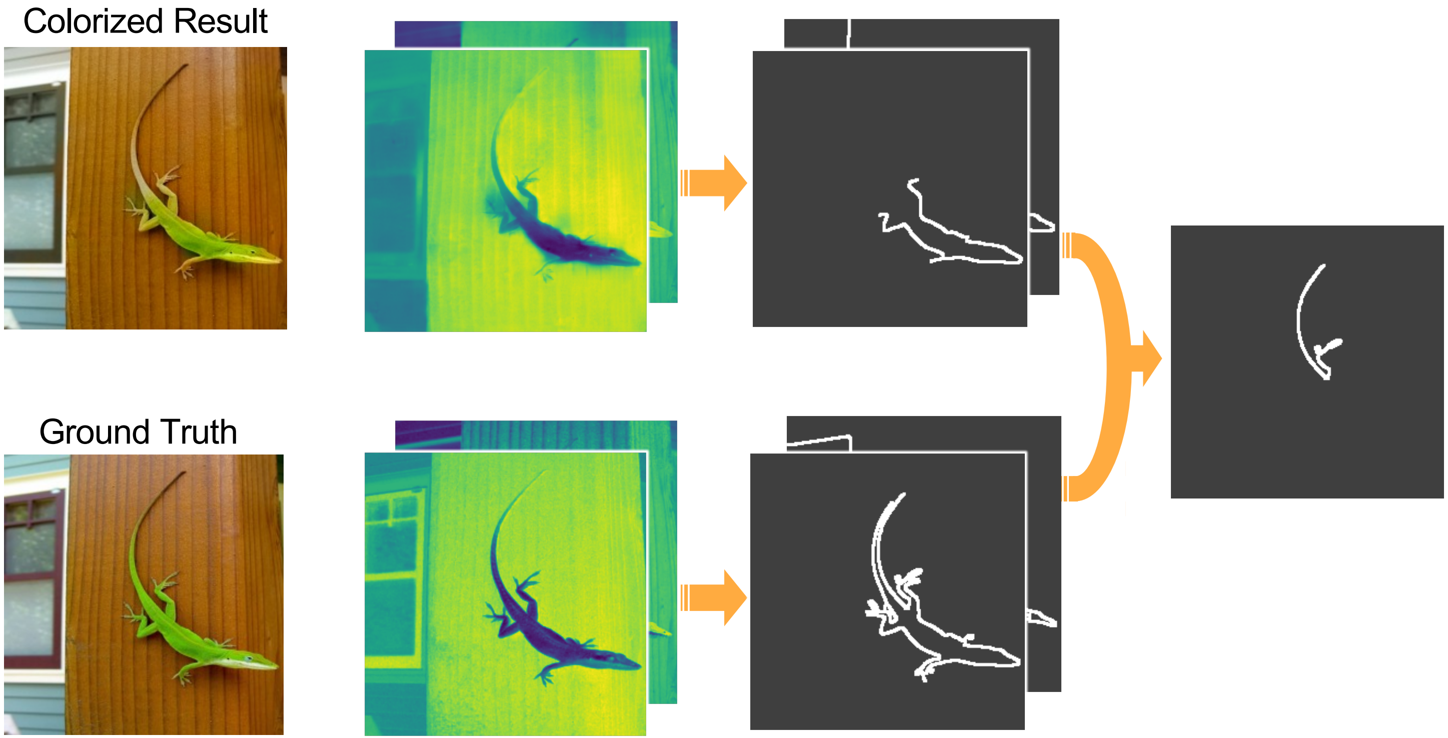
Ground Truth



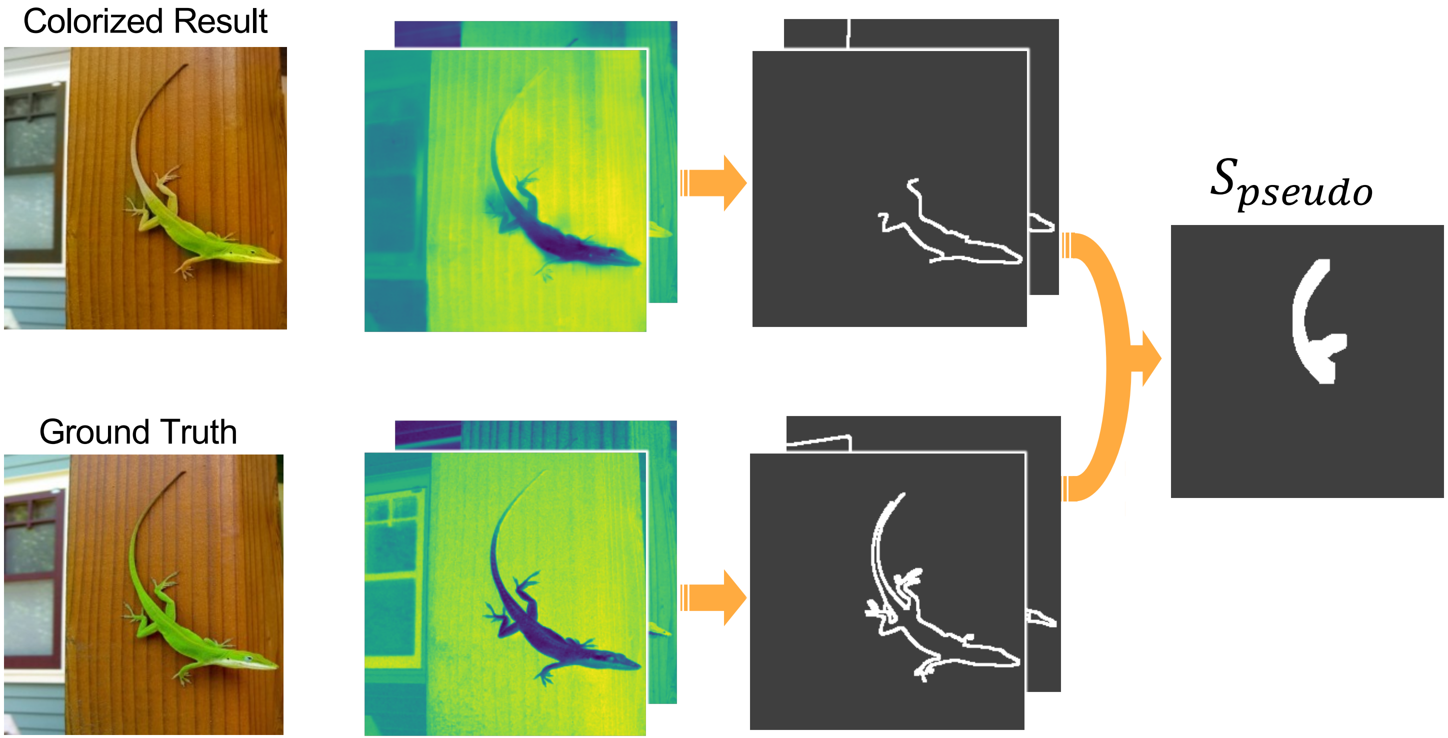
Generation of Pseudo Scribbles



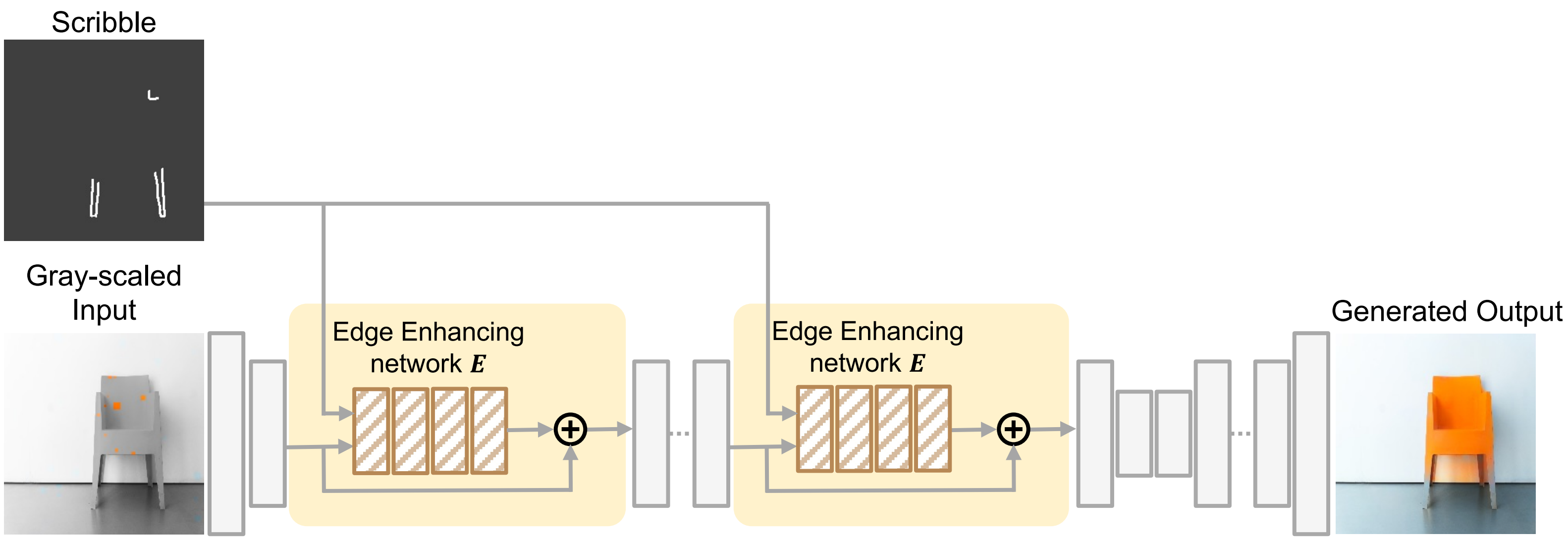
Generation of Pseudo Scribbles



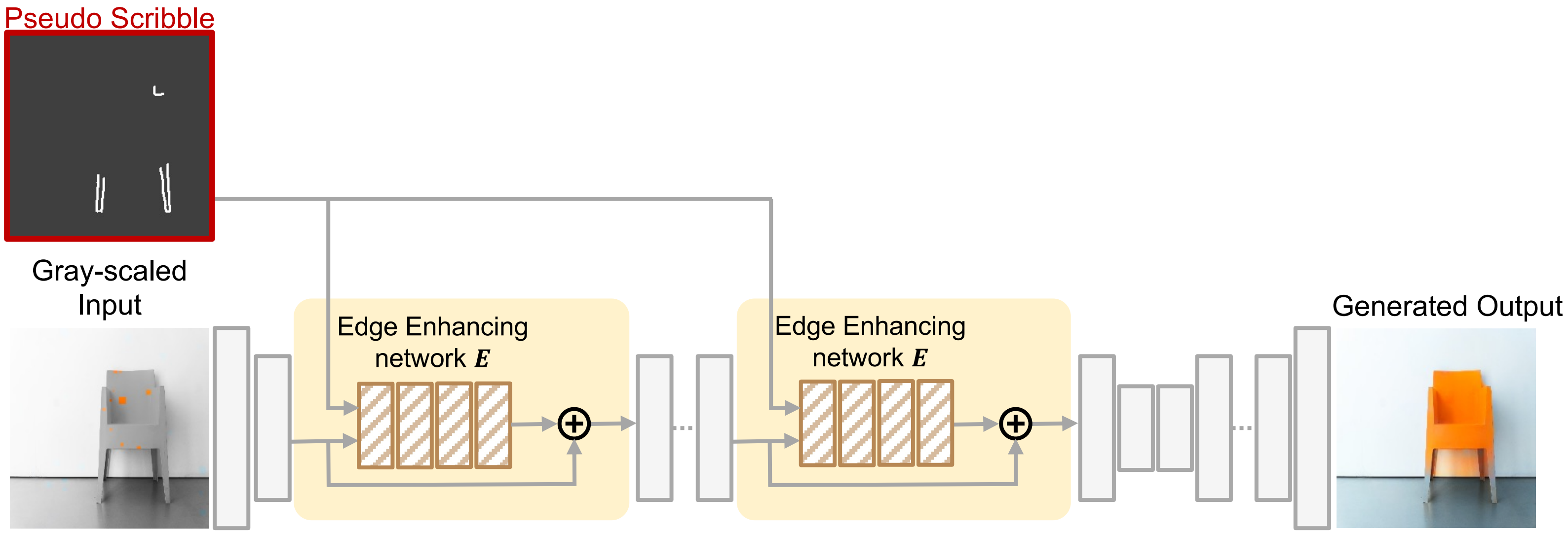
Generation of Pseudo Scribbles



Proposed Method: Overall Workflow

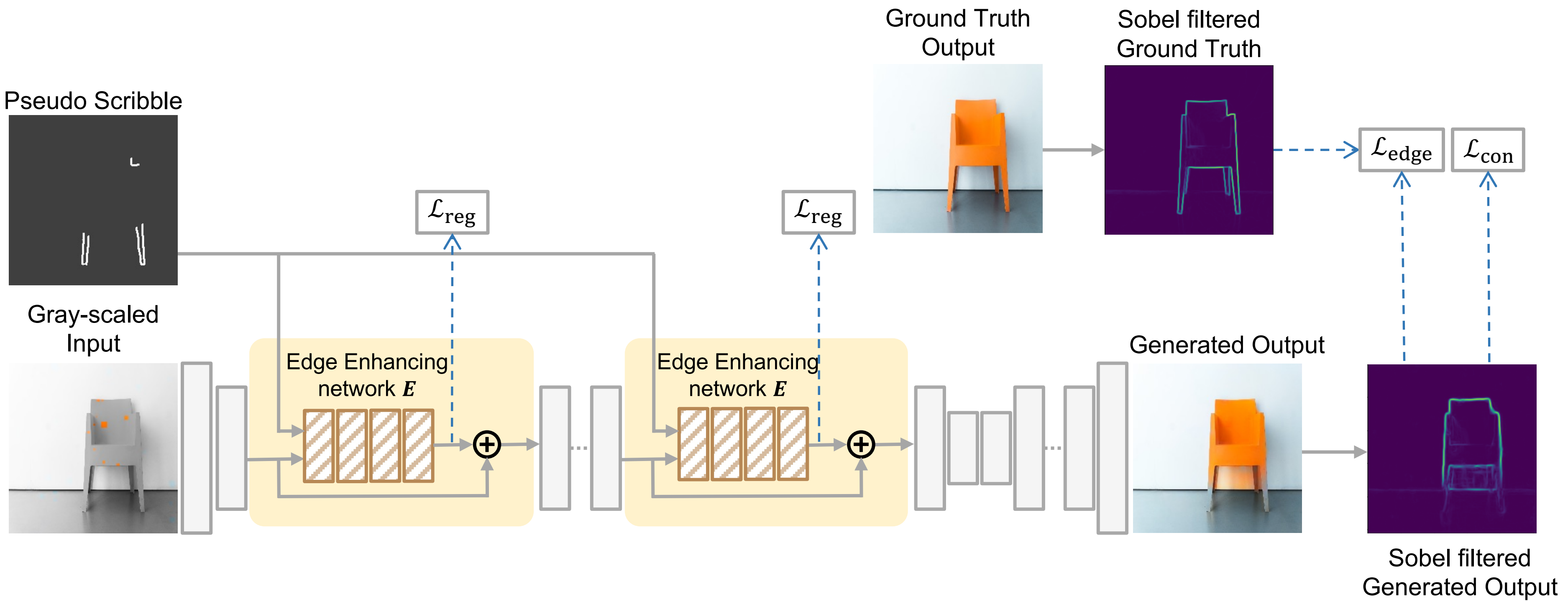


Proposed Method: Overall Workflow

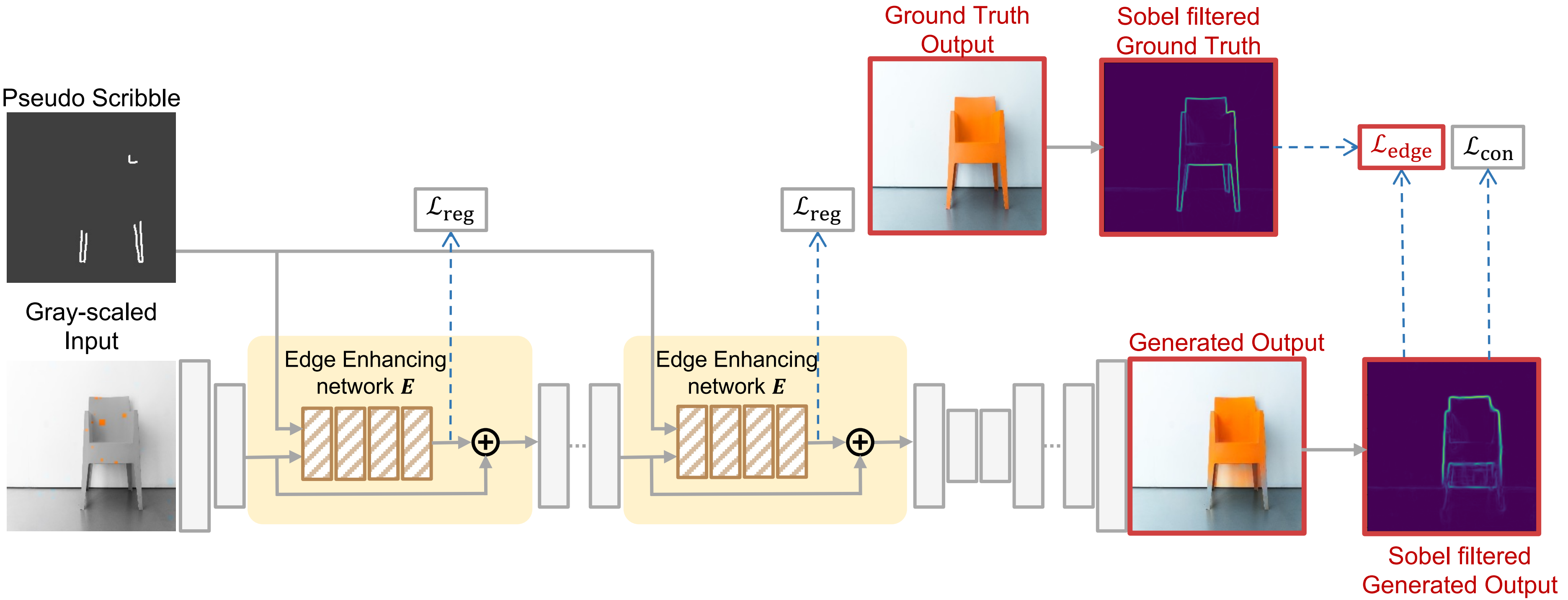


Pseudo scribbles are used during training
as ***approximation for user scribble***.

Proposed Method: Overall Workflow

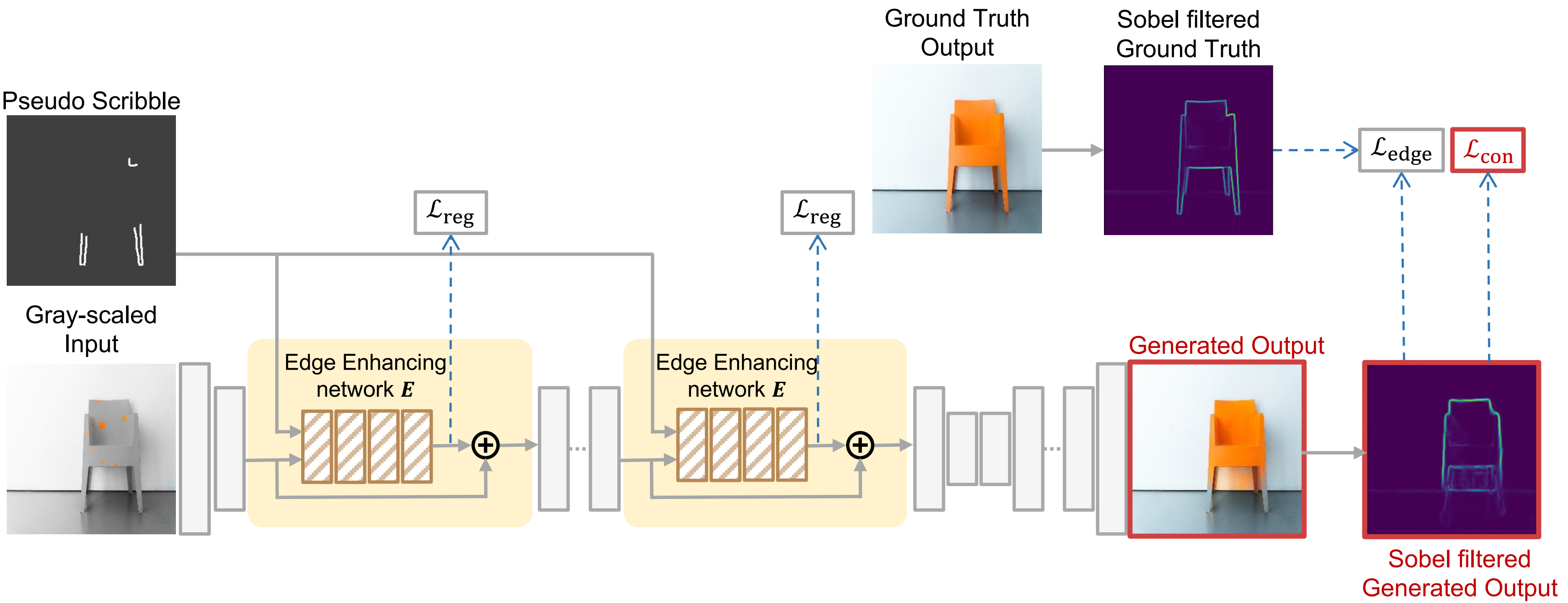


Proposed Method: Overall Workflow



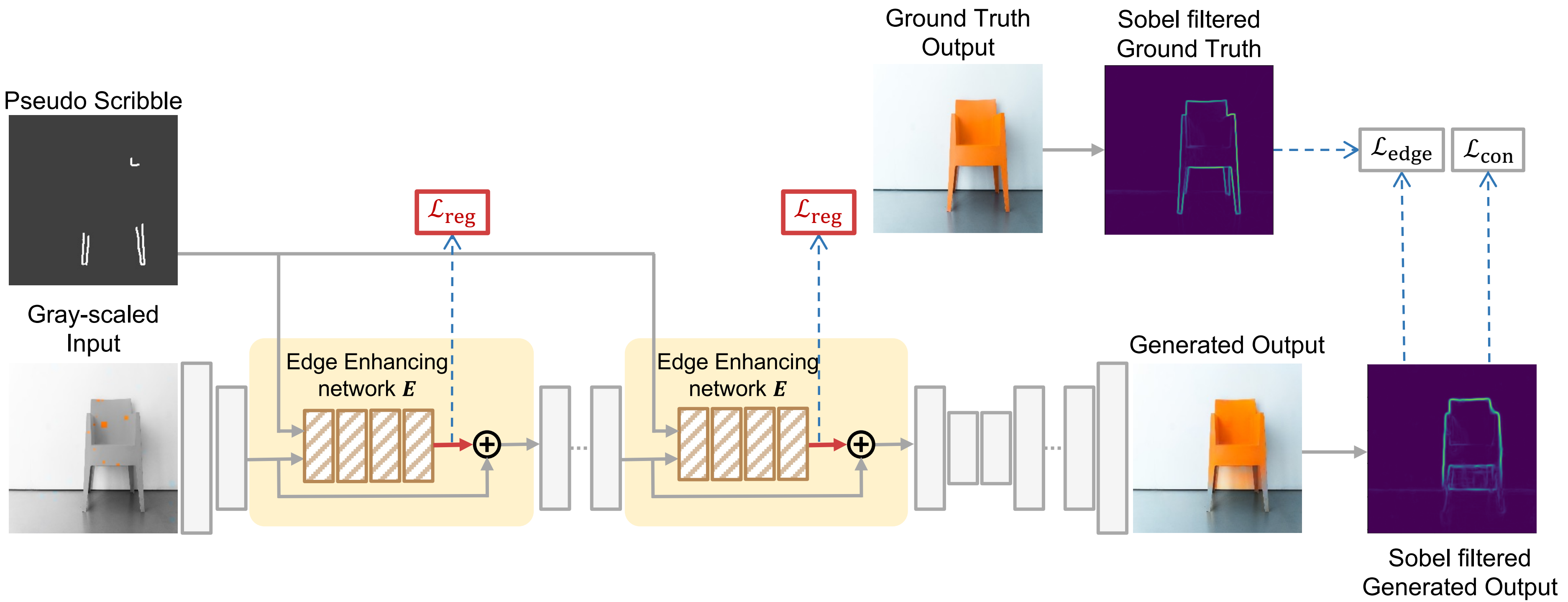
Edge enhancing network E is trained with ***1) edge-enhancing loss***

Proposed Method: Overall Workflow



Edge enhancing network E is trained with ***2) consistency loss***

Proposed Method: Overall Workflow



Edge enhancing network E is trained with ***3) feature-regularization loss***

Proposed Method: Edge-enhancing Loss

Initial Output

Enhanced Output

Edge-enhancing loss directly ***supervises learning of edge enhancement*** via sharp edge in ground truth.

Proposed Method: Edge-enhancing Loss

Ground Truth I_{gt}

Generated Output I

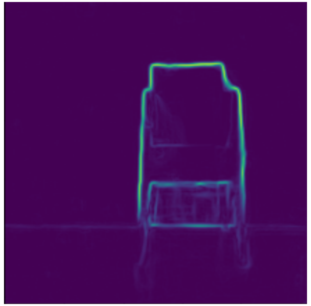
Proposed Method: Edge-enhancing Loss

Ground Truth I_{gt}

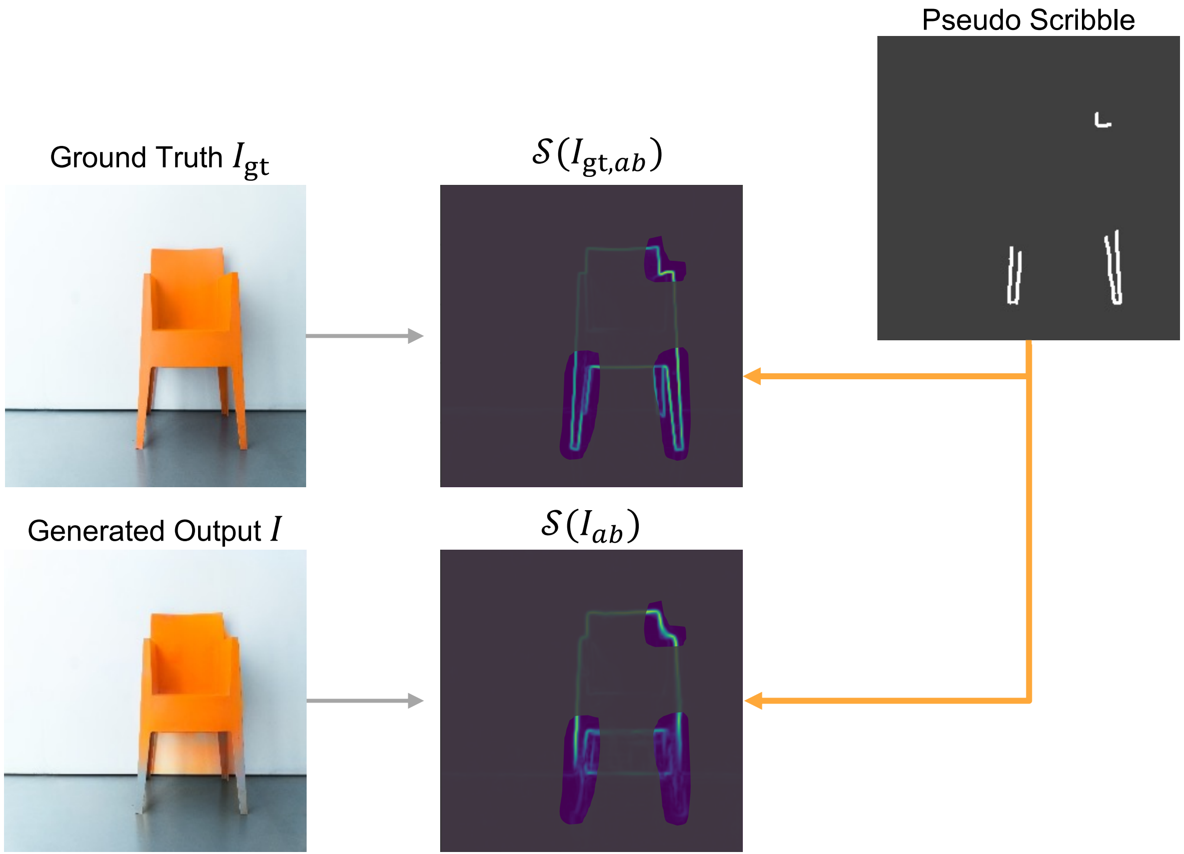
$\mathcal{S}(I_{\text{gt},ab})$

Generated Output I

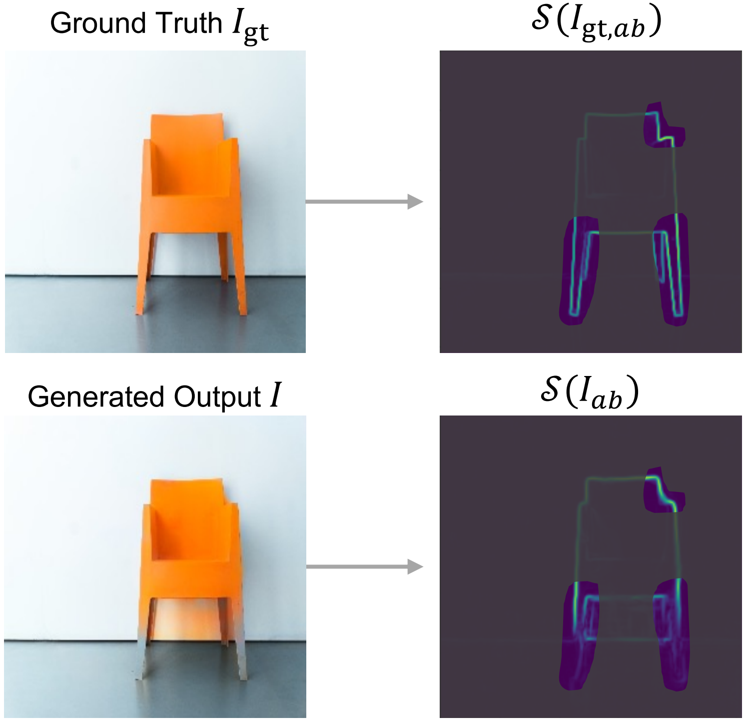
$\mathcal{S}(I_{ab})$



Proposed Method: Edge-enhancing Loss



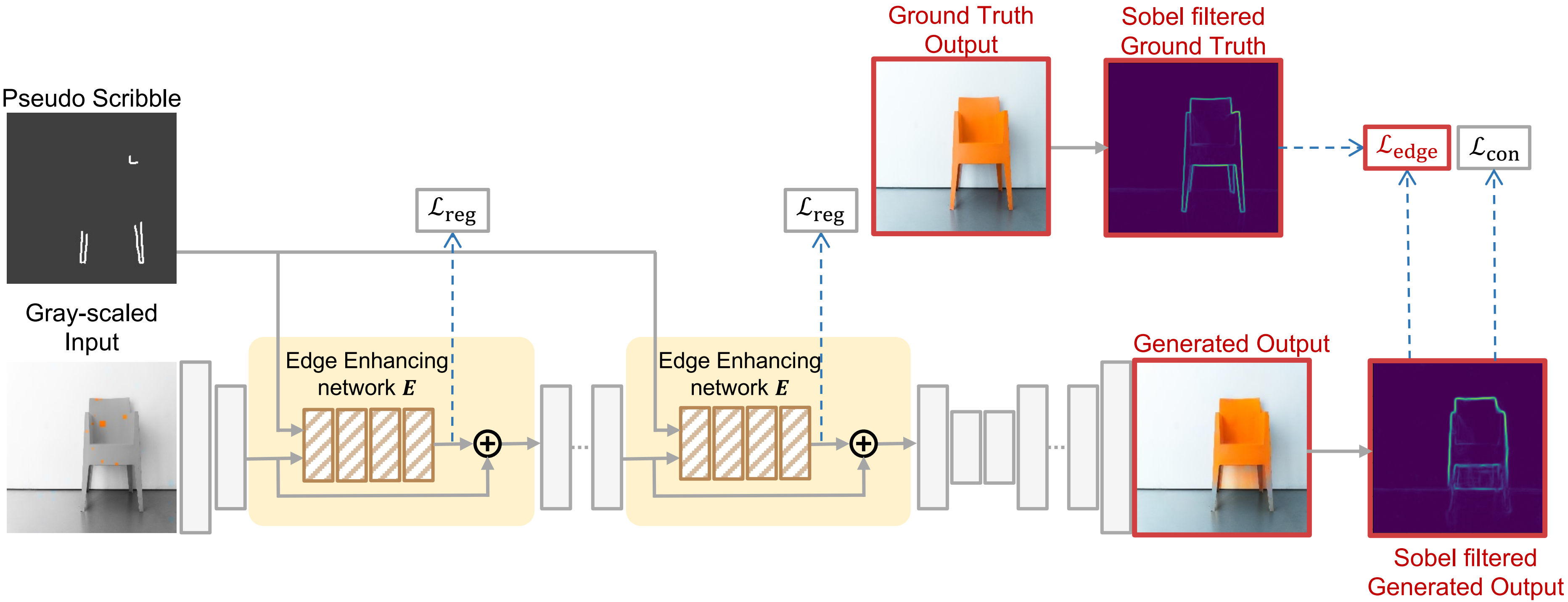
Proposed Method: Edge-enhancing Loss



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

$$S = \mathcal{S}(I_{ab}), S_{gt} = \mathcal{S}\left(I_{\mathrm{gt},ab}\right).$$

Proposed Method: Edge-enhancing Loss

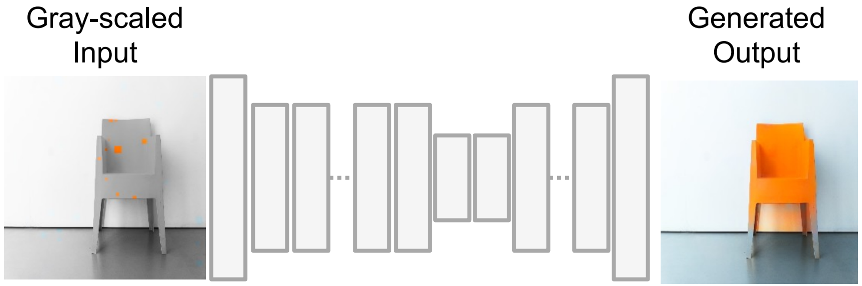


$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{edge}}$$

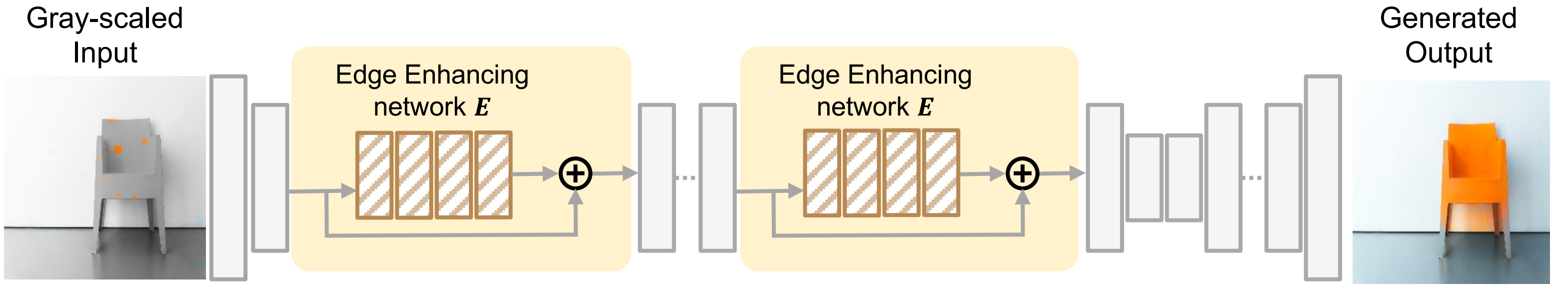
Proposed Method: Consistency Loss

Consistency Loss prevents the ***unintentional color changes*** which may appear outside of the annotated region.

Proposed Method: Consistency Loss



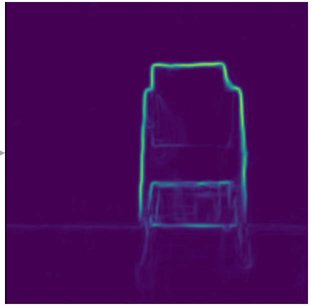
Proposed Method: Consistency Loss



Proposed Method: Consistency Loss

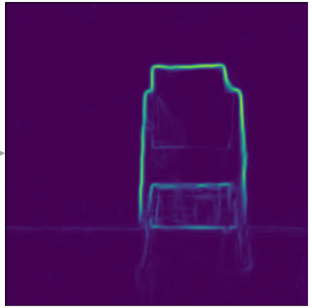
Initial Output I_{init}

$\mathcal{S}(I_{\text{init},ab})$

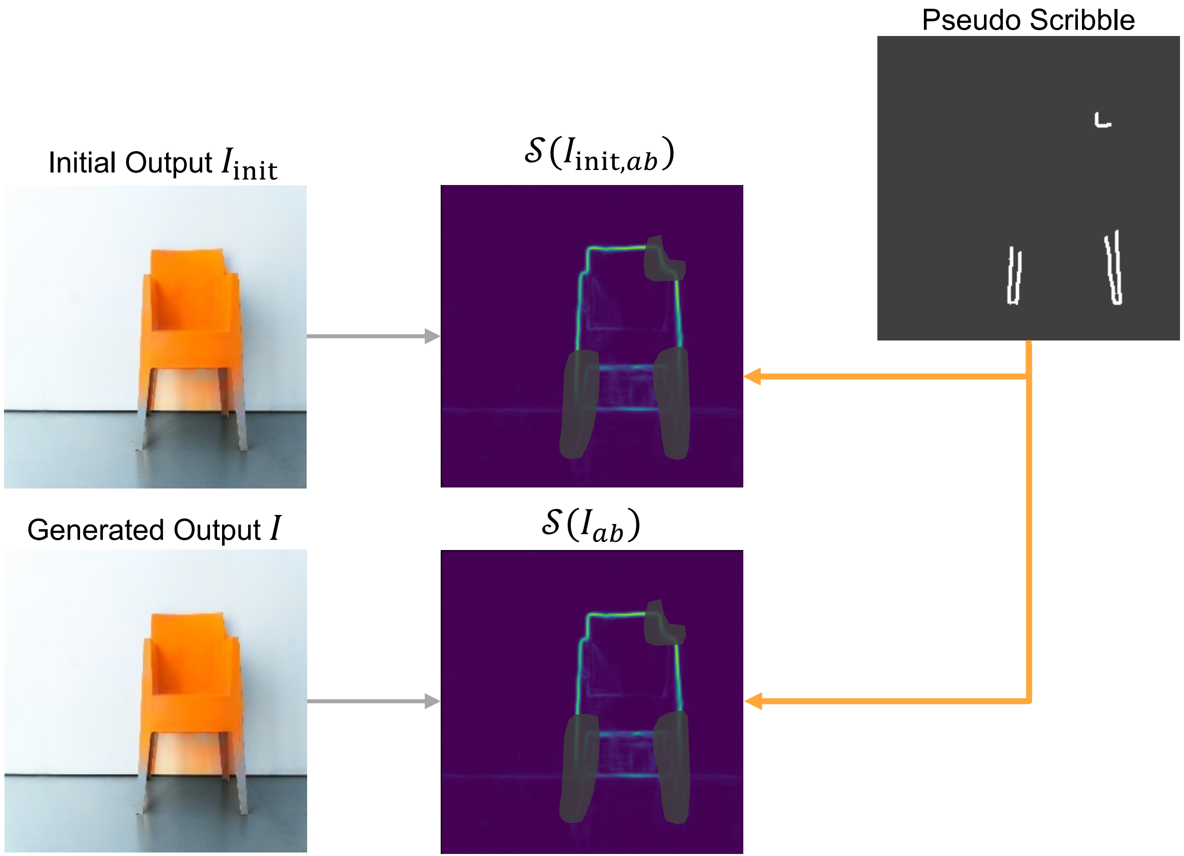


Generated Output I

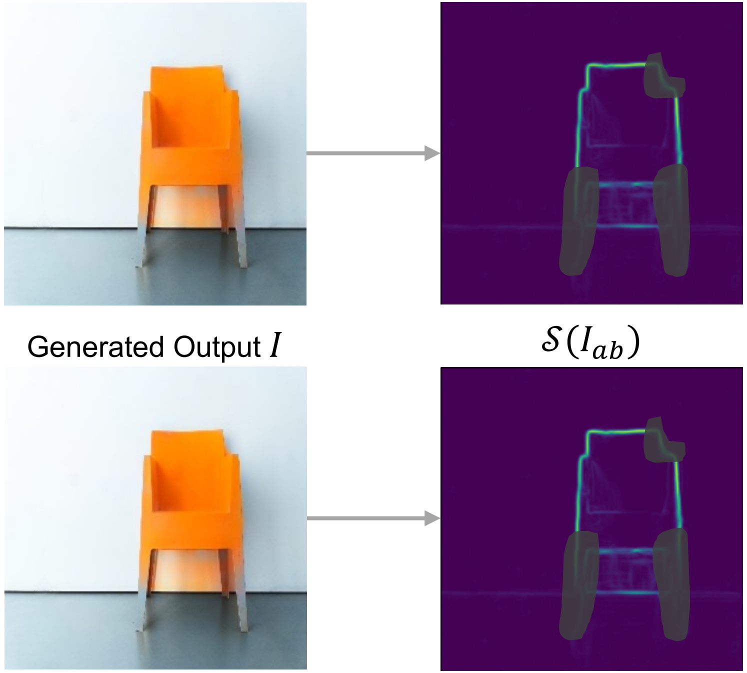
$\mathcal{S}(I_{ab})$



Proposed Method: Consistency Loss

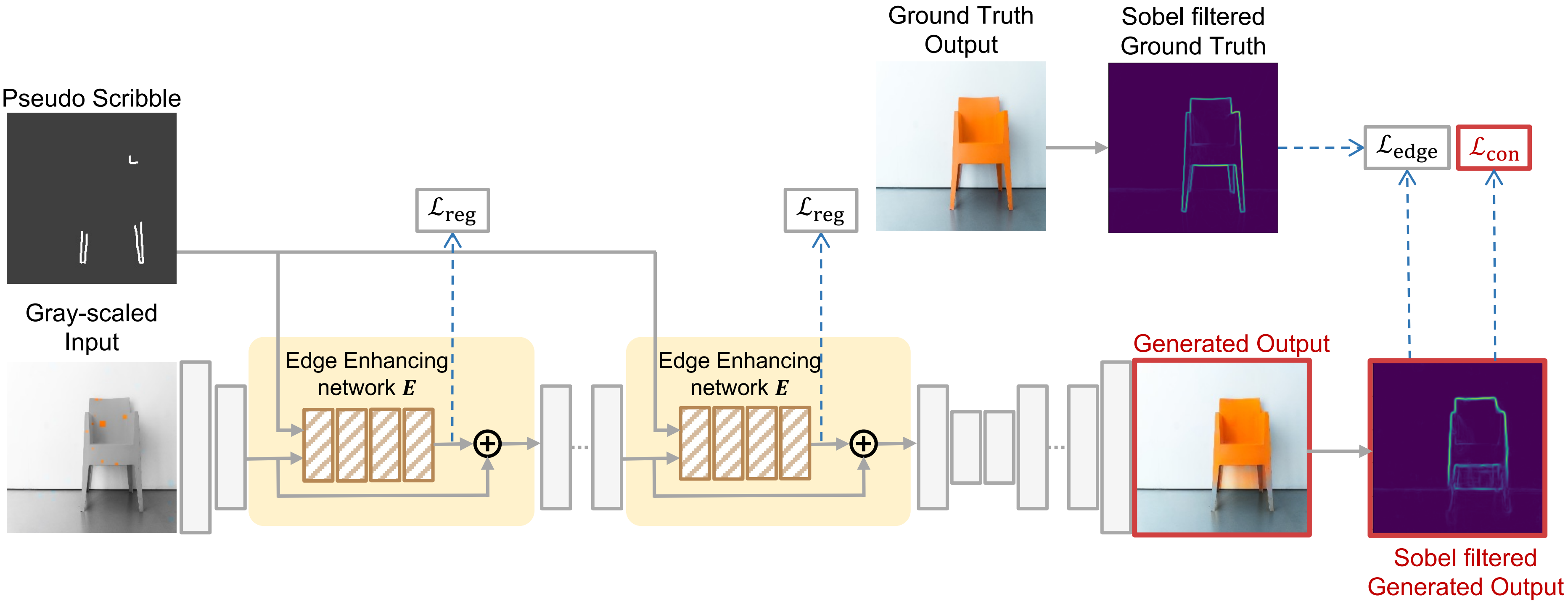


Proposed Method: Consistency Loss



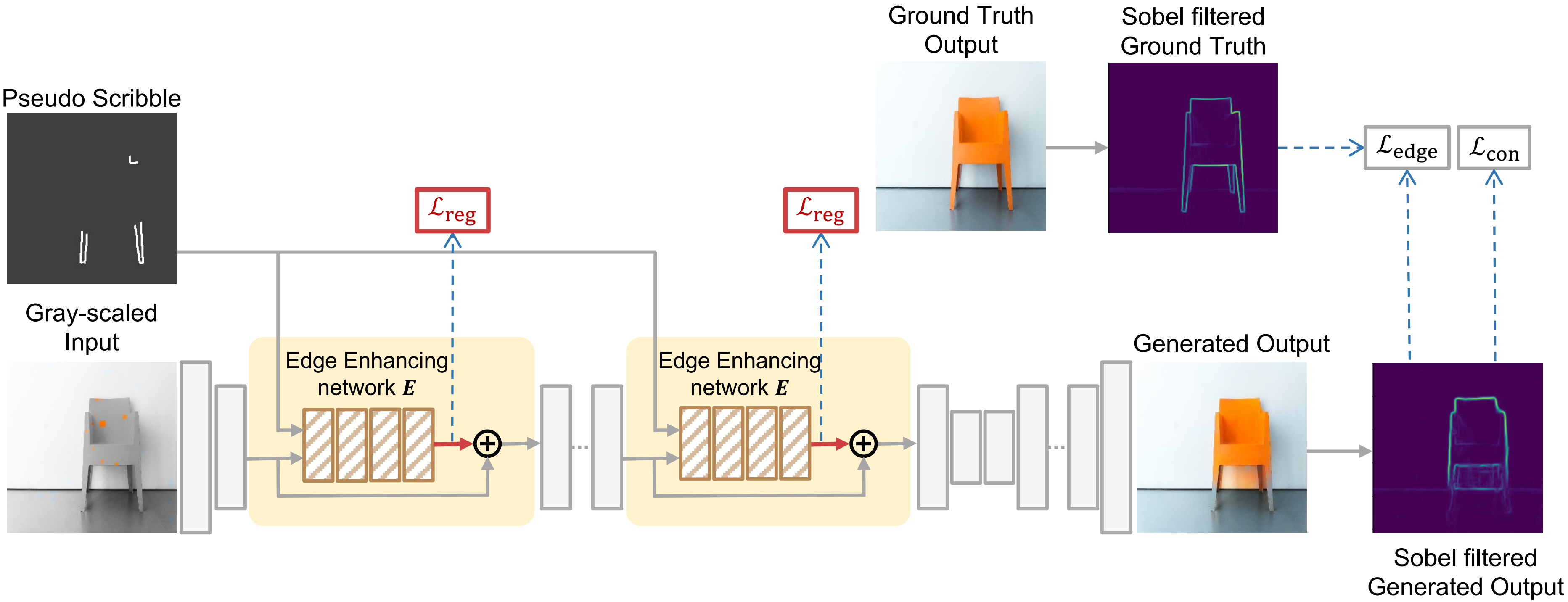
$$\mathcal{L}_{\text{con}} = \mathbb{E}_{x,y \notin P}[\|S(x,y) - S_{init}(x,y)\|_2^2]$$

Proposed Method: Consistency Loss



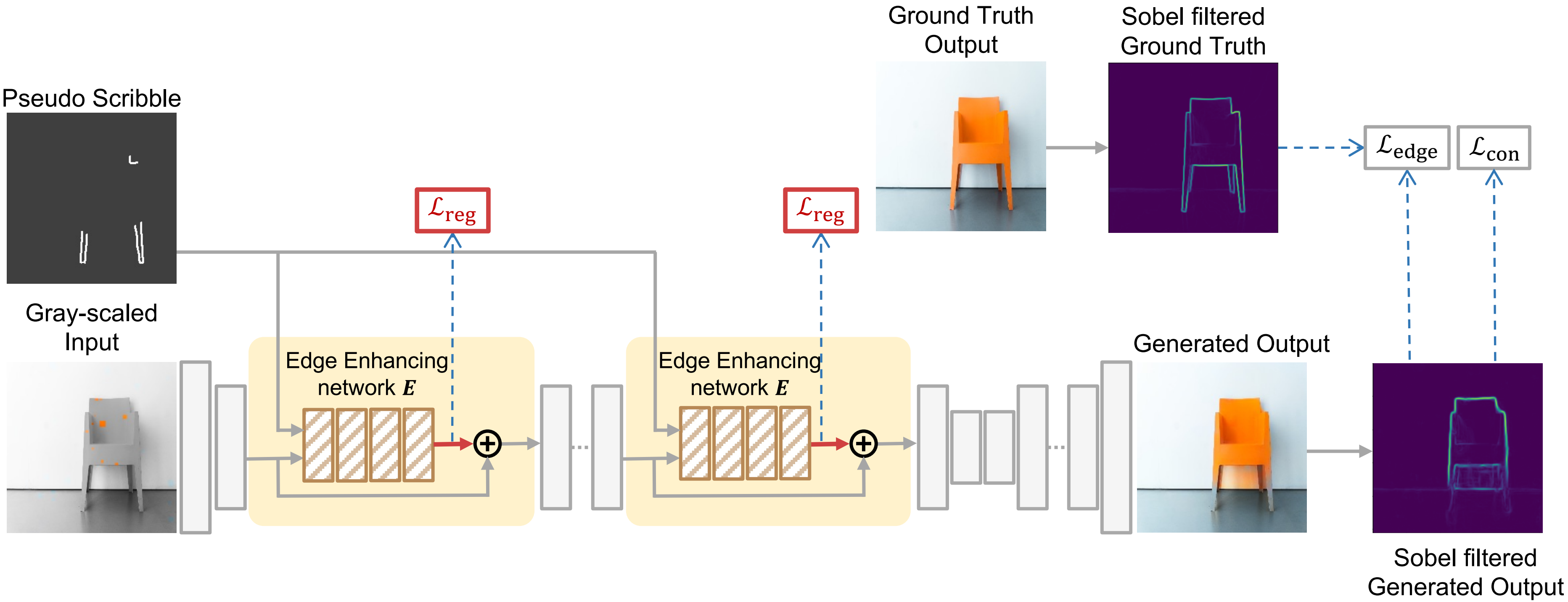
$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{edge}} + \lambda_{\text{con}}\mathcal{L}_{\text{con}}$$

Proposed Method: Feature Regularization Loss



Feature-regularization Loss minimizes the ***excessive perturbations*** on the refinement of activation maps by E.

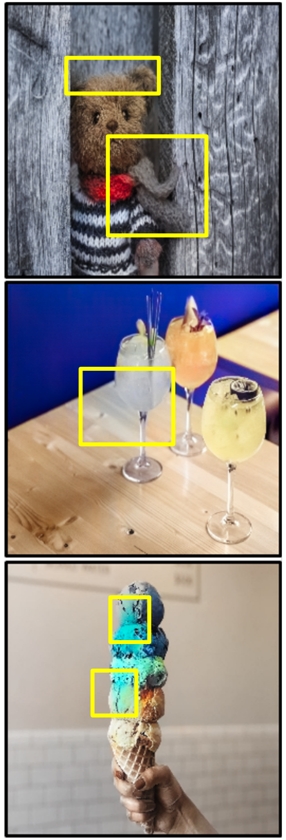
Proposed Method: Feature Regularization Loss



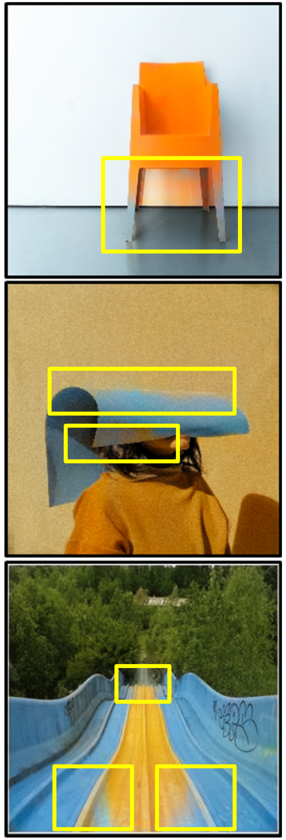
$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{edge}} + \lambda_{\text{con}}\mathcal{L}_{\text{con}} + \sum_{i} \lambda_{\text{reg}_i}\mathcal{L}_{\text{reg}_i}$$

Edge-enhancement Results: Gray-scale Colorization

Color Bleeding



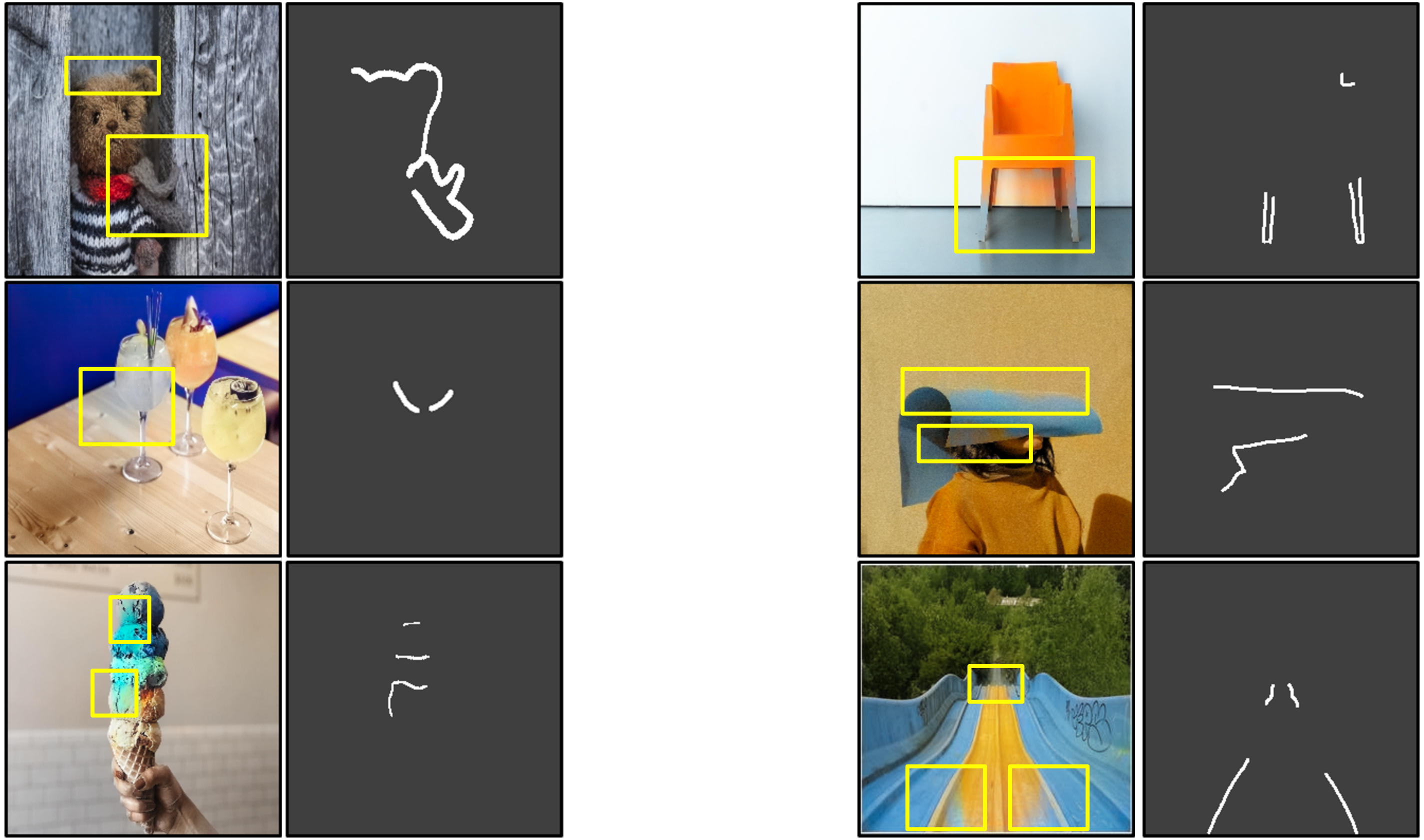
Color Bleeding



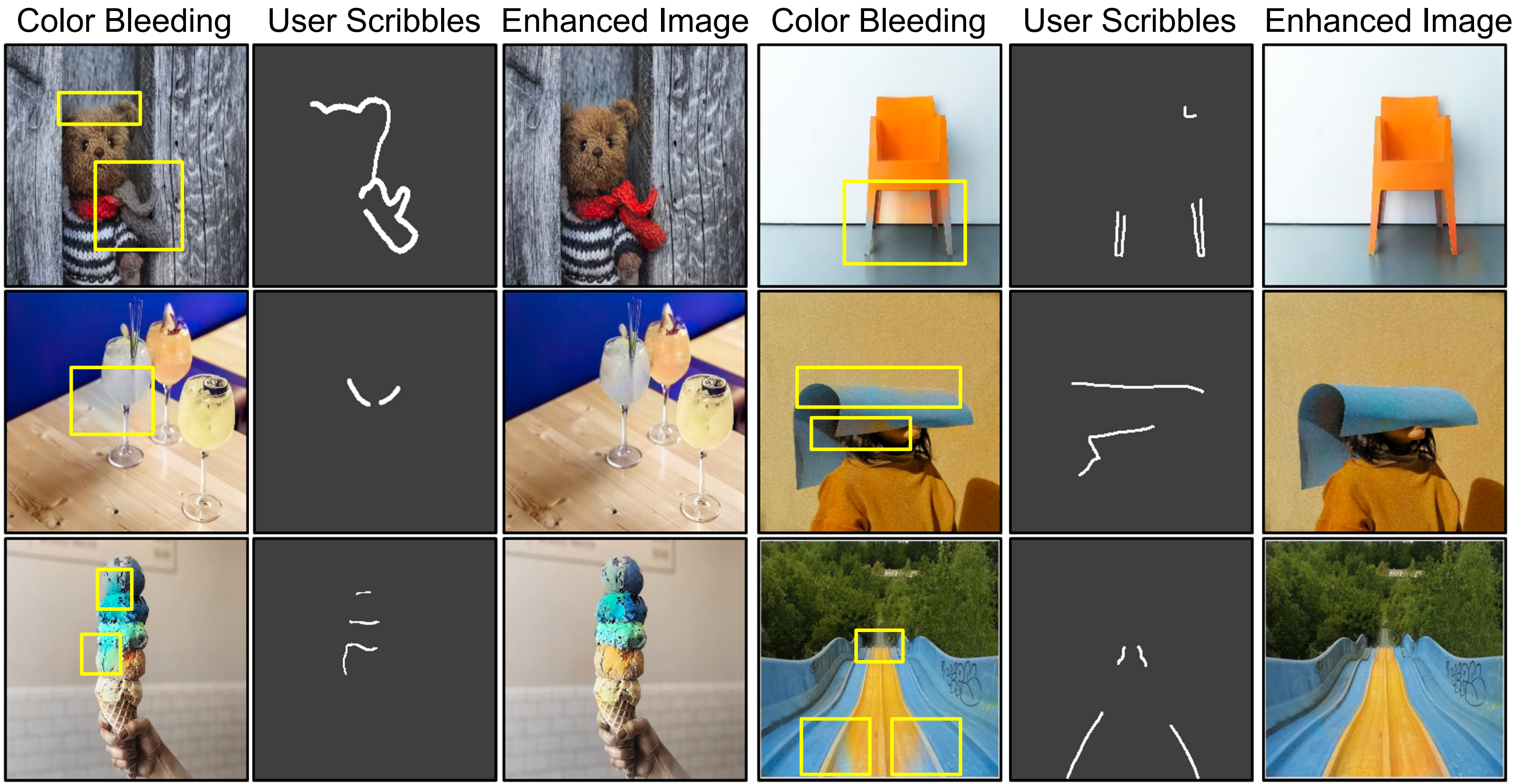
Edge-enhancement Results: Gray-scale Colorization

Color Bleeding User Scribbles

Color Bleeding User Scribbles



Edge-enhancement Results: Gray-scale Colorization



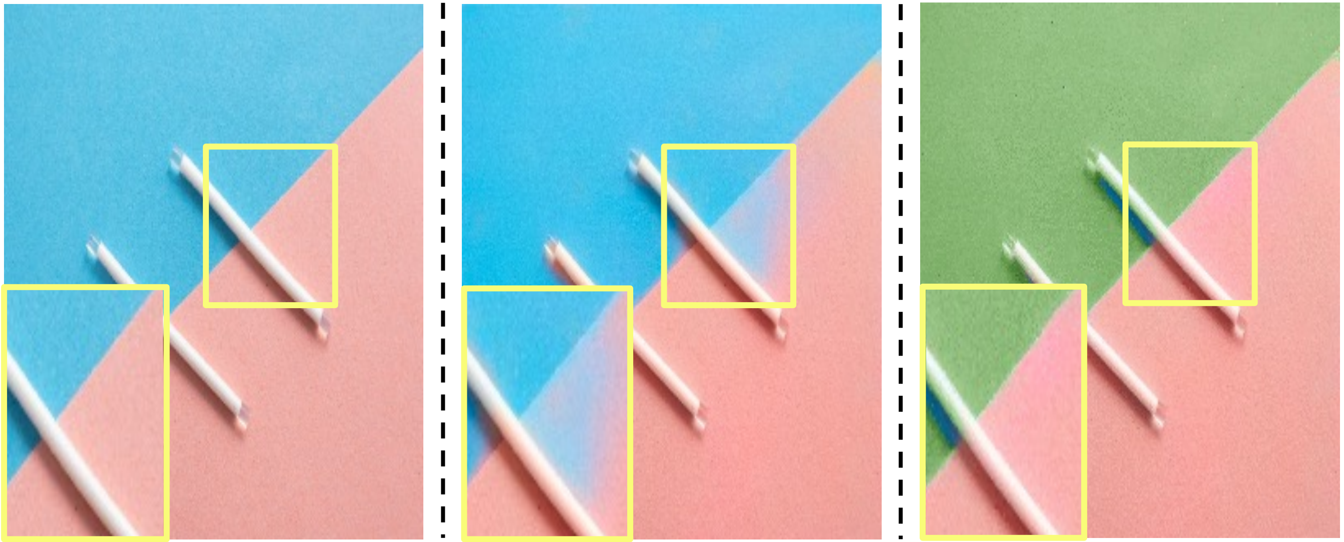
Quantitative Result: PSNR & LPIPS

Kernel Size	Methods	ImageNet ctest		COCO-Stuff		Place205	
		LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑
K=7	CIC [27]	0.248	13.281	0.247	13.368	0.254	13.577
	DeOldify [2]	0.250	13.234	0.251	13.059	0.227	14.258
	Zhang *et al.* [29]	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.* [29]*	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.* [19]*	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 [27]	0.172	21.001	0.164	21.456	0.153	21.873
	DeOldify [2]	0.159	21.433	0.149	21.985	0.156	21.933
	Zhang *et al.* [29]	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.* [29]*	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.* [19]*	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**

Our method achieves the ***SOTA*** colorization results in ImageNet, COCO-stuff, and Place205 against baselines.

Limitations of PSNR and LPIPS

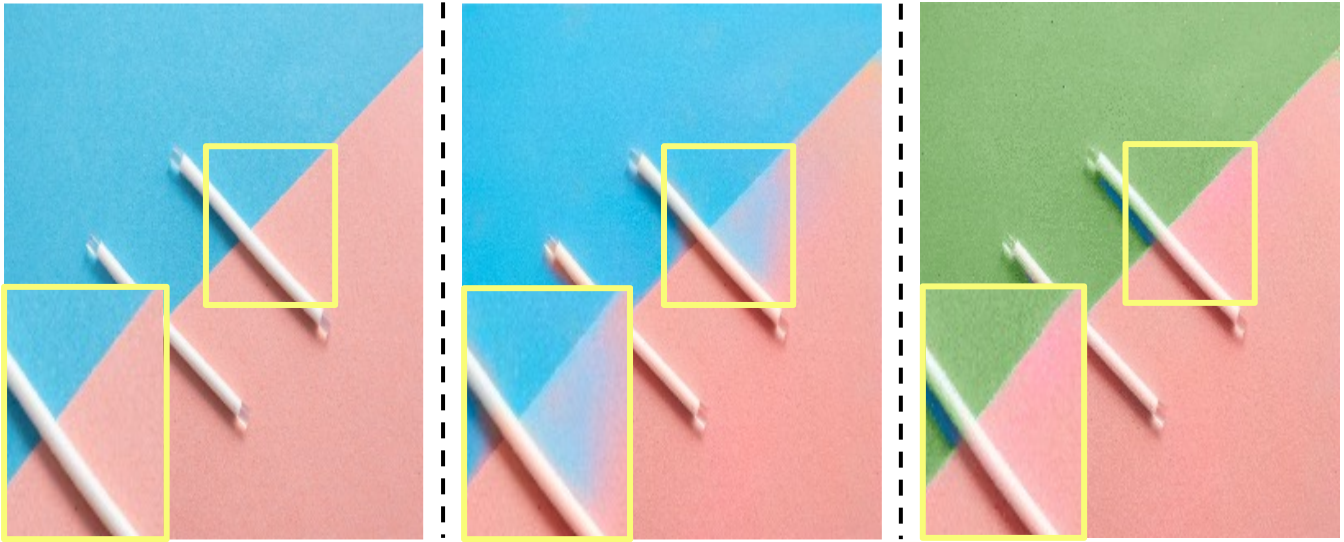
Ground Truth | Colorized Result #1 | Colorized Result #2



"Result #1 looks less realistic than Result #2 due to its color-bleeding effects."

Limitations of PSNR and LPIPS

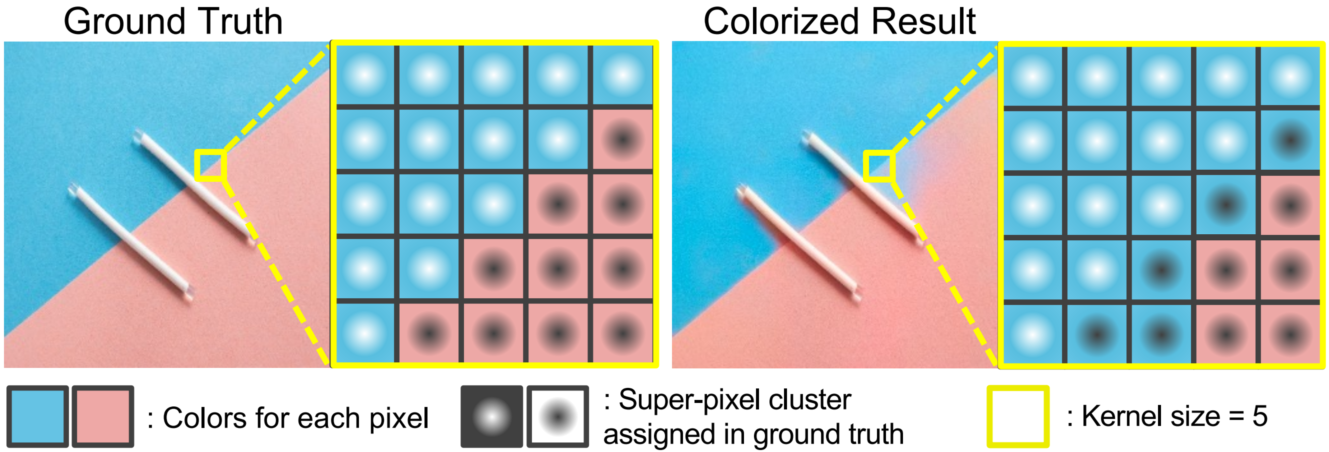
Ground Truth | Colorized Result #1 | Colorized Result #2



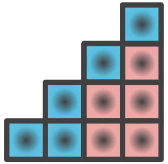
PSNR↑	**17.41**	8.73
LPIPS↓	**0.11**	0.29

"Result #2 has the better colorization quality than Result #1 ."

Proposed Evaluation Metric: Cluster Discrepancy Ratio

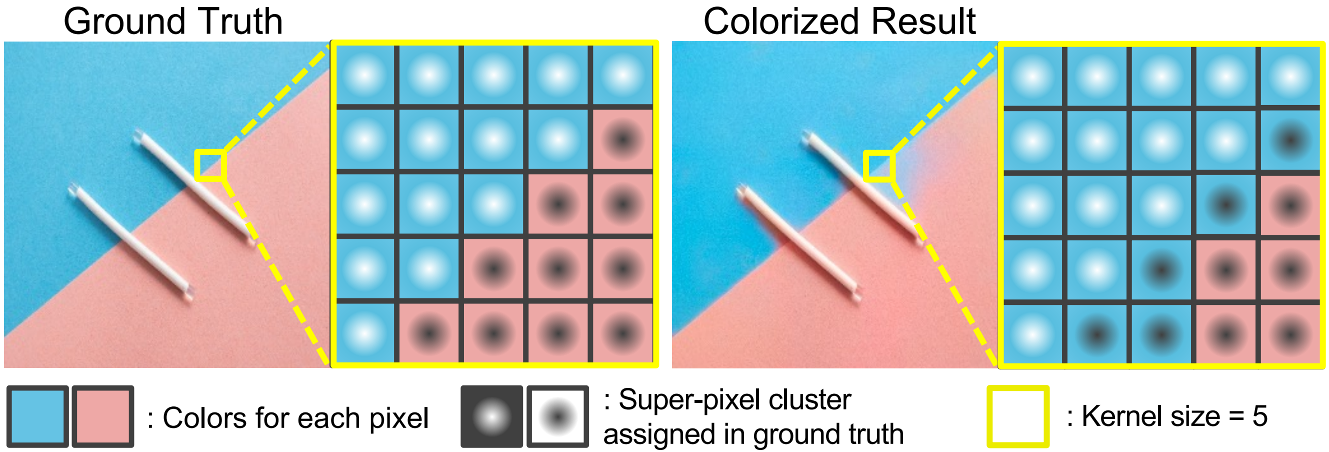


From the colorized result, we count the pixels that has



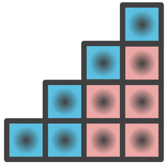
(a) the cluster (assigned in the ground truth) ***different*** from that of the ***center pixel*** in the kernel

Proposed Evaluation Metric: Cluster Discrepancy Ratio

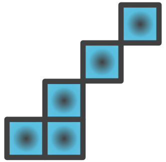


From the colorized result, we count the pixels that has

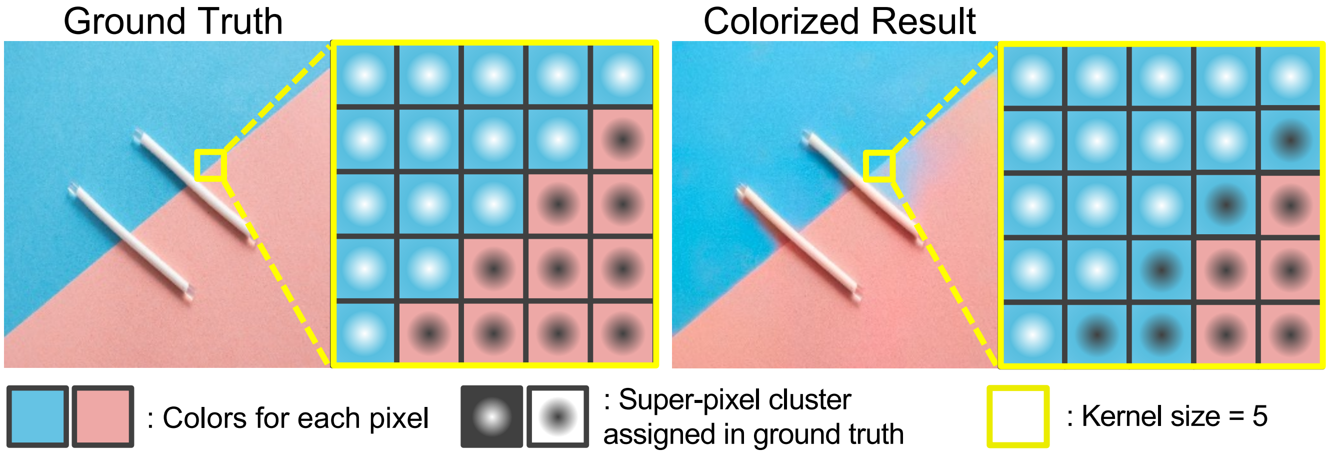
(a) the cluster (assigned in the ground truth) ***different*** from that of the ***center pixel*** in the kernel



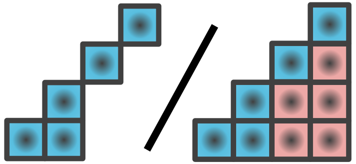
(b) also the cluster (assigned in the colorized result) is the ***same*** as that of the ***center pixel*** in the kernel



Proposed Evaluation Metric: Cluster Discrepancy Ratio

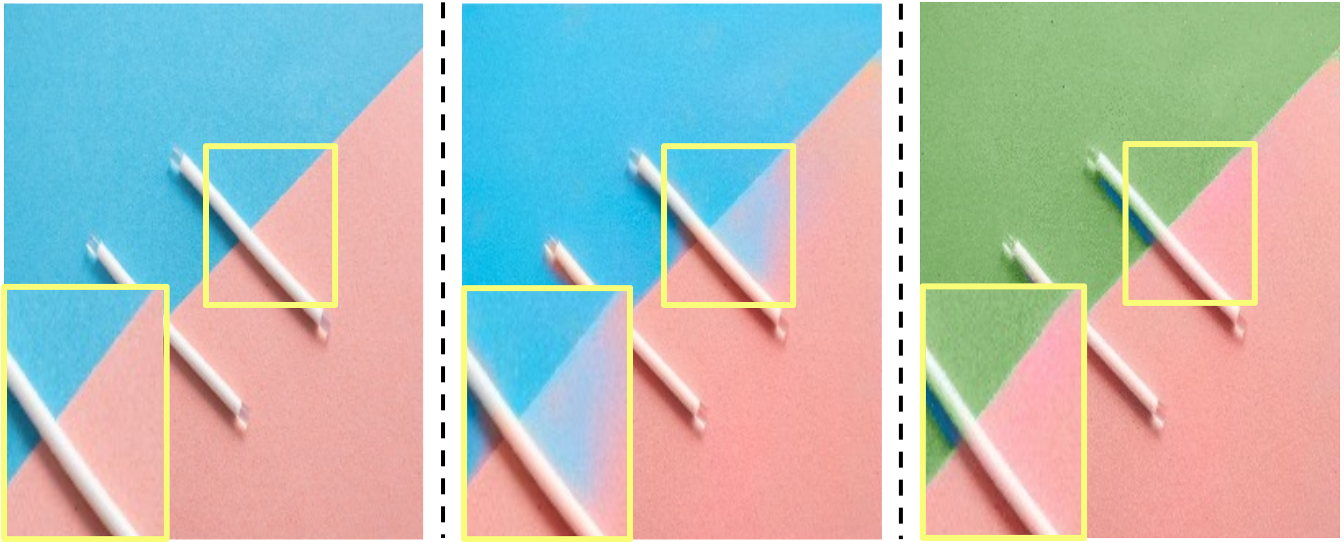


We calculate the ***ratio of two numbers*** and subtract it from one.

Cluster Discrepancy Ratio (CDR) = 1 − ()

Proposed Evaluation Metric: Cluster Discrepancy Ratio

	Ground Truth	Colorized Result #1	Colorized Result #2



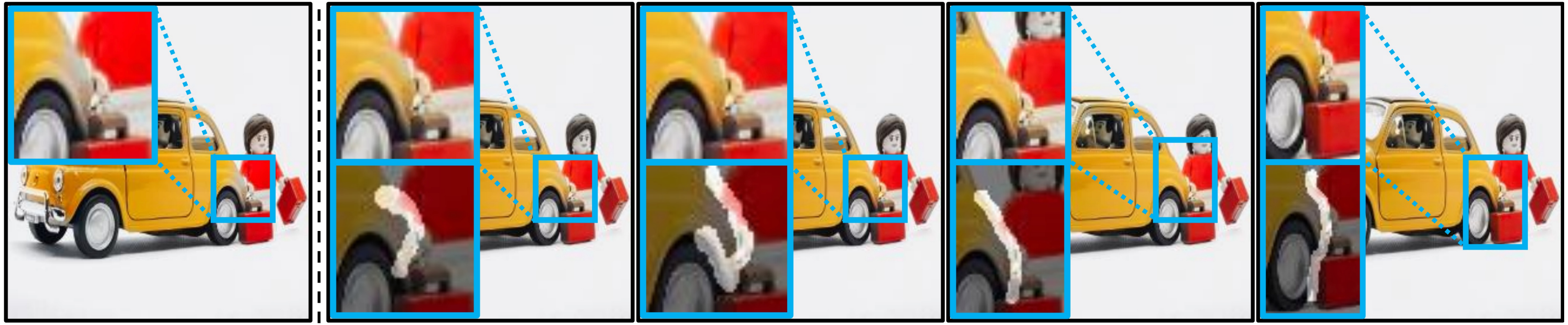
	Colorized Result #1	Colorized Result #2
PSNR↑	**17.41**	8.73
LPIPS↓	**0.11**	0.29
CDR↑	0.128	**0.838**

Quantitative Result: Cluster Discrepancy Ratio

Method	Cluster Discrepancy Ratio↑		
	ImageNet	COCO	Place205
CIC [27]	0.383	0.401	0.381
DeOldify [2]	0.437	0.445	0.441
Zhang *et al*. [29]	0.385	0.391	0.377
+ Ours	**0.502**	**0.521**	**0.473**
Zhang *et al*. [29]*	0.418	0.421	0.402
+ Ours *	**0.543**	**0.547**	**0.508**
Su *et al*. [19]*	0.336	0.325	0.336
+ Ours *	**0.394**	**0.398**	**0.371**

Our method achieves the ***SOTA*** colorization results in ImageNet, COCO-stuff, and Place205 against baselines.

User Study Result: Robust Enhancement across Different Users



	Colorized Result	User #1	User #2	User #3	User #4
PSNR ↑	13.628	16.379 (+ 2.751)	16.328 (+ 2.7)	16.89 (+ 3.262)	16.111 (+ 2.483)
LPIPS ↓	0.125	0.09 (- 0.035)	0.089 (- 0.036)	0.087 (- 0.038)	0.085 (- 0.04)
CDR ↑	0.22	0.237 (+ 0.017)	0.273 (+ 0.053)	0.263 (+ 0.043)	0.262 (+ 0.042)

Even when the user scribbles are significantly ***varying*** given the same color-bleeding images, our method ***robustly enhances*** the colorization results with such scribbles.

Thank you for watching!

Please check more information about our paper at

ArXiv

Project Page

Demo Video