Late-resizing: A Simple but Effective Sketch Extraction Strategy for Improving Generalization of Line-art Colorization

Dohyun Kim*† Dajung Je* Kwangjin Lee Moohyun Kim Han Kim
Naver Webtoon Corporation, Seoul, Korea
{dohyun9404, dajung.je, kwangjin.lee, moohyun, hanish.k}@webtoonscorp.com

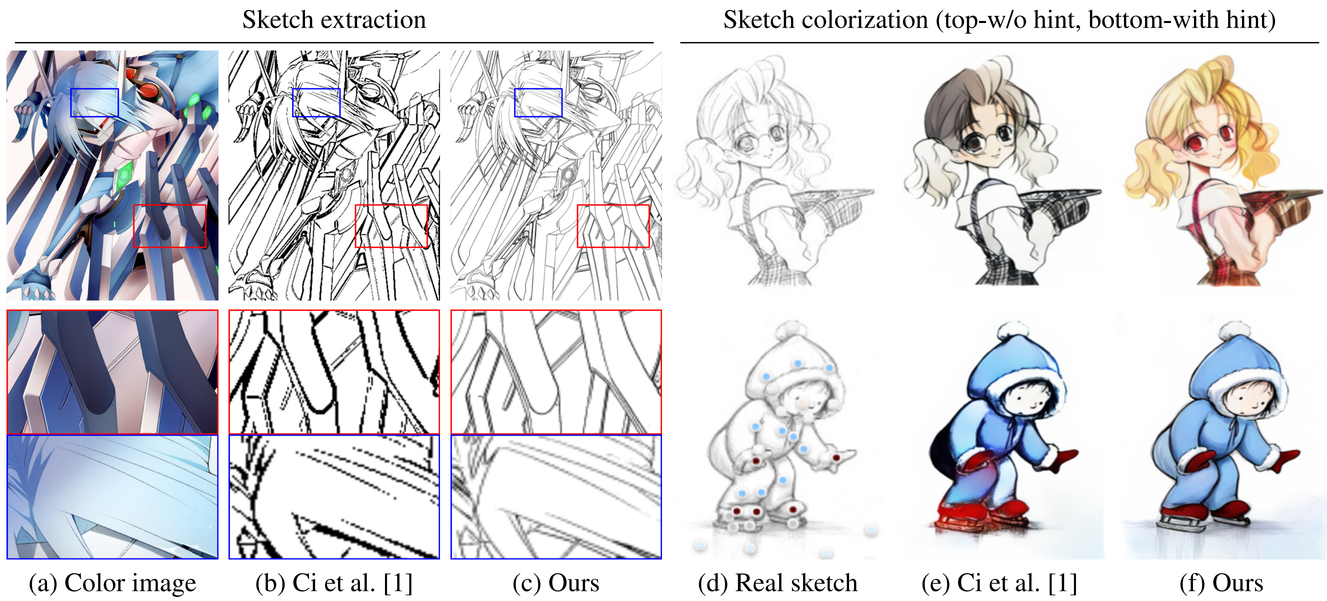


Figure 1. Examples of (a)-(c) sketch extraction used to generate trainset and (d)-(f) sketch colorization for real sketches with and without color hint. All samples were selected from the Safebooru dataset [2].

Abstract

Automatic line-art colorization is a demanding research field owing to its expensive and labor-intensive workload. Learning-based approaches have lately emerged to improve the quality of colorization. To handle the lack of paired data in line art and color images, sketch extraction has been widely adopted. This study primarily focuses on the resizing process applied within the sketch extraction procedure, which is essential for normalizing input sketches of various sizes to the target size of the colorization model. We first analyze the inherent risk in a conventional resizing strategy, i.e., early-resizing, which places the resizing step before the line detection process to ensure the practicality. Although the strategy is extensively used, it involves an often overlooked risk of significantly degrading the generalization of the colorization model. Thus, we propose a late-resizing strategy in which resizing is applied after the line detection step. The proposed late-resizing strategy has three advantages: prevention of a quality degradation in the color image, augmentation for downsizing artifacts, and alleviation of look-ahead bias. In conclusion, we present both quantitative and qualitative evaluations on representative learning-based line-art colorization methods, which verify the effectiveness of the proposed method in the generalization of the colorization model.

*These authors contribute equally.
†Corresponding author.

1. Introduction

Automatic line-art colorization is a demanding research field in the market owing to its expensive, time-consuming, and labor-intensive workload. In previous studies, traditional energy-based methods, such as LazyBrush [3] and Manga Colorization [4], were described to handle lines with a low shape complexity. However, they are insufficient to grasp high-level features, which leads to vacancies and other unavoidable errors in the detailed parts. To address these issues, researchers have recently explored data-driven colorization methods [1, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14], which utilize learned priors, such as color scribbles [1, 5, 7, 8, 10, 11, 14] and a reference image [9, 12]. To grasp well-generalized features, learning-based a line-art colorization approach requires a massive scale of paired data of line art and color images. However, this method suffers from a lack of data because paired line art is not only sparse but also labor-intensive when manually traced from a color image. To deal with a lack of data, synthetic sketch extraction methods have been adopted to utilize abundant real color images.

However, sketch extraction causes another challenge in reducing a domain gap between synthetic and real sketches, thus degrading the generalization of the colorization model. One of the characteristics that make the extraction of line art a challenge is the diversity in styles, such as intensity and thickness. To avoid sensitivity to noise and monotony in styles of traditional sketch extraction methods [15, 16], Liu et al. [7] and Sangkloy et al. [8] adopted an XDoG filter [17] that can extract sketches with different levels of detail through parameter adjustments. PaintsChainer [10] and Zhang et al. [11] presented learning-based sketch extraction methods to reflect both perceptually realistic characteristics and the semantics of the content, which are not grasped in hand-defined, prior-based methods [15, 16, 17]. To improve prior knowledge during training, these methods are combined in various ways using the post-processing techniques, such as an intensity adjustment or sketch simplification [18]

The sketch extraction process consists of three common parts: resizing, line segment detection, and post-processing. In particular, scale normalization is an essential process preceding the training and inference stage owing to the fixed kernel size of the CNN [19] architecture. By convention, the resizing step has long been applied before the line detection step (as shown in the top-right column in Fig. 2) for a few practical reasons, such as saving data storage and reducing the complexity of the sketch extraction operator. However, we determined that an *early resizing* not only can contaminate the line information while subsampling the source color images it can also complicate the subsequent line detection processes (Fig. 1 (b)). To improve the generalization of the line art colorization model, this study is aimed at mitigating to mitigate the risk of a domain gap, which is inherent in the resizing step.

We propose a simple, but effective, strategy for sketch extraction called late-resizing, in which the resizing is applied after the line detection step (as shown in the bottom-right column in Fig. 2). This simple idea alleviates the do-

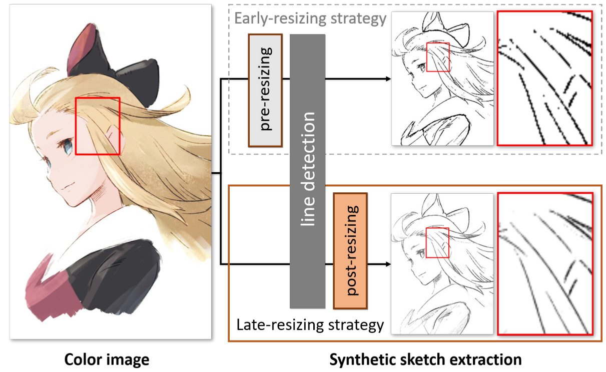


Figure 2. Examples of sketch extraction strategies.

main gap between the synthetic and real sketches from the following three perspectives. First, it prevents the resizing process from degrading the quality of the color images used as inputs in the line detection process. Second, it reproduces downsizing effects [20, 21], such as anti-aliasing and denoising, which occur during the scale-normalization process of the input sketch. Finally, it helps alleviate a look-ahead bias problem in which the model is fitted to the partial information of the posterior given to the prior, which causes a fatal degradation in the quality of the colorization for a real sketch. As one of the most powerful feature of the proposed late-resizing strategy, no further modifications are required for the adaptation unless the line detector has constraints in terms of the input size. To quantitatively evaluate whether the coloring model has failed to generalize, we also devised a saturation-sensitivity score that measures the level of activity of the input sketch is in the colorization model.

Our main contributions are as follows:

- We analyze the risk of the domain gap between synthetic and real sketches inherent in the resizing step, which is an essential process in a sketch extraction.
- We propose a simple but effective strategy for sketch extraction, called late-resizing, which improves the generalization of the line art colorization model, and is highly compatible with other existing methods.
- We devise a saturation sensitivity score to quantitatively evaluate the improvement of the generalization.
- We present both quantitative and qualitative evaluations on representative learning-based colorization methods and two different types of datasets, which verify the effectiveness of the proposed method.

2. Related Work

Leaning-Based Line-Art Colorization. Compared to gray-scale images, line-art is more challenging to colorize

Table 1. **Comparison of sketch extraction methods.**

Reference		Sketch extraction pipeline			
		Pre-resizing	Line detection	Cleaning	Post-resizing
(a)	Frans et al. [5]	256×256	Canny edge detector	thresholding	-
(b)	Auto-painter [7]	512×512	XDoG	tanh	-
(c)	PaintsChainer [10]	-	mophology + LNeT (9:1)	Otsu binarization (20%)	512×512
(d)	Style2Paints-v3 [11]	512×512	sketchKeras	not presented	256×256
(e)	AlacGAN [1]	512×512	XDoG	tanh	-
(f)	Tag2Pix [13]	512×512	sketchKeras + XDoG (1:1)	simplification & tanh	-
(g)	Ren et al. [14]	512×512	sketchKeras	not presented	-
(h)	Yuan and Simo-Serra [22]	512×512	sketchKeras + XDoG (1:1)	simplification & tanh	-

because only an abstracted borderline is present, without information on the expected texture or style of the content. In the recent paradigm of automatic line-art colorization, learning-based approaches have been used to combine both low, and high-level semantic information to produce perceptually satisfying colorization results. Pix2Pix [23] presents an outline for mapping one image to another by optimizing the pixel loss and an adversarial loss function in a supervised manner. Using this strategy, the usability and controllability of the colorization have been improved by conditioning additional user-interactive priors, such as color strokes [1, 5, 7, 8, 10, 11, 14], a color histogram [6], a reference image [9, 12], and even text [13]. Frans et al. [5], Zhang et al. [11], and Ren et al. [14] presented a multi-stage design for dividing a task into simpler and clearer subtasks. In addition, Ci et al. [1] adopted additional modules, i.e., ResNeXT [24] blocks and a dilated convolution [25], to increase the capacity and size of the receptive field of the colorization network, respectively. This model architecture inspired a later study [13]. Although these approaches have addressed the complexity of the colorization in a promising way, achieving a generalization for a real sketch remains a challenge owing to the difficulty in extracting sketches without a domain gap.

Sketch Extraction. To compare the existing sketch extraction methods in detail, we summarized the process of sketch extraction into four common steps: pre-resizing, post-resizing, line detection[1], and cleaning. Specifically, resizing is a significant process matchomg the target size of the colorization model and extracted sketch and is divided into two steps (i.e., pre-resizing and post-resizing) depending on whether it is applied before or after the line detection. The cleaning step is a common post-processing applied to remove the remaining artifacts from the previous step. Table 1 shows the summarized results of a sketch extraction method applied as a representative colorization approach [1, 7, 10, 11, 13, 14].

Earlier, in a study on sketch colorization [5], traditional algorithms such as a Canny edge detector [16] were used to detect line segments from color images. To avoid monotony in the style and sensitivity to noise found in traditional line detection methods, Auto-painter [7], Scribbler [8], and AlacGAN [1] adopt an XDoG filter [17], which can extract lines with different levels of detail through parameter adjustments. As presented in the original study on this topic [17], application of an XDoG filter generally precedes a hyperbolic tangent operation, which can be used for binarization of a sketch. PaintsChainer [10][2] uses a morphology-based line detection method, where a line is estimated from the difference between the source color image and its dilated results. A learning-based sketch extraction method, LNeT, was also presented based on HED [26], which reflects not only the perceptually realistic characteristics but also the semantics of the content, which have yet to be grasped in hand-defined, prior-based methods [15, 16, 17]. A morphology-based method and LNeT at a ratio of 9:1 were also used, and the extracted sketches were then binarized using an Otsu thresholding algorithm [27]. To decrease some of the noise and unimportant details generated by LNeT, Style2Paints-v3 [11] applied sketchKeras [28], which combines a rule-based algorithm and LNeT. In addition, Tag2Pix [13] uses sketchKeras and an XDoG filter at the same ratio, and a line simplification method [18] is additionally applied to adjust the presence of an unnecessary depiction or incorrect control of the line thickness. Moreover, Ren et al. [14] and Yuan and Simo-Serra [22] uses the sketchKeras [28] method without a blending or cleaning strategy.

For the resizing step, except for early PaintsChainer [10] and Style2Paints-v3 [11] approaches, all of the subsequent studies adopted an early-resizing strategy as a conventional strategy for the purpose of saving data storage. Contrary to this practice, this study demonstrates that a late-resizing strategy can improve the performance of existing studies particularly for generalization.

[1]We refer to line detection as the process of estimating the presence of line components, and sketch extraction as a broader meaning that includes all of the refinement processes of the estimated line components.

[2]Because the second stage of PaintsChainer is main stage for colorization, we focused on the sketch extraction process for this stage.

3. Methods

In this section, to show that an early-resizing strategy can degrade the generalization of the line-art colorization model (Section 3.1), we first analyze the risk of the domain gap between synthetic and real sketches inherent in the resizing step. Subsequently, we present the effectiveness of the proposed late-resizing[3] strategy, where the line detection process precedes the resizing of the color image that alleviates the degradation of the generalization (Section 3.2).

3.1. Motivation

A manual colorization process is conducted at a resolution of 1024×1024 or higher, whereas the output size of most colorization models is approximately 512×512. To simplify the equation, in our study, we assume that the size of the source color image is always larger than the target size of the sketch extraction. Subsequently, resizing needs to be applied solely through a reduction, i.e., downsizing.

In an early-resizing strategy, where the resizing of the color image precedes the line detection, the ideal sketch extraction method $\mathbb{S}^*$ satisfies the following:

$$\mathbb{S}^*(\downarrow(y_c + y_s)) = \downarrow(y_s), \tag{1}$$

where y is an arbitrary source color image, $\downarrow$ is a downsizing operator, y_s is an ideal paired sketch for color image y, and y_c is a color component that satisfies $y = y_s + y_c$. Assuming that $\mathbb{S}$ and $\downarrow$ are kernel-based convolution operations, Eq. (1) can be written as follows:

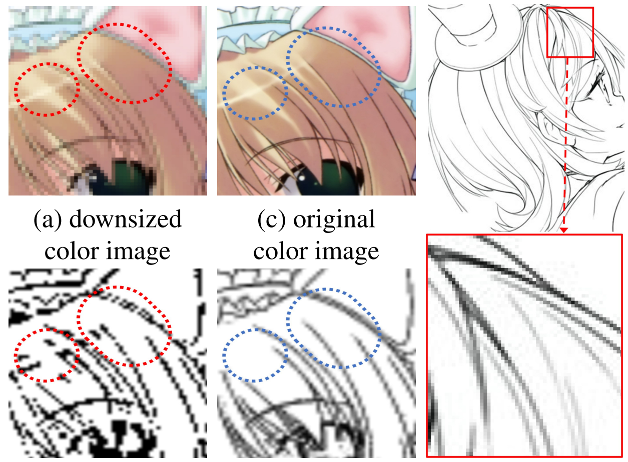
$$\mathbb{S}^*(\downarrow(y_c)) + \mathbb{S}^*(\downarrow(y_s)) = \downarrow(y_s). \tag{2}$$

In the ideal sketch extract method $\mathbb{S}^*$, downsized color component $\downarrow(y_c)$ should not remain in the extracted sketch $\mathbb{S}^*(\downarrow(y_c + y_s))$, whereas $\downarrow(y_s)$ should be intact. Thus, we can summarize the objective of $\mathbb{S}^*$ in twofold: inhibiting the activation of $\downarrow(y_c)$ and keeping $\downarrow(y_s)$ identical.

$$\mathbb{S}^*(\downarrow(y_c)) \to 0, \;\; \mathbb{S}^*(\downarrow(y_s)) \to \downarrow(y_s) \tag{3}$$

To solve Eq. (3), $\mathbb{S}^*$ needs to distinguish $\downarrow(y_s)$ from $\downarrow(y_c)$. However, this becomes an ill-posed problem because the interpolation in downsizing process not only dilutes y_s and y_c but also involves the loss of their surrounding pixel information. This necessitates $\mathbb{S}$ to restore the lost information of a line segment and estimate the type of interpolation incurred by an arbitrary pixel y_i. This implies that the early-resizing strategy has a risk of an incorrect sketch extraction. Figs. 3 (a) - (c) present the sample loss of the pixel information and the risk of incorrect extraction, where (a) denotes a color image downsized from 1860×1860 to 512×512, (b) denotes a synthetic sketch extracted from (a) through an XDoG filter [17], and (c) denotes the original color image. Compared to Fig. 3 (c), a significant quality degradation occurs in the line segments shown in Fig. 3 (a), as indicated by the red dotted circle. Finally, Fig. 3 (b) illustrates how the degraded quality in Fig. 3 (a) leads to broken lines and misleading priors, which indicate the presence of certain coloring techniques such as hair highlights in the sketch retrieved through the line detection step. This study aims at alleviating the gap between the synthetic and real sketches, which occurs because an incorrect sketch extraction is highly probable in early-resizing strategy.

[3] Because we focused on whether or not to apply the resize step before line detection, we refer to proposed methods as late-resizing, rather than post-resizing in a broader sense.



(a) downsized color image (c) original color image

(b) early-resizing (d) late-resizing (e) real sketch

Figure 3. Visual comparisons of synthetic and real sketches.

3.2. Late-resizing strategy

We propose a simple and intuitive strategy, called late-resizing, where the sketch extraction $\mathbb{S}^*$ precedes the resizing step.

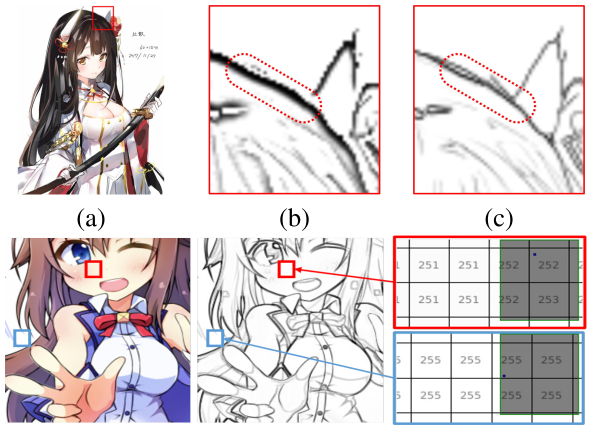
$$\downarrow(\mathbb{S}^*(y_c + y_s)) = \downarrow(y_s) \tag{4}$$

The objective of the sketch extraction $\mathbb{S}^*$ can be summarized as in Eq. (5) in the same manner as in Eq. (3). We can simplify it as Eq. (6).

$$\downarrow(\mathbb{S}^*(y_c)) \to 0, \quad \downarrow(\mathbb{S}^*(y_s)) \to \downarrow(y_s) \tag{5}$$

$$\mathbb{S}^*(y_c)) \to 0, \;\; \mathbb{S}^*(y_s)) \to y_s \tag{6}$$

The proposed late-resizing strategy differs from the traditional early-resizing strategy based on the fact that $\mathbb{S}^*$ becomes independent of the resizing process such that it is no longer affected by a downsizing. In the following subsections, we describe how this independence contributes to alleviating the domain gap between the synthetically extracted and real sketches.



(d)

Figure 4. Examples of (a)-(c) borderline artifact and (d) steganography phenomenon in line detection prior to binarization. The OpenCV [29] library was used to capture the pixel values in (d).

3.2.1 Prevention of quality degradation in color image

In the early-resizing strategy, the quality of source color images is more often degraded with unintended artifacts while downsizing, and the line detection quality is also affected by these distortion. Most distortions occur near the lines: including broken lines, loss of dim and delicate lines, and uneven irregular spots made from the interpolation between color and line pixels (Figs. 3 (a) and (b)). This places burden of restoring the lost pixel information on the sketch extraction process, which is almost impossible to achieve. By contrast, a late-resizing strategy naturally prevents a quality degradation of the color image through downsizing, as described in Eq. (6). As a result, late-resizing leads to more realistic, continuous lines of adequate thickness with less wandering speckles and spots, as shown in Fig. 3 (b). We can observe an alleviation in most of the artifacts present in Fig. 3 (b), as indicated by the blue dotted circle in Fig. 3 (d).

3.2.2 Alleviation of look-ahead bias

As another risk of the synthetic artifacts, they hide the information of the posterior to the prior in an imperceptible form. As a result, colorization models trained on these ill-extracted sketches tend to suffer from look-ahead bias [30, 31] and a fatal degradation in the colorization quality for authentic sketches. One of the most frequent and serious synthetic artifacts is dimmed borderlines spreading into the foreground areas of a line-art, as shown in the circled area of Fig. 4 (b). This is mainly caused by the ambiguity of the sketch and color components around the borderline (Eq. (3)). The extracted lines tend to be thicker and more spread on the segment colorized in darker (i.e., more saturated) colors. Colorization models trained with these artifacts are prone to interpret thicker and darker lines as prior indicators that an area needs to be colorized with darker colors.

Here, it is noteworthy that these artifacts appear more often and to a more severe extent with an early-resizing strategy. Fig. 4 (c) shows how the proposed late-resizing strategy further mitigates the spread at the borderlines. This is because the quality degradation of the color images caused by an early resizing makes it more difficult for the line extractor to distinguish between the sketch and color components. It is vital to maintain the highest quality of color images when lines are extracted to minimize distortions of the borderlines, which makes late-resizing the optimal strategy.

3.2.3 Downsizing augmentation

The independent downsizing process of the extracted sketches itself has an augmentation effect that improves the generalization of the colorization model. In the inference stage of colorization, the size of an input sketch should be normalized to the proper target size of the colorization model. Thus, reproducing the influence of interpolation on sketch images during the scale normalization, is another objective of the sketch extraction process. However, the more complex the interpolation method is, particularly the higher the number of scaling factors adopted, the more difficult it is to reproduce the influence of the interpolation. With the sketches being normalized to the target scale and thus already losing pixel information, the use of an early-resizing strategy becomes a challenge. By contrast, in the late-resizing strategy, it is intuitive that once $\mathbb{S}(y_s) \rightarrow y_s$ is correctly extracted, such influence is naturally reproduced in the subsequent downsizing process. Fig. 3 (d) shows the effectiveness of the proposed method in reproducing the influence of the interpolation, such as anti-aliasing and sub-sampling, similar to that of a real sketch presented in Fig. 3 (e).

3.2.4 Mitigating side effects of binarization

As summarized in Table 1, binarization or alternative methods such as thresholding and simplification are widely accepted in colorization studies for handling a crucial steganography phenomenon. This is because, without binarization, an imperfect activation of 255 values in the foreground areas in synthetic sketches leads to a strong look-ahead bias.

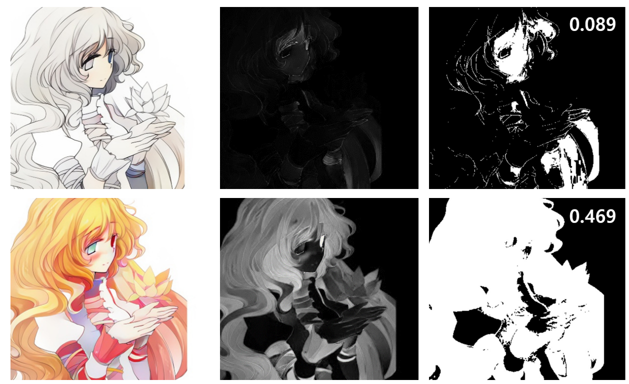
Fig. 4 (d) shows a detected line from source color image, which has yet to be binarized. The red box in Fig. 4 (d) shows how the pixels that should ideally be activated to a full value of 255 (which indicates a plain white area) fail to do so and remain at 252 or 253, whereas the blue box of Fig. 4 (d) shows well the activated 255 values. This difference comes from the difference in the source colors that each area is extracted from; here, the sketch in the blue box

Table 2. **Experiment settings and quantitative results. Here, ↓ indicates that the lower the score, the higher the performance. Fatally low SS values are marked in red, and the lower FID score for each comparison is marked in blue, except for a comparison of the synthetic sketch set.**

Method	Model	Sketch extraction pipeline				Score (FID↓/SS)		
		Pre-resize	Line detection	Binarization	Post-resize	Synthetic	SB line art	Webtoon
(a) S2Pv3 [11]	UNeT	256	XDoG	tanh	-	108.11/0.251	168.38/0.006	122.31/0.243
(b) S2Pv3-ours	UNeT	-	XDoG	tanh	256	106.21/0.357	153.82/0.341	114.12/0.302
(c) Alac [1]	AlacGAN	512	XDoG	tanh	-	21.95/0.231	107.56/0.036	42.36/0.258
(d) Alac-ours	AlacGAN	-	XDoG	tanh	512	21.44/0.209	47.08/0.216	43.56/0.224
(e) Alac-ours w/o bin	AlacGAN	-	XDoG	-	512	23.27/0.279	115.03/0.141	74.70/0.285

is extracted from plain white area, whereas the sketch in the red box is from the skin color. This activation failure becomes more apparent with more saturated colors. In real sketches, all areas other than the sketch itself are empty with a plain white background. As a result, to predict what areas to colorize and what areas to leave empty colorization models trained with these artifacts are prone to incorrectly rely on an imperfect activation in the sketches, inferred not from lines but what should be empty areas encompassed by them.

However, as a serious drawback of binarization, most real sketches used by artists are grayscale with pixel intensity information. Thus, it is highly beneficial for late-resizing strategy blends lines to closely mimic the line intensities of real sketches through an interpolation with empty areas while downsizing the extracted and binarized sketches. Comparing Fig. 3 (d) with Figs. 3 (b) and (e), a late-resizing strategy effectively brings a binarized sketch extremely close to a real sketch by less blending of the thicker lines through an interpolation whereas thinner liners are more deeply affected by the interpolation. By contrast, an early-resized sketch remains binarized.



(c) colorized result (d) saturation map (e) sensitivity map
Figure 5. Example of saturation-sensitivity score.

4. Experiments

4.1. Comparison baselines

To demonstrate the effectiveness of the proposed methods on various model architectures and training strategies, we conducted qualitative and quantitative comparisons for the two representative auto-colorization methods (i.e., Style2Paints-v3 [11] and AlacGAN [1]) and for two test sets (i.e., Safebooru and the Webtoon line-art dataset).

To analyze the robustness according to the model architecture, we compared our approach with two representative hint-based line-art colorization methods, Style2Paints [11] and AlacGAN [1]. For Style2Paints, we only handle the draft stage to focus on the difference in the degree of color expression according to the sketch, rather than the effect of refinement. We train the baselines from scratch using the Safebooru [2] dataset, and thereby use the pre-trained models to exclude the influence of additional practical factors not mentioned in the original papers of baselines. In addition all models are sufficiently trained until they no longer show any improvement in performance for the validation set. For the consistency of the experiment, the XDoG line detector was used for all experiments. Table 2 compares the baselines and the corresponding methods using our proposed method. As a downsizing operator, we used Inter-area interpolation implemented in the OpenCV library [29].

For the test set, we collected a total of 2000 real line-art data from the Safebooru [2] dataset. Although this set was not used for training, it is useful for experiments that focus on the level of the line segment, rather than the shape of the contents, in that the domain gap problem for the shape is reduced because they share the same source. In addition, we manually collected from Webtoon artists approximately 2000 datasets, called a Webtoon set, which have different painting styles distinct from Safebooru. The Safebooru line-art set is dominated by rough sketches with a wide range of intensities, whereas the Webtoon set is dominated by flat and sparse images.

4.2. Evaluation metrics

Fréchet Inception Distance (FID). FID [32] is a well-known metric for evaluating the performance of a generative model by measuring the Wasserstein-2 distance between the feature space representations of the target images

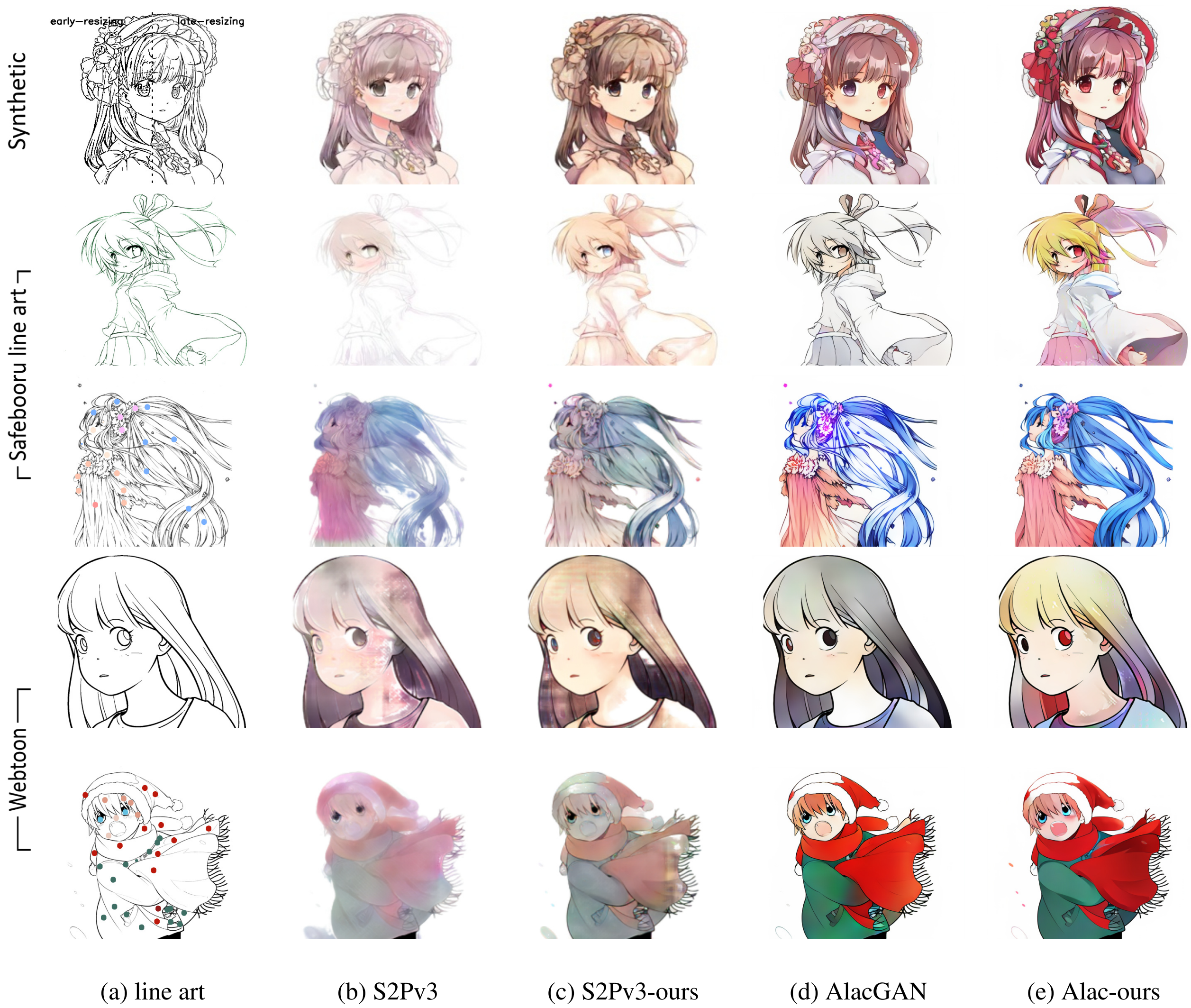


Figure 6. Visual comparison according to the application of the proposed method. Each synthetic sketch for inference was extracted in the same manner as used to train the corresponding model. The first row of each real test is the result of colorization without color hints, and the bottom row is the result of colorization when color hints are given. All sketches, including synthetic sketches, were not shown during the training process.

and the generated outputs. A low FID score indicates that the generated output has a close distribution of real data in terms of quality and diversity. Following Ci et al. [1], for fairness in applying the hint injection method, we evaluated only the results of the colorization without.

Saturation-Sensitivity Score. However, the FID is limited to separate two important aspects of the quality of generative models [33]. To explicitly measure the risk of an under-generalization, we propose a saturation-sensitivity (SS) score inspired by the light-sensitivity score presented by Ren et al. [14]. Intuitively, the color expression for the black and white sketch increases the saturation of the resulting colorized image. Therefore, if the increment of the saturation is insufficient, this implies that the activation for the corresponding sketch has not properly occurred. Based on this intuition, we propose a method for evaluating the generalization of the model by measuring the number of pixels above the pre-defined threshold in the saturation component of the colorized result. The SS score is formulated as follows:

$$\psi(Y^{sat}) = \frac{1}{n} \sum_{y^s \in Y^{sat}} \sum_{y_i^s \in y^s} \mathbb{B}(1 - y_i^s), \tag{7}$$

where ψ, $\mathbb{B}$, $y^s \in [0,1]^{H \times W}$, Y^{sat}, and y_i^s represent the SS score, binarization operation, saturation component of the colorized result in the HSV space, the set of y^s, and i-th element of y^s, respectively. Here, n, H, and W denote the number of colorized results and the height and width of the colorized result, respectively. A value of 0.07 was

(a) line-art (b) Alac-ours w/o bin (c) Alac-ours w/ bin

Figure 7. Visual comparisons of the importance of binarization. The synthetic sketch for inference was extracted in the same manner as used to train the corresponding model, and all sketches were colorized without color hints.

used as the threshold value for binarization in our comparison. Note that the SS score represents a risk for an under-generalization only when it is measured at an extremely low value, and a high score does not represent a good colorization performance. Fig. 5 shows an example of the process used calculate the SS score.

4.3. Comparisons to baselines

As summarized in Table 2, the only differences between Figs. 6 (b) and (c) and between Figs. 6 (d) and (e) are the resizing strategy. All the models showed a flawless performance for the synthetic sketches.

For the Safebooru line-art set, in the case of Figs. 6 (b) and (d), the color was under-saturated even when hints were given, as shown in the third row. Color bleeding occurred in Fig. 6 (b) and the color is less filled in Fig. 6 (d). We determined that the early-resizing method was not generalized for the Safebooru line art set, as depicted by the low SS scores of 0.006 and 0.036, respectively. However, as shown in Figs. 6 (c) and (e), when using the proposed late-resizing strategy, the coloring quality improved both when the hint was given and was not. Specifically, as shown in Fig. 6 (c), the phase expression was stronger in the result without color hints, and when hints were fed, color bleeding was alleviated. We determined that this occurred because the model in Fig. 6 (c) grasped a better understanding of the features of the input sketch than that in Fig. 6 (b). We found that this is because a higher capacity model is more effective in learning the augmentation effect obtained through a late resizing. As mentioned in Section 4.1, for the Safebooru line art set, the influence of the domain gap in the shape of the contents is small, and it was therefore concluded that a sketch extraction has a large effect on this result.

For the Webtoon set, despite the gap in the shape of the contents, Fig. 6 (b) shows a better color expression than the results for the Safebooru line-art set irrespective of the hints. We found in which the improvement results from the characteristic that the flat line segment is similar to that of the sketch extracted in Fig. 6 (b). From this analysis, we suggest that the domain gap for the characteristic of the line segment is as important as that of the shape of the contents. As shown in both Figs. 6 (c) and (e), a similar performance improvement occurred, as in the case for the Safebooru line-art set. In particular, when a hint was given, the result of Fig. 6 (e) showed more detailed and richer colorization results than that in Fig. 6 (d) without color bleeding or missing colors. This indicates that the model used for Fig. 6 (e) has not lost its generalization even for the Webtoon set.

As shown in Table 2, both the FID and SS scores of the model using the proposed late-resizing strategy were relatively high or comparable to the corresponding models using the early-resizing strategy.

In conclusion, we can verify that the late-resizing strategy proposed in this paper is effective in improving the generalization irrespective of the model architecture applied. More results can be found in the supplementary material, including comparisons with a baseline when using different types of sketch extraction techniques in combination.

4.4. Importance of binarization in late-resizing

As shown in Fig. 7 (b), the absence of the binarization process caused a fatal performance degradation, in which a color was faintly expressed; however, it presented flawless colorization results for the synthetic sketch. This might have resulted from the look-ahead bias for the region information, which had artifacts and was an unclean area in the sketch, as discussed in Section 3.2.2. By contrast, in the case of using binarization as shown in Fig. 7 (c) showed plausible result. This indicates that the proposed strategy has a synergy with binarization, thereby alleviating the bias.

5. Concluding Remarks

In this paper, we proposed a simple but effective strategy for sketch extraction called late-resizing, in which resizing was applied after in our proposed method, detection step. Compared with a traditional early-resizing strategy, the line detection process is no longer dependent on the resizing process, which contributes to alleviating the domain gap problem between the synthetic and real sketches and the look-ahead bias. The experimental results showed the potential to improve the generalization of the colorization model. However, the expected benefit of the proposed method may be limited when the difference between the size of the source color image and target size is insignificant. In future studies research we will adopt super resolution methods to upscale the source color images and utilize a late-resizing method. We expect that the proposed method will contribute to alleviating the ambiguity of the sketch extraction strategy for future line-art colorization studies.

References

[1] Yuanzheng Ci, Xinzhu Ma, Zhihui Wang, Haojie Li, and Zhongxuan Luo. User-guided deep anime line art colorization with conditional adversarial networks. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1536–1544, 2018.

[2] Anonymous, Danbooru community, and Gwern Branwen. Danbooru2020: A large-scale crowdsourced and tagged anime illustration dataset. `https://www.gwern.net/Danbooru2020`, January 2021. Accessed: DATE.

[3] Daniel Sỳkora, John Dingliana, and Steven Collins. Lazybrush: Flexible painting tool for hand-drawn cartoons. In *Computer Graphics Forum*, volume 28, pages 599–608. Wiley Online Library, 2009.

[4] Yingge Qu, Tien-Tsin Wong, and Pheng-Ann Heng. Manga colorization. *ACM Transactions on Graphics (TOG)*, 25(3):1214–1220, 2006.

[5] Kevin Frans. Outline colorization through tandem adversarial networks. *arXiv preprint arXiv:1704.08834*, 2017.

[6] Chie Furusawa, Kazuyuki Hiroshiba, Keisuke Ogaki, and Yuri Odagiri. Comicolorization: semi-automatic manga colorization. In *SIGGRAPH Asia 2017 Technical Briefs*, pages 1–4. 2017.

[7] Yifan Liu, Zengchang Qin, Tao Wan, and Zhenbo Luo. Autopainter: Cartoon image generation from sketch by using conditional wasserstein generative adversarial networks. *Neurocomputing*, 311:78–87, 2018.

[8] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2017.

[9] Lvmin Zhang, Yi Ji, Xin Lin, and Chunping Liu. Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan. In *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 506–511. IEEE, 2017.

[10] Preferred Networks. Paintschainer, 2017.

[11] Tien-Tsin Wong Yi Ji LvMin, Chengze Li Zhang, and Chun-Ping Liu. Two-stage sketch colorization. *ACM Transactions on Graphics*, 37(6), 2018.

[12] Junsoo Lee, Eungyeup Kim, Yunsung Lee, Dongjun Kim, Jaehyuk Chang, and Jaegul Choo. Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5801–5810, 2020.

[13] Hyunsu Kim, Ho Young Jhoo, Eunhyeok Park, and Sungjoo Yoo. Tag2pix: Line art colorization using text tag with secat and changing loss. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9056–9065, 2019.

[14] Hui Ren, Jia Li, and Nan Gao. Two-stage sketch colorization with color parsing. *IEEE Access*, 8:44599–44610, 2019.

[15] N. Kanopoulos, N. Vasanthavada, and R. L. Baker. Image edge detection filter using the sobel operator. *IEEE Journal of Solid-State Circuits*, 23(2):358–367, 1988.

[16] John F. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

[17] Holger Winnemöller. Xdog: Advanced image stylization with extended difference-of-gaussians. In John P. Collomosse, Paul Asente, and Stephen N. Spencer, editors, *9th International Symposium on Non-Photorealistic Animation and Rendering, NPAR 2011, Vancouver, BC, Canada, August 5-7, 2011, Proceedings*, pages 147–156. ACM, 2011.

[18] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2017.

[19] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[20] Lin Zhong, Sunghyun Cho, Dimitris N. Metaxas, Sylvain Paris, and Jue Wang. Handling noise in single image deblurring using directional filters. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 612–619. IEEE Computer Society, 2013.

[21] Agung W Setiawan, Amir Faisal, and Nova Resfita. Effect of image downsizing and color reduction on skin cancer pre-screening. In *2020 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, pages 148–151. IEEE, 2020.

[22] Mingcheng Yuan and Edgar Simo-Serra. Line art colorization with concatenated spatial attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3946–3950, 2021.

[23] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[24] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[25] F Yu and V Koltun. Multi-scale context aggregation by dilated convolutions. arxiv 2015. *arXiv preprint arXiv:1511.07122*, 615, 2019.

[26] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *Int. J. Comput. Vis.*, 125(1-3):3–18, 2017.

[27] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEETrans. Sys. Man. Cyber,*, 9:62–66, 1979.

[28] lllyasviel. sketchkeras, 2018.

[29] Itseez. Open source computer vision library. https://github.com/itseez/opencv, 2015.

[30] Hui Guo and Robert Savickas. Idiosyncratic volatility, stock market volatility, and expected stock returns. *Journal of Business & Economic Statistics*, 24(1):43–56, 2006.

[31] Rolf W. Banz and WILLIAM J. Breen. Sample-dependent results using accounting and market data: Some evidence. *The Journal of Finance*, 41(4):779–793, 1986.

[32] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[33] Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In *International Conference on Machine Learning*, pages 7176–7185. PMLR, 2020.