IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 July 2023, accepted 21 September 2023, date of publication 10 October 2023, date of current version 13 October 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3323216*

RESEARCH ARTICLE

# Robust Manga Page Colorization via Coloring Latent Space

**MAKSIM GOLYADKIN**[1,2,4] **AND ILYA MAKAROV**[1,3,4]

[1]Artificial Intelligence Research Institute (AIRI), 105064 Moscow, Russia
[2]International Laboratory for Intelligent Systems and Structural Analysis, HSE University, 101000 Moscow, Russia
[3]AI Center, National University of Science and Technology MISIS, 119991 Moscow, Russia
[4]Laboratory of Algorithms and Technologies for Network Analysis, HSE University, 603155 Nizhny Novgorod, Russia

Corresponding author: Maksim Golyadkin (golyadkin@airi.net)

This work was supported in part on Section 2 by the Strategic Project "Digital Business" within the framework of the Strategic Academic Leadership Program "Priority 2030" at the National University of Science and Technology (NUST) MISiS, in part by the Basic Research Program at the National Research University Higher School of Economics (HSE University), and in part by the Computational Resources of HPC Facilities at HSE University.

**ABSTRACT** Manga (Japanese comics) are commonly drawn with black ink on paper. Colorization of manga pages can enrich the visual content and provide a better reading experience. However, the existing colorization approaches are not sufficiently robust. In this paper, we propose a two-stage approach for manga page colorization that supports sampling and color modification with color hints. In the first step, we employ the Pixel2Style2Pixel architecture to map the black-and-white manga image into the latent space of StyleGAN pretrained on the highly blurred colored manga images that we call Coloring Latent Space. The latent vector is automatically or manually modified and fed into the StyleGAN synthesis network to generate a coloring draft that sets the overall color distribution for the image. In the second step, heavy Pix2Pix-like conditional GAN fuses the information from the coloring draft and user-defined color hints and generates the final high-quality coloring. Our method partially overcomes the multimodality of the considered problem and generates diverse but consistent colorings without user input. The visual comparison, the quantitative evaluation with Frechet Inception Distance, and the qualitative evaluation via Mean Opinion Score exhibit the superiority of our approach over the existing state-of-the-art manga pages colorization method.

**INDEX TERMS** Image colorization, deep learning, image generation, semi-supervised learning.

## I. INTRODUCTION

Image colorization both challenging and fascinating task. Color has a profound effect on a person's perception of the world, so the ability to bring new depth to images through coloring arouses genuine interest. For example, coloring books are popular among children, as well as legacy photo colorization among adults. However, manual colorization is heavily time-consuming even for professional artists, not to mention hobbyists, and requires a certain amount of physical effort. It can be beneficial for developing children's fine motor skills but is quite discouraging regarding industrial applications. Therefore, there is an interest in developing algorithms that can simplify this process.

At the beginning of the century, there were proposed approaches for image colorization that used human assistance in picking colors, which demanded less effort but still involved a great deal of interaction. The new wave of interest is related to the introduction of learning-based methods. The encouraging results of their application to image colorization were not surprising since machine learning algorithms had achieved success in various fields of science and entertainment. Several approaches based on training CNN deep learning models on large-scale image datasets in an end-to-end fashion have been proposed [1], [2]. Some of them perform the colorization manually [3], whereas others use user-constructed input [4], [5].

Even more challenging task is the manga page colorization. Real objects have not so much variance in visual appearance, but drawings of the same things, especially ones of different styles, can have notably distinct appearances whereas retaining the natural features that can be recognized by a human but hardly by a computer. There is a large industry of manga production in Japan that has been creating a

The associate editor coordinating the review of this manuscript and approving it for publication was Yongjie Li.

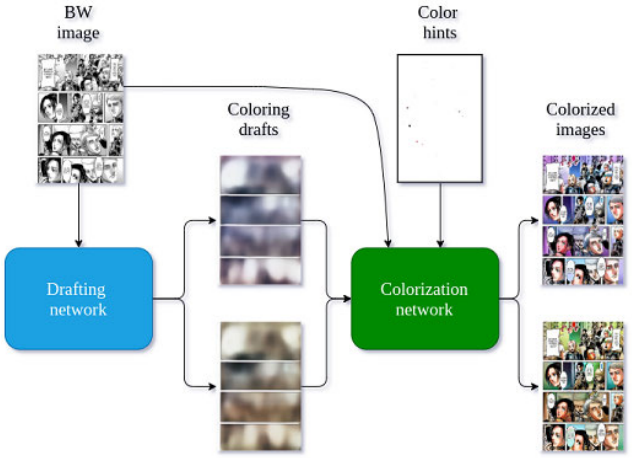


**FIGURE 1.** The overview of the proposed two-stage approach. In the first stage, the drafting network generates distinct coloring drafts for a given manga page image using style mixing. In the second stage, the colorization network produces a coloring for the black-and-white image by using the generated coloring drafts and user-defined color hints.

massive amount of black-and-white images for a few decades. Manga images are richly diversified in terms of content and visual appearance but share a common structure and drawing approach. Therefore, it makes sense to consider it as a self-sufficient data collection distinct from the other artistic images. A typical manga page consists of several panels with varying types of content. This content includes a lot of intersections, occlusions and shape distortion. Therefore, manga images are a good benchmark for computer vision algorithms due to their complexity. In addition, there is significantly more black-and-white manga than color one, providing great potential for semi-supervised methods [6], [7].

Most of the existing methods of artistic image colorization [8], [9], [10] are actually designed for the colorization of sketches drawn in the Japanese style. Most of these sketches depict one or more characters on a homogeneous background, which is substantially more primitive content than manga pages. The direct application of these methods does not allow to obtain qualitative results because of the domain gap caused by training on synthetic data, which is generated by the methods used in these works. Rare works focused directly on the colorization of manga pages [9], [11] produce colorings of insufficient quality.

Existing colorization methods suffer from the mode collapse intrinsic to adversarial learning. Hence, they have a strong preference for some colors. This limitation may be acceptable for natural images but imposes an expressiveness limitation for artistic ones. Therefore, there is a need for an approach that is not affected by the problems caused by the application of adversarial learning to dense prediction. This problem may be coped with the employment of user input, but this approach requires additional manipulation. Our work focuses on the improvement of automatic colorization.

In this work, we propose an approach for the colorization of black-and-white manga pages that employs the generative power of the pretrained StyleGAN [12], [13] for the coloring generation and supports its modification with color hints. The use of StyleGAN provides a different approach to the colorization problem, and its rich latent space contributes to the generation of more believable images, as well as enables to produce semantically meaningful transformations by changing the values of the style vector.

Our contributions are summarized as follows:

- We have explored the ability of StyleGAN to learn the distribution of color manga images and its ability to build a proper latent space for existing projection methods.
- We have proposed a learning-based manga colorization method that utilizes pretrained StyleGAN and exhibits high visual performance in automatic mode, as well as qualitatively supports image modification via user input. Employment of StyleGAN increases the robustness of the model.
- We have investigated the ability of existing colorization methods trained on a synthetic pairwise dataset to generalize well on real black-and-white manga and have verified whether domain adaptation techniques can improve generalizability.

The paper is organized as follows. We review existing methods of image and manga colorization. Then we introduce our approach and explain the ideas behind it. Next, we describe the training dataset, its preprocessing, and the training procedure. Finally, we exhibit the training results and compare our approach with existing methods.

## II. RELATED WORK

Recently, there has been a significant amount of attention to the utilization of machine learning techniques for colorization. Among existing works [1], [2], the deep convolutional neural network has become the mainstream approach to learn color prediction from a large-scale dataset. [14] and [15] utilize PixelCNN [16] and Axial Transformer [17] respectively to produce colorization in autoregressive way. The most common approach to this task involves the utilization of the adversarial loss, similarly to the Pix2Pix [18] model.

Other CNN based methods are combined with user interactions. [4], [19] propose to train and inference a deep network given the grayscale image and user-defined color dots. This approach allows the user to control the colorization process. Some works [20], [21] suggest a reference-based methods using a deep learning approach. The colorization network learns to match semantic features of input and reference images to coherently colorize similar objects.

There have also appeared works on the colorization of artistic images. In wide variety of works [8], [10], [22], [23], [24], [25], [26], [27], [28] high-quality colorization are performed with user-constructed scribbles. Oppositely, there

are a vast amount of methods [9], [29], [30] that employ reference images. In addition, some methods [31], [32], [33] define colors with text. [11] combined both approaches. Reference [34] showed that sequential prediction of color and density can improve model performance. Yoo et al. [35] employed a memory network that enhances the colorization of rare objects. [36] introduced a method that colorizes manga through unsupervised domain translation between black-and-white manga and colored American comics. Reference [37] proposed a method that uses active-learning-based framework to colorize a set of sketch images using a single colored reference image. Reference [38] suggested an approach for manga page colorization with color hints.

There are approaches using pretrained StyleGAN to assist image colorization [39], [40]. These methods are based on the generation of reference color images whose content is close to the input image. However, StyleGAN is not capable of generating manga images that are hard to distinguish from real ones. Therefore, we use StyleGAN to generate very rough colorings and to provide latent space specifically for these colorings. Reference [41] employs rough colorings for colorization and shows their effectiveness. The model for rough coloring generation is also trained with adversarial loss, so the constraint on color diversity remains. We overcome it with our approach.

## III. METHOD

In this work, we propose a method for automatic multimodal manga page colorization that supports modification with color hints (see Fig. 1). We use ideas from [8] and employ two neural networks for colorization: the drafting network and the colorization network. The first model utilizes pretrained StyleGAN to generate highly blurred images (coloring drafts) that set the overall color distribution of the image, and the colorization network, based on this coloring draft and the inputted color hints, generates a high-quality color image.

We prefer to colorize the entire page rather than individual panels for the following reason. Manga, especially modern manga, often have non-trivial shapes and arrangement of panels, which makes automatic splitting quite a challenge by itself. In addition, some pages contain overlapping panels, objects or speech balloons may overlap several panels at once, making splitting impossible. Meanwhile, the manga page has only two possible shapes, like a majority of other books (rectangle with a ''portrait'' and ''landscape'' orientation), and doesn't require such kind of preprocessing.

We train our model using paired data with synthetic black-and-white images generated from color ones and unpaired real black-and-white images. Training consists of two steps. In the first step, we train only the colorization network using synthetic data for supervised training and real data for adversarial domain adaptation. It means that we use the discriminator not only to distinguish between real color images and colorized synthetic images but also between the real color images and colorized real black-and-white images. Thus, the discriminator knowledge about color images is used to improve the generator performance on the distribution of real black-and-white images. In the second step, we use several snapshots of a well-trained colorization model to create the paired dataset for real black-and-white images by predicting colorings for them. In other words, we employ a self-training approach. Distinct colorings are used to prevent overfitting and to help the model learn how to extract information from drafts and color hints. In the end, we train the drafting network and colorization network independently in a supervised manner with synthetic data and real data with pseudolabels.

We intentionally avoid using an architecture that utilizes StyleGAN as a feature bank like GLEAN [42] for drafting network. Such architecture is end-to-end trained, which means that it is basically a cGAN and is subject to the same problems common to cGANs resulting from the training with a combination of supervised and adversarial losses. We experimented with the GLEAN architecture and found no improvement over a single colorization network. The conditional generation is more complex than the unconditional one since model weights must simultaneously fit all possible inputs, thus implicitly modeling the conditional distribution on the colorings and explicitly sampling from it. Thus, the estimated distribution is less expressive in comparison to the unconditional approaches. Our approach introduces two kinds of generation with two neural networks: the first consists of unconditional generation of coloring draft (unconditional in terms of color distribution since we apply style mixing with random CLS vector), the second is conditional generation based on coloring draft. We bypass the expressivity constraint since the result of conditional generation depends directly on the input coloring draft that is generated unconditionally. Colorization with a fixed coloring draft implies only a small amount of variation about itself, which greatly simplifies the implicit estimation of the coloring distribution.

We describe the architecture and loss functions of the drafting model and the colorization model, as well as the ideas that include in their design, in Section III-A and Section III-B correspondingly.

### A. DRAFTING NETWORK

During the training of the colorization network, which receives only color hints, we noticed that when there are few or no color hints, our model gives preference to a few colors when colorizing images of any style and content, which means it generates colors from a certain mode, ignoring all others. On the one hand, it worsens the automatic colorization of images, and on the other hand, it impedes training. We also observed that our model is able to change this color mode within a few hundred training iterations, depending on the input images. Such behavior is understandable since colorization is a multimodal task that implies a variety of acceptable outputs for a single input. Therefore, reducing the

**FIGURE 2.** Example of training data: black-and-white image, color image, color hint, coloring draft. The first two images represent black-and-white and color versions of the same manga page. The third image corresponds to a random color hint generated with the color image. The fourth image is a coloring draft generated with the color image.

set of acceptable solutions by setting a constraint using a coloring draft helps us to simplify the colorization task for the colorization network and to get a better model after training.

We do not expect the model to be able to generate high-quality images and, as will be shown later, it is not able to do that because the manga is quite complex data and it would be too audacious to expect that StyleGAN can build such a latent space that can generate scenes of arbitrary content. We want StyleGAN to generate something in between a color palette and a usual color image (see Fig. 3). We use StyleGAN to get a mapping between the pretrained StyleGAN latent space that has useful properties and is called Coloring Latent Space (CLS) and the space of coloring drafts. We can sample different colorings with the mapping network of StyleGAN and style mixing, thereby obtaining multimodal coloring sampling. We allow such sampling because we want the coloring draft to primarily affect the background color and intractable objects and to have little effect on objects that have a stable visual appearance in various manga, like characters' faces. Such a degree of freedom is acceptable because the artistic images do not have to represent the real world and admit large variation in the visual representation. Moreover, the use of style mixing increases the robustness of our approach. By choosing some vector and mixing it with the predicted CLS vectors, we can lock the color appearance of color drafts. This way, spatial color distribution for different views of the same page can be fastened by fixing the vector of StyleGAN latent space.

The use of blurred images is caused by the limitations of the neural GAN projection methods. If we train StyleGAN on sharper images, the drafting network method still generates the same blurred images. In this case, the latent vector corresponding to the coloring draft will be an outlier with respect to the distribution of the latent vectors for the training dataset. As a result, style mixing for the coloring draft does not work properly. In other words, we cannot reduce the blur level due to the limited capabilities of modern neural projection methods. A slight increase in blur level does not change the results, but a significant one worsens the performance because the coloring draft loses a lot of information, becomes almost a monochrome image, and can be replaced by a three-dimensional vector. To sum up, we choose the minimum blur level that can be reproduced by the projection method.

We use the Pixel2Style2Pixel [43] architecture for this model. Neural network receives a black-and-white manga image $X \in \mathbb{R}^{256\times256\times1}$ and outputs a coloring draft $S \in \mathbb{R}^{256\times256\times3}$. Thus, the model takes a manga image and maps it into the $\mathcal{W}^+$ [44] space of color manga images, meaning that it selects a vector that corresponds to the coloring that matches the input. Then the obtained vector is converted into a coloring draft using the pre-trained StyleGAN. We use the following loss for training:

$$\mathcal{L}_{Draft} = \lambda_{MSE}\mathcal{L}_{MSE} + \lambda_{LPIPS}\mathcal{L}_{LPIPS} \tag{1}$$

where $\mathcal{L}_{MSE}$ - mean squared error, $\mathcal{L}_{LPIPS}$ - LPIPS loss presented in [45].

We use a pixel level MSE loss to enforce the generator to produce similar colorings to the ground truth. LPIPS loss is applied to ensure similarity not only at the pixel level but also at the structural level.

### B. COLORIZATION NETWORK

Colorization network receives three objects: black-and-white image $X \in \mathbb{R}^{h\times w\times1}$, color hint $H \in \mathbb{R}^{h\times w\times4}$, and coloring draft $S \in \mathbb{R}^{h\times w\times3}$. It outputs the color image $\hat{Y} \in \mathbb{R}^{h\times w\times3}$ in RGB format. An example of a training sample is presented in Fig. 2. We use fixed values of $h$ and $w$ during training, but they may be arbitrary during inference since our model is fully convolutional.

Similarly to [38], we used the model proposed in [10] and modified it for our task. As a result, our model looks as follows:

#### 1) GENERATOR

The generator has a UNet-like architecture with the SE-ResNeXt50 [46], [47] as an encoder that is pretrained for tag prediction with line art anime data [31]. The other input branch is used to propagate color from a color hint and coloring draft. It is a small CNN that reduces the size of the concatenated image and color hint to match the size of the encoder output. In contrast to [10], we replaced

**FIGURE 3.** Color images and corresponding coloring drafts. The coloring draft looks like a blurred version of the color image.

ResNeXt blocks with SE-ResNext blocks in the decoder. We noticed that utilization of the Squeeze-and-Excitation operation increases the capacity and simplifies training.

#### 2) DISCRIMINATOR

We use the discriminator architecture from [10]. The only difference is that we apply spectral normalization [48] to convolutional layers. We have noticed that applying WGAN-GP loss results in colorizations that are not as colorful as those obtained with classic GAN loss, and its calculation requires more time and resources because of the gradient penalty. However, the classic adversarial loss function is quite unstable, resulting in artifacts in the image and even divergence, so we use spectral normalization as a compromise since it stabilizes the discriminator training and reduces generator loss weights sensitivity without any significant computational overhead.

We employ the method proposed in [10] to simulate color hints during training.

In the first step of training, we use adversarial adaptation approach since the synthetic data is quite different from the real data, and the model trained only with synthetic data colorizes the real ones poorly. However, the quality of the model on real data changes a lot during training, so we pick snapshots with better performance and generate pseudolabels for real data to use it in supervised training, which is considerably more stable. The performance of the snapshots is manually evaluated with human assistance. As a result, our loss function for the first step looks as follows:

$$\mathcal{L}_G = \lambda_{MAE}\mathcal{L}_{MAE} + \lambda_{per}\mathcal{L}_{per} + \lambda_{synthetic}\mathcal{L}_{G_{adv}}^{synthetic} + \lambda_{real}^{G}\mathcal{L}_{G_{adv}}^{real}, \tag{2}$$

$$\mathcal{L}_D = \mathcal{L}_{D_{adv}}^{synthetic} + \lambda_{real}^{D}\mathcal{L}_{D_{adv}}^{real}, \tag{3}$$

where $\mathcal{L}_{MAE}$ - mean absolute error (MAE), $\mathcal{L}_{per}$ - perceptual loss [49], $\mathcal{L}_{G_{adv}}$ and $\mathcal{L}_{D_{adv}}$ - classic adversarial loss [50]. The $\mathcal{L}_{G_{adv}}^{synthetic}$ and $\mathcal{L}_{D_{adv}}^{synthetic}$ are calculated for real color and synthetic black-and-white images, and the $\mathcal{L}_{G_{adv}}^{real}$ and $\mathcal{L}_{D_{adv}}^{real}$ are calculated for real color and real black-and-white images. The adversarial loss in the second step is calculated for synthetic and real data simultaneously. For a formal description of the loss functions, see AppendixC.

## IV. EXPERIMENTS

In this section, we describe the work with data and neural networks. In Section IV-A, we describe the process of building a training set: data collection, data processing, and diversification methods. Then in Section IV-B we describe the training process. In Section IV-C, the training results are presented. We give illustrative examples of our model's work to prove that our approach works. We perform a visual and quantitative comparison of the models to confirm the effectiveness of the proposed ideas in Section IV-D.

### A. DATASET

#### 1) DATA COLLECTION

We used web scraping to build our dataset. In total, approximately 82,000 images of color manga pages have been collected. Moreover, around 40000 color manga images from the Danbooru [51] dataset were employed. In order to improve the generalizability and perform model comparison, we have collected a lot of titles of different styles that have only the black-and-white version. They totaled about 55,000 images. For testing and comparison purposes, we used the Manga109 [52], [53]. It is the biggest public manga dataset that is composed of 109 manga volumes drawn by professional manga artists, which is around 21000 black-and-white manga page images. This dataset contains a lot of manga created in the 80s and 90s, which have significant style differences from the manga we utilize in training. So we use it to show how well our model generalizes and performs on unfamiliar manga styles.

#### 2) DATA MATCHING

We have collected a large number of pairs of black-and-white and color manga, but they were obtained from different sources. Therefore, they may contain different translations, fonts, and even drawings. Moreover, the images of compatible pages may be different because most manga are published on paper and then digitized, so the images have different sizes and indents from the edges. All of this leads to the fact that we cannot establish the pixel-by-pixel correspondence that is necessary to train the Pix2Pix model.

We tried to exploit the following fact to establish a pixel-by-pixel correspondence: the colored images have black borders that constitute the black-and-white image, so the correspondence can be established not between the images but only between their black pixels.

The following procedure was used to measure image correspondence:

1) Cut off several rows and columns of pixels from the edges of the color image.
2) Resize the obtained image to the size of black-and-white.
3) Build a black pixel mask based on a color image
4) Apply this mask to the black-and-white image.
5) Calculate the error as a deviation of pixels corresponding to the mask from the black.

Thus, we can build a match for black-and-white and color images whose content differs only by color. Unfortunately, our algorithm is not suitable for images with different content,

and most of the pairs we collected are exactly like that. We were only able to build a correspondence for the Demon Slayer manga images, and its colorized images are only about 1000, so we can't directly learn how to map black-and-white manga to color. However, the obtained pairs are used for fine-tuning and play an important role in model training.

#### 3) DATA GENERATION

Synthetic black-and-white images are generated from color ones to build a paired dataset for supervised training. However, we are not able to create a paired dataset by transforming color images to grayscale with a weighted sum of channels because the pixel distribution of manga images significantly differs from pixel distribution of color image channels, so a model trained with these data would poorly generalize on real black-and-white manga images. Therefore, we use the following sequence of preprocessing steps to generate plausible synthetic data:

1) We apply the xDoG [54] algorithm for edge detection to the color data. It generates more visually appealing images than classic methods like Canny edge detector or Sobel operator, and it's frequently used in line-art colorization. The algorithm extracts the borders and objects with high quality, but the obtained images are much more sparse than most manga styles therefore training with such data would result in poor performance on real data. We use parameters similar to [38].
2) We train and apply a neural network to map images generated with xDoG to real black-and-white images. To build a paired dataset, we apply xDoG to the output domain images. Then we use the trained model to transform the data generated in the first step. The predicted images are more similar to the real data but have noticeable differences at the pixel level. The architecture of the model is similar to the colorization network.
3) The CUT [55] model is applied to perform unidirectional domain translation from the domain of images generated at the previous step into the domain of real images. The model is trained on $64 \times 64$ crops to properly estimate the pixel distribution of the target domain images.

Since we are using a training dataset consisting of 100,000 images, the employment of data augmentations for diversification is beneficial. To prevent overfitting with image border margins, we apply horizontal flipping and random cropping. Since the number of gray pixels in the image depends on the style and varies a lot, but it does not change much in synthetic data, we also apply color jitter augmentation.

For data description and details of data augmentation, see Appendix B.

### B. TRAINING

The models are implemented with PyTorch [56] library for Python. The optimization is performed with Adam [57] because it has fast convergence in most cases and it's especially well-suited for GAN training. Image augmentations are performed with the Albumentations [58]. We use a device with NVIDIA Tesla A100 80Gb for computing. We employ the following training process for our models:

#### 1) StyleGAN

We have trained this model for unconditional manga generation since we have to use a pretrained generator in the drafting network. All collected color manga images, which are about 120,000, were used. We limited the size of images to $256 \times 256$ since StyleGAN requires a lot of computational power. Several models were trained for comparison: StyleGAN2 with randomly initialized weights and parameters corresponding to the [13], StyleGAN2 pretrained with the Celeba HQ dataset with parameters corresponding to the [13], and StyleGAN2 and StyleGAN3 pretrained with the Danbooru dataset [51].

#### 2) DRAFTING NETWORK

We trained the drafting network for two tasks: mapping synthetic black-and-white images into a CLS vector that corresponds to the appropriate coloring draft, and GAN inversion, which means mapping color manga images into CLS. We chose the loss weights without profound reasoning because the network generates intermediate results, and its quality is difficult to evaluate. The $\lambda_{pr}$ parameter was chosen large enough to affect the result but did not impede the training. We also trained several models with different dimensionality of CLS to find out its influence on the result.

#### 3) COLORIZATION NETWORK

We use small batch size and short training duration since the model overfits with synthetic data even if we use real data along with it. Following the [59], the imbalanced learning rate is applied to equalize the number of weight updates for the generator and the discriminator. Coloring drafts for color images and self-training colorings are generated using the GAN inversion model rather than the drafting model to avoid overfitting. We performed a grid search to determine the optimal loss weights and found that the balance between the MAE and perceptual losses does not significantly affect the final result, although omitting one of them does. Meanwhile, the weight of the adversarial loss is quite sensitive, so its slight change may lead to a disturbance of the training balance between the generator and the discriminator resulting in unsuccessful training.

For training details and hyperparameters values, see Appendix D.

### C. RESULTS

#### 1) MANGA GENERATION WITH STYLEGAN

The first model with randomly initialized weights failed to learn our dataset. We have tried various combinations of hyperparameters, but training still led to a divergence or mode collapse. The second, third, and fourth models were able to converge, with the third and fourth models beginning to generate quality images much earlier than the second. The objects in the generated images still do not always have a sharp shape. However, the images, in general, are strongly reminiscent of color manga.

The obtained CLS supports style mixing. This way, it is possible to transfer the color scheme of one image to another. We calculated the FID metric to compare the quality of the models and it was equal to 8.3 for the second model, 3.6 for the third, and 4.1 for the fourth. Due to the worse results and the higher computational complexity of StyleGAN3, we decided to stick with StyleGAN2.

#### 2) DRAFTING NETWORK

We tried to train the model to map input images into vectors that correspond to color manga images. Yet, the resulting models generate images consisting of several panels that contain blurry color blobs. From the examples, it can be seen in Fig. 3 that the resultant images have a similar panel structure, but the complex patterns cause difficulties. The color blobs contained in the panels correspond to the dominant colors in the original image. We also used iterative projection methods, but they showed similar problems. Apparently, manga images are too complex data for existing methods, although the resulting models still can be useful.

#### 3) COLORIZATION NETWORK

The model exhibits a high-quality performance of automatic coloring without color hints for images of different manga styles. Also, the colorization does not depend on the aspect ratio of the manga page, so double-page spreads are colorized with the same quality as the standard pages.

Coloring draft influences the image color palette but does not change the coloring of objects, about which the model is strongly confident. In this way, it is possible to perform sampling to obtain a variety of colorings. Additionally, we can use color images as references. By mapping this image into CLS using a model performing GAN Inversion and executing style mixing, we can get a coloring based on the colors of the reference image.

Failure cases often correspond to cases where the manga style is quite unusual or scenes are fairly cluttered. The colorization of an object can be inhomogeneous if it is overlapped by some lines. A manga consists of many lines, where the line can be either a minor background or part of an important object, so object detection for manga is quite challenging.

A large number of manga generation and colorization examples are provided in the AppendixE.

### D. MODEL COMPARISON

Two datasets are used for the comparison. Firstly, we employ a holdout set consisting of 5000 randomly selected black-and-white images of the collected dataset, which are not used for domain adaptation and self-training. Secondly, we use the Manga109 dataset, which contains manga titles that are not presented in the first dataset. Thus, the image style of the first dataset is familiar to the models, as other images sharing these styles were used in training explicitly or implicitly. In contrast, images of the second dataset have styles previously unseen by models, so we use them to estimate the degree of performance degradation with the introduction of new drawing styles.

We utilize existing methods of image translation and manga colorization to show the superiority of our approach in automatic colorization. Quantitative and qualitative (unless otherwise specified) comparisons are performed in fully automatic mode, i.e., without user input like color hints. We train Pix2Pix [18] on a synthetic paired dataset. AlacGAN [10] and the model from [38] are trained according to the procedures described in the corresponding papers. Style2Paint [8] and ScreenStyle [36] have no training code in the public domain and we were unsuccessful in obtaining it from the authors, so we use shared weights of models pretrained on other data. We understand that such utilization of these methods introduces certain unfairness into comparison, but we believe that it is still fair enough to show the superiority of our approach. To begin with, the training process of these models is sophisticated and contains many obscure details, so an inaccurate reimplementation would also lead to unfair comparison. Since the Manga109 dataset is used for numerical and visual comparison, it provides a domain shift for all models (except ScreenStyle). This way, we compare models fairer by testing on unseen images. In fact, Style2Paints was designed for conditional generation requiring color hints to generate reasonable coloring. We use it to show how our model outperforms the conditional methods if there are no color hints or an extremely small number of them. Style2Paints was trained on a much larger dataset of anime-styled images, which also contains manga images, so we think that the superiority of our method in the considered condition of a small amount of user input is shown quite fairly. Regarding ScreenStyle, we used the weights of this model that was trained on Manga109, which means that the dataset used for comparison is basically a training dataset for this model. So the comparison is unfair against our model, which still outperforms ScreenStyle due to its high generalization ability.

We do not employ for the comparison the recently proposed approaches that strongly rely on user input [24], [26], [27] since they have the same problems with automatical colorization as Style2Paints. Moreover, we do not perform the comparison with recent methods [25], [28] that provide incremental improvements to [10] because those improvements is negligible regarding the domain gap that affects the models

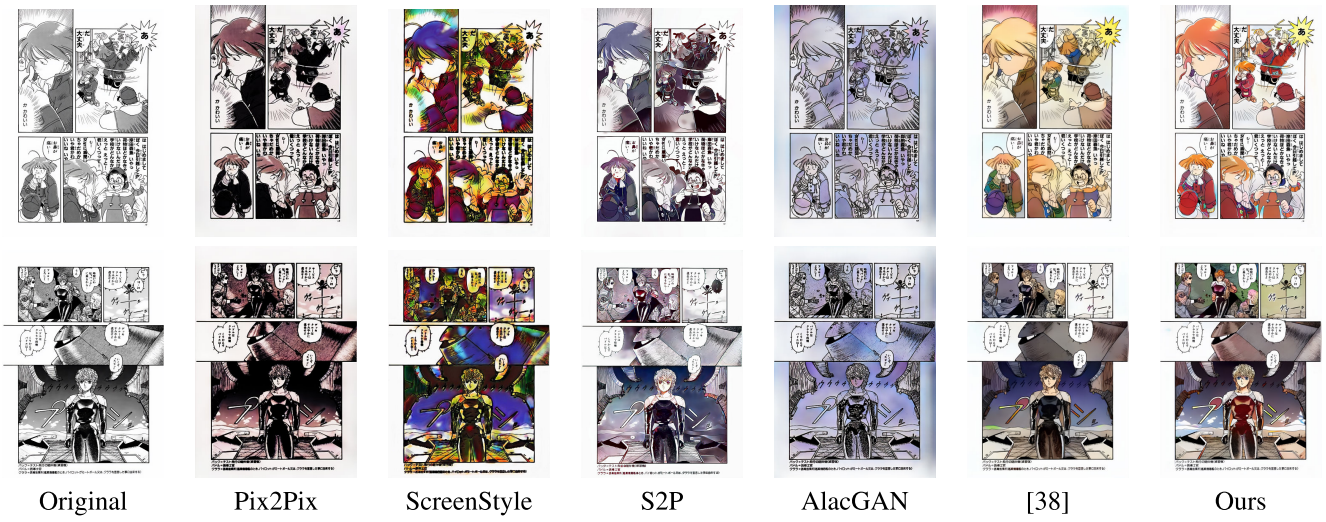


**FIGURE 4.** Visual comparison of the existing colorization methods. Our model generates the most plausible coloring.

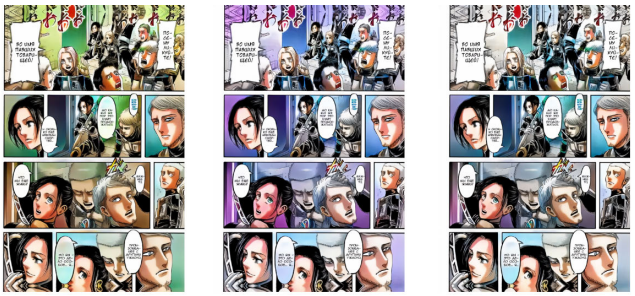

**FIGURE 5.** Colorization sampling. Three colorizations with the same characters colorization but different background colors (green, violet, blue).

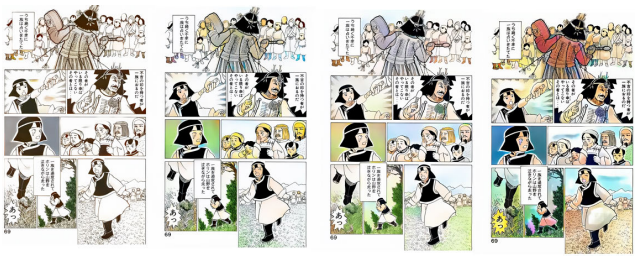

**FIGURE 6.** The influence of domain adaptation: without domain adaptation, + improved data generation (steps 2 and 3), + adversarial domain adaptation, + self-training. The first image is almost black-and-white. The second image has colorized characters' faces but black-and-white background. The third image is fully colorized, but the colors are pale. The fourth image is fully colorized with vivid colors.

**FIGURE 7.** Comparison of colorization with color hint. From left to right: color hint, Style2Paints, [38], our. The first colorization has regions that are not colorized if they miss color hint. The second colorization has all regions colorized, but some of them have improper colors. The third colorization has all regions are properly colorized.

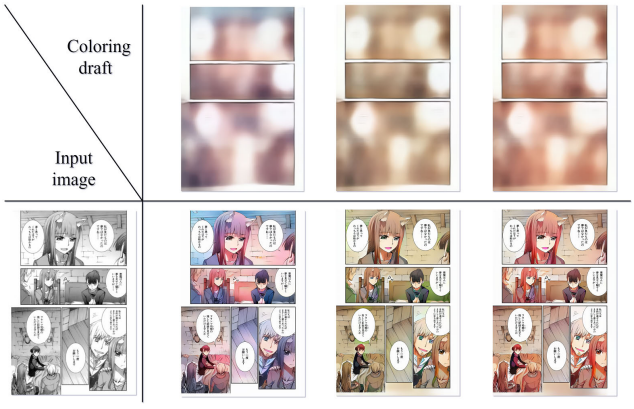


**FIGURE 8.** Correspondence between the coloring draft and the final coloring. Coloring draft affects color distribution of the colorization.

similarly to [10]. Therefore, we believe that a comparison with Style2Paints, AlacGAN, and [38] is the essential one.

The following comparison approaches are used:

- **Visual comparison**.
- **Fréchet inception distance**. FID evaluates the similarity between the distribution of generated images and the distribution of real images. It's a Wasserstein distance between two multidimensional Gaussian distributions of intermediate activations of pretrained InceptionV3. Parameters of these distributions are estimated with testing data. We consider FID an appropriate metric

**FIGURE 9.** Robustness comparison: original, [38], only colorization network, ours. The two input images represent the same page with a minor shift. The first colorization significantly changes in terms of texture and color distribution. The second colorization changes in terms of color distribution. The third colorization differs in details but preserve texture and color distribution.

**TABLE 1.** Model comparison with FID (lesser is better).

| Model | Our dataset | Manga109 |
|---|---|---|
| Pix2Pix [18] | 30.21 | 34.48 |
| AlacGAN [10] | 24.17 | 29.4 |
| ScreenStyle [36] | 23.16 | 22.12 |
| [38] | 19.48 | 21.02 |
| Our (without drafting network) | 17.00 | 20.59 |
| Our (with empty coloring draft) | 14.53 | 20.11 |
| Our (512 CLS) | **12.04** | **18.36** |
| Our (1024 CLS) | 12.10 | 18.52 |

**TABLE 2.** Model comparison with MOS (greater is better).

| Model | Our dataset | Manga109 |
|---|---|---|
| AlacGAN [10] | 1.7 | 1.6 |
| Style2Paints [8] | 2.8 | 2.6 |
| [38] | 3.3 | 3.0 |
| Our (512 CLS) | **3.9** | **3.2** |

**TABLE 3.** Ablation study for domain adaptation methods with FID (lesser is better).

| Model | Our dataset | Manga109 |
|---|---|---|
| No domain adaptation | 24.19 | 27.39 |
| +data generation step 2 | 17.03 | 20.74 |
| +data generation step 3 | 16.70 | 20.55 |
| +adversarial domain adaptation | 12.61 | 18.70 |
| +self-training (our) | **12.04** | **18.36** |

because we noticed during our experiments that its significant decrease coincides with an improvement in the quality of colorization concerning our perception.

- **Mean opinion score**. We randomly picked 50 colorized images from each dataset with a random replacement of images with inappropriate content and asked 20 people to rate the colorization quality using a five-point scale, where 1 represents ''Bad'' and 5 stands for ''Excellent''. The result is the mean of these votes. Respondents were not informed which model was used for colorization, so human' assessment was blinded.

**TABLE 4.** Inference time comparison (seconds).

| Model | Preprocessing | Colorization |
|---|---|---|
| Style2Paints | **1.94** | 8.17 |
| [38] | 3.52 | 0.73 |
| Our | 2.56 | **0.68** |

As can be seen in Table 1, Table 2 and Fig. 4, our modified architecture and training process produce a model that is capable of generating better quality images than other existing approaches. Pix2Pix and AlacGAN are unable to generate any coloring. Style2Paints generates coloring for objects filled with screentones, but barely colors objects without them because it heavily relies on user input. Screenstyle generates a certain coloring, but it is very coarse and has limited consistency with the image content. Model from [38] produces a fairly qualitative coloring, but our model recognizes objects in the image better, generates more varied colorings, and creates fine details, such as glares on the skin, that make the colorings more profound.

You can also notice that a model that was trained with coloring drafts but doesn't use them in the inference works better than a model that was trained without them. It shows that the use of coloring drafts simplifies training and allows the model to more effectively learn the proper color distribution. In addition, the table shows that using a model with a more qualitative StyleGAN trained with a larger CLS dimensionality does not lead to better results. This may be an indication of the fact that the enlarged CLS does not help to improve the performance of existing projection methods in dealing with manga images.

An example of multimodal colorization is presented in Fig. 5 and Fig. 8. The background is more affected when the color of the faces remains the same. The Table 3 shows the effectiveness of the applied domain adaptation approaches. The Fig. 6 exhibits the case when self-training significantly improves the quality of coloring. The comparison of the coloring with the color hint is illustrated in the Fig. 7. Our model is superior in colorization of regions not marked with color hints, thereby reducing the amount of user input and reducing the time for creating the desired coloring.

In Fig. 9, we exhibit that our approach is more robust than [38] and coloring network that preform colorization with empty color draft. We exclude other aproaches from comparison since they are not able to generate plausible colorings for manga. We use two images of the same page that are shifted and have different translation and pixel distribution since have been captured from distinct pages. Our model preserves color distribution over objects better than competitors, thus verifying the validity of our contribution.

We also compare the inference speed for models that support color hints. The comparison for inference of 512 × 768 images presented in Table 4. It was performed on a device with Core i5-8300H CPU and GeForce GTX 1060 GPU. We separate preprocessing and colorization because colorization with color hints involves single preprocessing

and iterative refinement via color hint modification. Ours performs colorization 10x faster than Style2Paints and slightly faster than [38]. Such computational complexity facilitates the utilization of our approach for AR applications, even for mobile ones.

## V. CONCLUSION

In this paper, we proposed an approach for robust multimodal manga page colorization with color hints using StyleGAN. We collected a dataset of the manually colored manga pages and used it with the proposed data synthesis approach to generate a paired dataset that highly resembles real black-and-white manga images and reduce the domain gap.

We determined that StyleGAN is able to learn the distribution of color manga images but its latent space is too complex for existing projection methods. However, even such space admits typical manipulations that can be used to construct a method for the generation of color cues that simplify the training and the inference of the colorization task. We also showed that a model that struggles less with task multimodality generates better colorings.

Our method is not very suitable for natural image colorization since the variety of colors of real-world objects is comparably small, e.g. a red banana seems to be unrealistic. Therefore, regularization of color variation is one of the key directions for future work.

## APPENDIX A
## APPENDIX OVERVIEW

In the appendix we present:

- Overview of the collected data and the applied augmentations (Appendix B)
- Formal definition of the training loss functions (Appendix C)
- Description of the training process and hyperparameters (Appendix D)
- Additional examples of manga page colorization (Appendix E)

## APPENDIX B
## DATA PREPARATION

### A. DATA COLLECTION

Traditionally, manga is produced with black ink, so most existing works are drawn in black-and-white. Color manga often exist either as a colorized edition of a popular black-and-white manga provided by a major publisher or as a colorization made by admirers, although there are some lesser-known manga that are executed in color initially. There are also many colored Korean and Chinese comics called Manhwa and Manhua. However, their drawing style and page structure are quite different from Japanese manga, so we decided to use them as little as possible.

We have collected about 82,000 images of the following color manga:

- Naruto
- One Piece
- Bleach
- Akira
- JoJo's Bizarre Adventure
- Demon Slayer
- Tales of Demons and Gods
- Genshin impact
- Another Emperor Reborn

To gather about 55000 real black-and-white images, we used the following black-and-white manga:

- Berserk
- Claymore
- Dr. Stone
- Hellsing
- Shape of voice
- Monster
- One-Punch Man
- Attack on Titan
- 5 Centimeters per Second
- The Monster Next to Me
- All You Need Is Kill
- Neon Genesis Evangelion
- Spice and Wolf
- Goodnight Punpun
- Phoenix
- The Ancient Magus' Bride

Danbooru [51] dataset is used for colored data diversification. It is a crowdsourced dataset consisting of anime illustrations created by hobbyist artists, where all images have a set of tags that describe them. We use 'colored', 'colorized', and 'comic' tags to select 40000 appropriate images.

### B. DATA AUGMENTATION

We use the following augmentations:

- Horizontal flipping with $p = 0.5$
- Random mirror padding with no more than 15 pixels, followed by random cropping of size $768 \times 512$. We have tried to train the model with $512 \times 512$ square images, but it resulted in degraded performance. Perhaps it is because the scenes near the image edges are cropped, which makes them difficult to understand and colorize. Thus, we train the model with images that have a 3:2 aspect ratio.
- Color jitter augmentation. In order to make our model overfit the synthetic data less strongly, we apply this augmentation to modify the brightness, contrast, and saturation of the input images. In addition, to ensure that the model does not overfit with the color features remaining in generated images, we apply this augmentation to the color images before preprocessing.
- For similar reasons with $p = 0.03$, we feed the network not with a synthetic image but with a grayscale image obtained from a color image with a weighted sum of the channels.

**FIGURE 10.** Images generated with StyleGAN without truncation trick. The images are easily distinguishable from the real manga pages but have their distinctive features. That is, StyleGAN learns page structure.

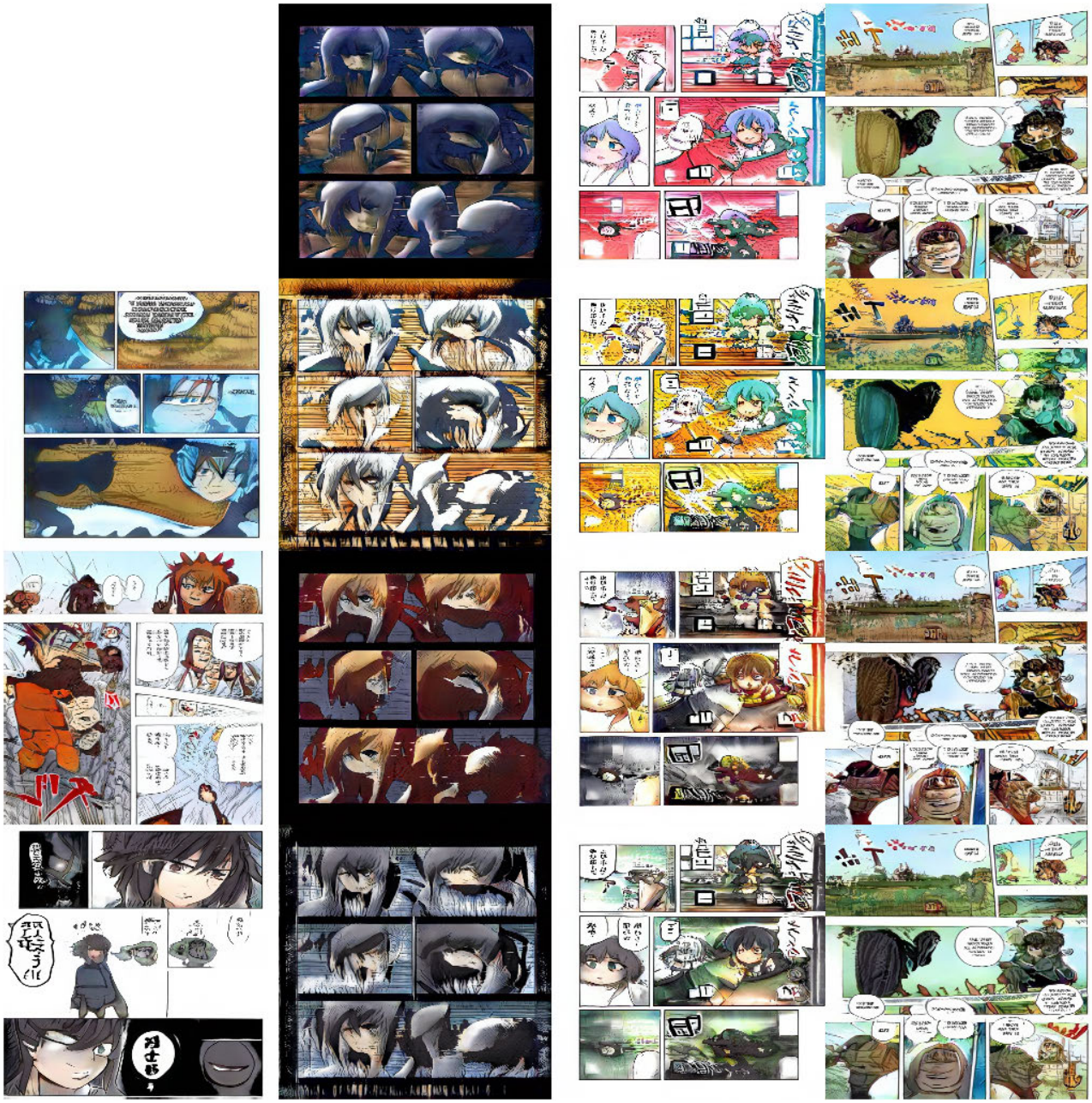

**FIGURE 11.** Style mixing of randomly sampled images. The structure of the page is preserved, but color distribution is modified in accordance with mixed image.

## APPENDIX C
## METHOD

In this section, we present mathematical formulations of loss functions used in training. To begin with, the following notation is introduced: synthetic black-and-white image $X_{syn} \in \mathbb{R}^{H\times W\times 1}$, real black-and-white image $X_{real} \in \mathbb{R}^{H\times W\times 1}$, resized synthetic black-and-white image $X_{res} \in \mathbb{R}^{256\times 256\times 1}$, color hint $H \in \mathbb{R}^{H\times W\times 4}$, and coloring draft $S \in \mathbb{R}^{256\times 256\times 3}$, ground truth color image $Y \in \mathbb{R}^{H\times W\times 3}$, resized ground truth color image $Y_{res} \in \mathbb{R}^{256\times 256\times 3}$, drafting network $\mathcal{S}$, coloring network (generator) $\mathcal{G}$, discriminator $\mathcal{D}$, $i$-th feature map of the pretrained VGG $\mathcal{V}_i$.

For the drafting network, loss functions are calculated as follows:

$$\mathcal{L}_{MSE} = \frac{1}{256\times 256\times 3}||\mathcal{S}(X_{res}) - Y_{res}||_2^2 \tag{4}$$

$$\mathcal{L}_{LPIPS} = \sum_l \frac{1}{H_l W_l}\sum_{h,w}||w_l \odot (\tilde{\mathcal{V}}_l(\mathcal{S}(X_{res}))^{hw} - \tilde{\mathcal{V}}_l(Y_{res})^{hw}))||_2^2, \tag{5}$$


**FIGURE 12.** Colorization of randomly picked images without color hints.

where $w_l \in \mathbb{R}^{C_l}$ - pretrained vector for the $l$-th feature map, $H_l$, $W_l$, $C_l$ - spatial dimensions of the $l$-th feature map, $\tilde{\mathcal{V}}_l(\cdot)$ - output of $\mathcal{V}_l(\cdot)$ unit-normalized in channel dimension, and $||\cdot||_2$ denotes Frobenius norm.

For the colorization network, loss functions are calculated as follows:

$$\mathcal{L}_{per} = \sum_l \frac{1}{C_l H_l W_l} ||\mathcal{V}_l(\mathcal{G}(X_{syn}, H, S)) - \mathcal{V}_l(Y)||_2^2, \quad (6)$$


**FIGURE 13.** Colorization of the double-page spreads. The double-page spread colorization has the same quality as for single-page images.

$$\mathcal{L}_{MAE} = \frac{1}{3HW}||\mathcal{G}(X_{syn}, H, S) - Y_{res}||_1 \tag{7}$$

$$\mathcal{L}_{G_{adv}}^{synthetic} = \log(1 - \mathcal{D}(\mathcal{G}(X_{syn}, H, S))), \tag{8}$$

$$\mathcal{L}_{G_{adv}}^{real} = \log(1 - \mathcal{D}(\mathcal{G}(X_{real}, \hat{H}, S))), \tag{9}$$

$$\mathcal{L}_{D_{adv}}^{synthetic} = -\log \mathcal{D}(Y) - \log(1 - \mathcal{D}(\mathcal{G}(X_{syn}, H, S))), \tag{10}$$

$$\mathcal{L}_{D_{adv}}^{real} = -\log \mathcal{D}(Y) - \log(1 - \mathcal{D}(\mathcal{G}(X_{real}, \hat{H}, S))), \tag{11}$$

where $\hat{H}$ - empty color hint.

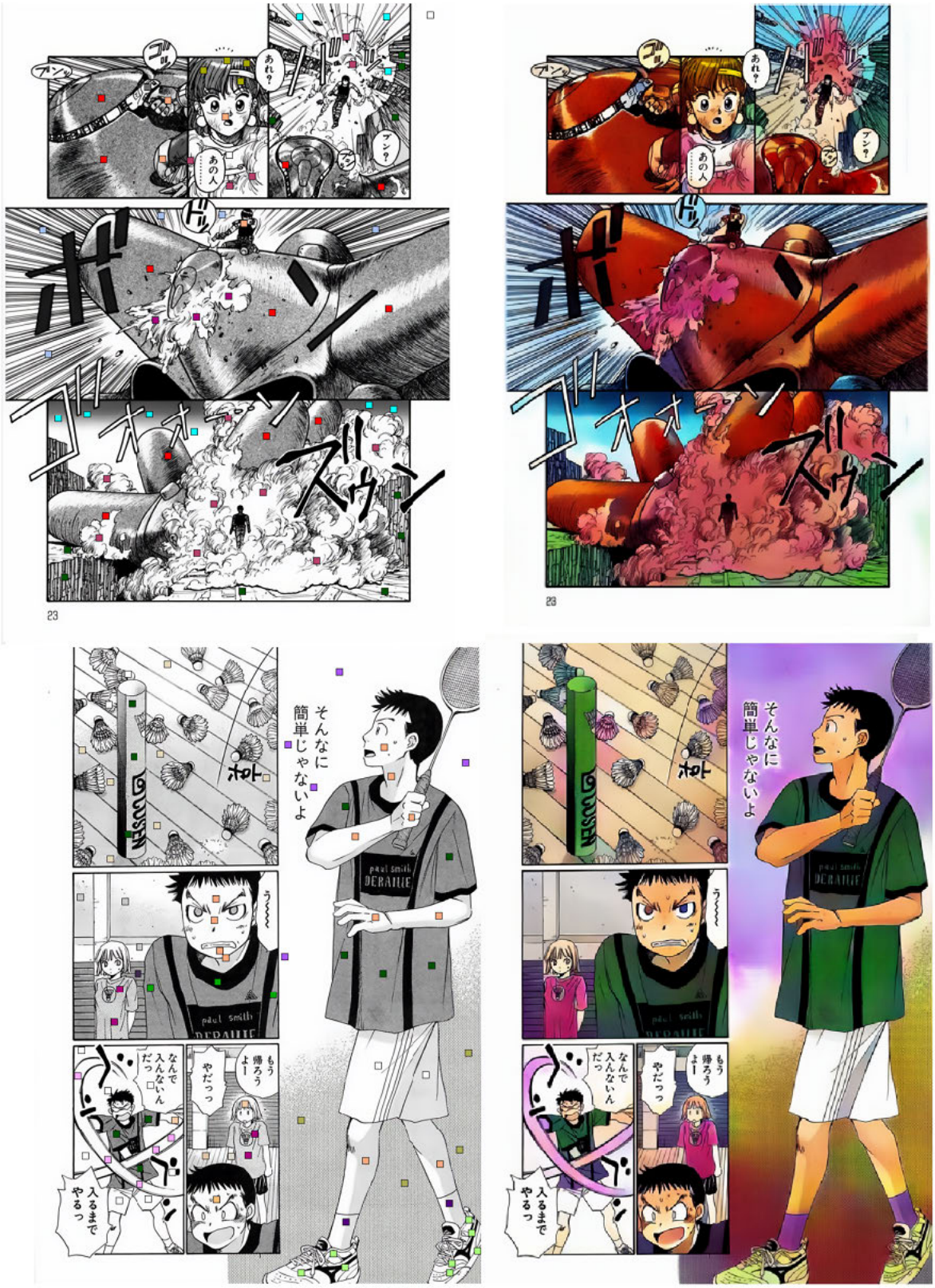


**FIGURE 14.** Colorization with color hints. The double-page spread colorization has the same quality as for single-page images.

## APPENDIX D
## TRAINING PROCESS

We initialize the weights with Xavier initialization and perform optimization with Adam [57] for all models.

**StyleGAN2**. We set the batch size to 32 and the learning rate to $10^{-3}$. The models are trained for 700,000 iterations with a gradual learning rate decrease.

**Drafting network**. We set parameters $\lambda_{L2}$, $\lambda_{LPIPS}$ to be equal 1 and 0.5 respectively. The models are trained for 12 epochs using synthetic data with the learning rate of $10^{-4}$ and the batch size of 8.

**Colorization network**. In the first step, we train this model for 15 epochs with a batch size of 4 and a learning rate of $10^{-4}$ and $4 \times 10^{-4}$ for the generator and discriminator. In the second step, batch size is 8 with the same learning rate. The loss weights are $\lambda_{L1} = 10$, $\lambda_{per} = 1$, $\lambda_{synthetic} = 1.5$, $\lambda^{G}_{real} = 0.4$, $\lambda^{D}_{real} = 0.7$ for the first step and $\lambda_{L1} = 10$, $\lambda_{per} = 1$, $\lambda_{synthetic} = 1.2$ for the second.

We use the following color cues generation schedule to maintain a training balance between the generator and the discriminator:

1) At the first epoch, we generate color hints with the probability of 0.3, the model gets an empty color hint otherwise; coloring draft is empty.
2) At the second epoch, the probability of color hints generation is 0.6, coloring draft is empty.
3) On the third, the probability of color hints generation is 0.3, and the probability of coloring draft generation is 0.4.
4) On the fourth epoch and beyond, the probability of color hints generation is 0.5, coloring draft - 0.8.

We use this scheme because if the generator receives color cues too often, it is easy to generate realistic images. Therefore, if we do not reduce their proportion, the discriminator will certainly lose to the generator and will not be able to learn useful features that will improve the generator performance.

In the first step, we fine-tune the model with a paired dataset of real data after training. We perform training for no more than two epochs, as the model is overfitting for a certain style of black-and-white manga. Such fine-tuning diversifies the model's colorings since the model is trained in supervised mode with pixel structures that are not present in data. It does not happen during training because the model is only trained with real images using adversarial loss.

## APPENDIX E
## RESULTS

Fig. 10 and Fig. 11 give examples of manga page images generated with StyleGAN. To exhibit the results of colorization without color hints, we randomly selected a set of images with random replacement if the image content is not appropriate (e.g., nudity). The results are presented in Fig. 12. We also demonstrate in Fig. 13 that the capability of our model to colorize double-page images. Colorization with color hints is illustrated in Fig. 14.

## REFERENCES

[1] Z. Cheng, Q. Yang, and B. Sheng, "Deep colorization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 415–423.
[2] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be color! Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–11, Jul. 2016.
[3] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Computer Vision—ECCV*. Amsterdam, The Netherlands: Springer, 2016, pp. 649–666.
[4] R. Zhang, J.-Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, and A. A. Efros, "Real-time user-guided image colorization with learned deep priors," 2017, *arXiv:1705.02999*.
[5] E. Kim, S. Lee, J. Park, S. Choi, C. Seo, and J. Choo, "Deep edge-aware interactive colorization against color-bleeding effects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14647–14656.
[6] M. Golyadkin, V. Pozdnyakov, L. Zhukov, and I. Makarov, "SensorSCAN: Self-supervised learning and deep clustering for fault diagnosis in chemical processes," *Artif. Intell.*, vol. 324, Nov. 2023, Art. no. 104012.
[7] X. Li, I. Makarov, and D. Kiselev, "Predicting molecule toxicity via descriptor-based graph self-supervised learning," *IEEE Access*, vol. 11, pp. 91842–91849, 2023.
[8] L. Zhang, C. Li, T.-T. Wong, Y. Ji, and C. Liu, "Two-stage sketch colorization," *ACM Trans. Graph.*, vol. 37, no. 6, pp. 1–14, Dec. 2018.
[9] L. Zhang, Y. Ji, X. Lin, and C. Liu, "Style transfer for anime sketches with enhanced residual U-net and auxiliary classifier GAN," in *Proc. 4th IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nov. 2017, pp. 506–511.
[10] Y. Ci, X. Ma, Z. Wang, H. Li, and Z. Luo, "User-guided deep anime line art colorization with conditional adversarial networks," in *Proc. 26th ACM Int. Conf. Multimedia*, 2018, pp. 1536–1544.
[11] C. Furusawa, K. Hiroshiba, K. Ogaki, and Y. Odagiri, "Comicolorization: Semi-automatic Manga colorization," in *Proc. SIGGRAPH Asia Tech. Briefs*, 2017, pp. 1–4.
[12] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4396–4405.
[13] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8107–8116.
[14] S. Guadarrama, R. Dahl, D. Bieber, M. Norouzi, J. Shlens, and K. Murphy, "PixColor: Pixel recursive colorization," 2017, *arXiv:1705.07208*.
[15] M. Kumar, D. Weissenborn, and N. Kalchbrenner, "Colorization transformer," 2021, *arXiv:2102.04432*.
[16] A. Van Den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1747–1756.
[17] J. Ho, N. Kalchbrenner, D. Weissenborn, and T. Salimans, "Axial attention in multidimensional transformers," 2019, *arXiv:1912.12180*.
[18] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5967–5976.
[19] Y. Xiao, P. Zhou, Y. Zheng, and C.-S. Leung, "Interactive deep colorization using simultaneous global and local inputs," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 1887–1891.
[20] M. He, D. Chen, J. Liao, P. V. Sander, and L. Yuan, "Deep exemplar-based colorization," *ACM Trans. Graph.*, vol. 37, no. 4, pp. 1–16, Aug. 2018.
[21] P. Lu, J. Yu, X. Peng, Z. Zhao, and X. Wang, "Gray2ColorNet: Transfer more colors from reference image," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 3210–3218.
[22] Y. Hati, G. Jouet, F. Rousseaux, and C. Duhart, "PaintsTorch: A user-guided anime line art colorization tool with double generator conditional adversarial network," in *Proc. Eur. Conf. Vis. Media Prod.*, Dec. 2019, pp. 1–10.
[23] F. C. Silva, P. André Lima de Castro, H. R. Júnior, and E. C. Marujo, "Mangan: Assisting colorization of Manga characters concept art using conditional GAN," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 3257–3261.
[24] L. Zhang, C. Li, E. Simo-Serra, Y. Ji, T.-T. Wong, and C. Liu, "User-guided line art flat filling with split filling mechanism," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9884–9893.
[25] M. Yuan and E. Simo-Serra, "Line art colorization with concatenated spatial attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 3941–3945.

[26] Y. Shimizu, R. Furuta, D. Ouyang, Y. Taniguchi, R. Hinami, and S. Ishiwatari, "Painting style-aware Manga colorization based on generative adversarial networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 1739–1743.

[27] J. Zhang, S. Zhu, K. Liu, and X. Liu, "UGSC-GAN: User-guided sketch colorization with deep convolution generative adversarial networks," *Comput. Animation Virtual Worlds*, vol. 33, no. 1, p. e2032, Jan. 2022.

[28] Z. Dou, N. Wang, B. Li, Z. Wang, H. Li, and B. Liu, "Dual color space guided sketch colorization," *IEEE Trans. Image Process.*, vol. 30, pp. 7292–7304, 2021.

[29] 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. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5800–5809.

[30] Q. Zhang, B. Wang, W. Wen, H. Li, and J. Liu, "Line art correlation matching feature transfer network for automatic animation colorization," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3871–3880.

[31] H. Kim, H. Y. Jhoo, E. Park, and S. Yoo, "Tag2Pix: Line art colorization using text tag with SECat and changing loss," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9055–9064.

[32] C. Zou, H. Mo, C. Gao, R. Du, and H. Fu, "Language-based colorization of scene sketches," *ACM Trans. Graph.*, vol. 38, no. 6, pp. 1–16, Dec. 2019.

[33] R. Cao, H. Mo, and C. Gao, "Line art colorization based on explicit region segmentation," *Comput. Graph. Forum*, vol. 40, no. 7, 2021, pp. 1–10.

[34] K. Frans, "Outline colorization through tandem adversarial networks," 2017, *arXiv:1704.08834*.

[35] S. Yoo, H. Bahng, S. Chung, J. Lee, J. Chang, and J. Choo, "Coloring with limited data: Few-shot colorization via memory augmented networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11275–11284.

[36] M. Xie, C. Li, X. Liu, and T.-T. Wong, "Manga filling style conversion with screentone variational autoencoder," *ACM Trans. Graph.*, vol. 39, no. 6, pp. 1–15, Dec. 2020.

[37] S.-Y. Chen, J.-Q. Zhang, L. Gao, Y. He, S. Xia, M. Shi, and F.-L. Zhang, "Active colorization for cartoon line drawings," *IEEE Trans. Vis. Comput. Graphics*, vol. 28, no. 2, pp. 1198–1208, Feb. 2022.

[38] M. Golyadkin and I. Makarov, "Semi-automatic Manga colorization using conditional adversarial networks," in *Analysis of Images, Social Networks and Texts*. Berlin, Germany: Springer, 2021, pp. 230–242.

[39] X. Luo, X. Zhang, P. Yoo, R. Martin-Brualla, J. Lawrence, and S. M. Seitz, "Time-travel rephotography," *ACM Trans. Graph.*, vol. 40, no. 6, pp. 1–12, Dec. 2021.

[40] Y. Wu, X. Wang, Y. Li, H. Zhang, X. Zhao, and Y. Shan, "Towards vivid and diverse image colorization with generative color prior," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14357–14366.

[41] Y.-J. Wei, T.-T. Wei, T.-Y. Kuo, and P.-C. Su, "Two-stage pyramidal convolutional neural networks for image colorization," *APSIPA Trans. Signal Inf. Process.*, vol. 10, no. 1, p. e15, 2021.

[42] K. C. K. Chan, X. Wang, X. Xu, J. Gu, and C. C. Loy, "GLEAN: Generative latent bank for large-factor image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14240–14249.

[43] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: A styleGAN encoder for image-to-image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2287–2296.

[44] R. Abdal, Y. Qin, and P. Wonka, "Image2StyleGAN: How to embed images into the StyleGAN latent space?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4431–4440.

[45] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.

[46] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5987–5995.

[47] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[48] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," 2018, *arXiv:1802.05957*.

[49] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision—ECCV*. Amsterdam, The Netherlands: Springer, 2016, pp. 694–711.

[50] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[51] A. Community and G. Branwen. (2021). *Danbooru2020: A Large-Scale Crowdourced and Tagged Anime Illustration Dataset*. [Online]. Available: https://www.gwern.net/Danbooru2020

[52] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, "Sketch-based Manga retrieval using Manga109 dataset," *Multimedia Tools Appl.*, vol. 76, no. 20, pp. 21811–21838, Oct. 2017.

[53] K. Aizawa, A. Fujimoto, A. Otsubo, T. Ogawa, Y. Matsui, K. Tsubota, and H. Ikuta, "Building a Manga dataset 'Manga109' with annotations for multimedia applications," *IEEE MultimediaMag.*, vol. 27, no. 2, pp. 8–18, Apr./Jun. 2020.

[54] H. Winnemöller, J. E. Kyprianidis, and S. C. Olsen, "XDoG: An extended difference-of-Gaussians compendium including advanced image stylization," *Comput. Graph.*, vol. 36, no. 6, pp. 740–753, Oct. 2012.

[55] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu, "Contrastive learning for unpaired image-to-image translation," in *Computer Vision—ECCV*. Glasgow, U.K.: Springer, 2020, pp. 319–345.

[56] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, and L. Antiga, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.

[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[58] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, p. 125, Feb. 2020.

[59] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7354–7363.

**MAKSIM GOLYADKIN** received the master's degree in applied mathematics and informatics and the Ph.D. degree in computer science from HSE University. He is currently a Researcher in the field of graph neural networks applications to the industrial AI with the Artificial Intelligence Research Institute (AIRI), Moscow, Russia. His contributions include initial idea, model and experiment design, and paper preparation.

**ILYA MAKAROV** received the Specialist degree in mathematics from Lomonosov Moscow State University, Moscow, Russia, and the Ph.D. degree in computer science from the University of Ljubljana, Ljubljana, Slovenia.

Since 2011, he has been a Lecturer with the School of Data Analysis and Artificial Intelligence, HSE University, where he was the School Deputy Head, from 2012 to 2016, and is currently an Associate Professor and a Senior Research Fellow. He was the Program Director of the BigData Academy MADE from VK, and a Researcher with the Samsung-PDMI Joint AI Center, St. Petersburg Department, V. A. Steklov Mathematical Institute, Russian Academy of Sciences, Saint Petersburg, Russia. He is also a Senior Research Fellow with the Artificial Intelligence Research Institute (AIRI), Moscow, where he leads the research in industrial AI. He became the Head of the AI Center and the Data Science Tech Master Program in NLP, National University of Science and Technology MISIS. His contributions include paper revision, help with experiment and model design, and research supervision.