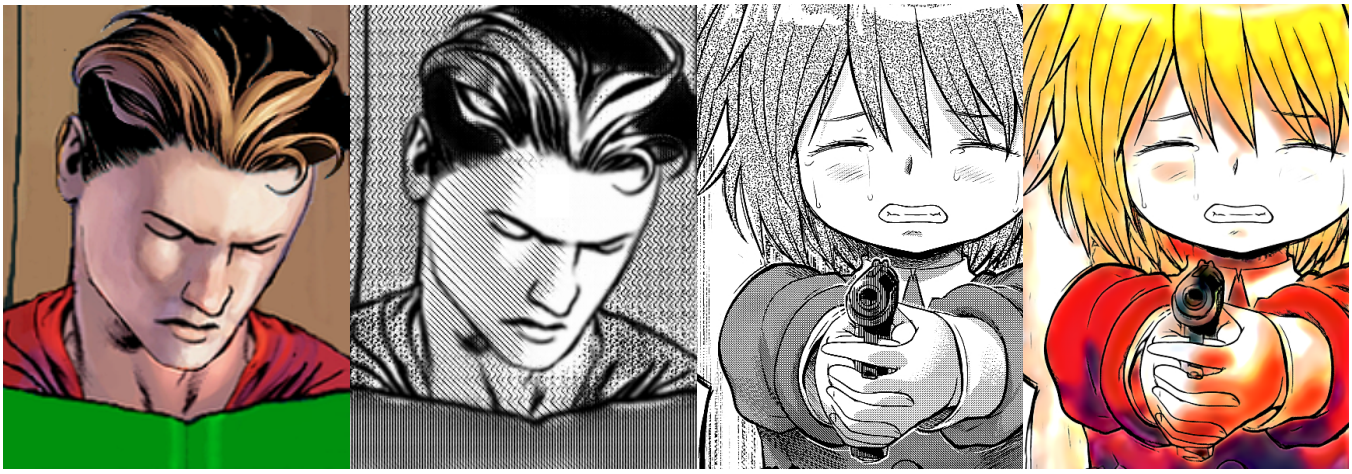
Manga Filling Style Conversion with Screentone Variational Autoencoder

MINSHAN XIE*, The Chinese University of Hong Kong
CHENGZE LI*, The Chinese University of Hong Kong
XUETING LIU, Caritas Institute of Higher Education
TIEN-TSIN WONG, The Chinese University of Hong Kong



(a) Western comic (b) Our screened comic (c) Japanese manga (d) Our colorized manga

Fig. 1. Our method automatically translates the filling styles between the color comic and screened manga. (c) KaerimichiNoMajo ©A-10

Western color comics and Japanese-style screened manga are two popular comic styles. They mainly differ in the style of region-filling. However, the conversion between the two region-filling styles is very challenging, and manually done currently. In this paper, we identify that the major obstacle in the conversion between the two filling styles stems from the difference between the fundamental properties of screened region-filling and colored region-filling. To resolve this obstacle, we propose a screentone variational autoencoder, ScreenVAE, to map the screened manga to an intermediate domain. This intermediate domain can summarize local texture characteristics and is interpolative. With this domain, we effectively unify the properties of screening and color-filling, and ease the learning for bidirectional translation between screened manga and color comics. To carry out the bidirectional translation, we further propose a network to learn the translation between the intermediate domain and color comics. Our model can generate quality screened manga given a color comic, and generate color comic that retains the original screening intention by the bitonal manga artist. Several results are shown to demonstrate the effectiveness and convenience of the proposed method. We also demonstrate how the intermediate domain can assist other applications such as manga inpainting and photo-to-comic conversion.

CCS Concepts: • **Applied computing → Fine arts**.

Additional Key Words and Phrases: Manga production, Screentone, Image-to-image translation, Variational Auto-Encoder

ACM Reference Format:
Minshan XIE*, Chengze LI*, Xueting LIU, and Tien-Tsin WONG. 2020. Manga Filling Style Conversion with Screentone Variational Autoencoder. *ACM Trans. Graph.* 39, 6, Article 226 (December 2020), 15 pages. https://doi.org/10.1145/3414685.3417873

* Equal contribution.
Authors' addresses: Minshan XIE*, The Chinese University of Hong Kong, msxie@cse.cuhk.edu.hk; Chengze LI*, The Chinese University of Hong Kong, czli@cse.cuhk.edu.hk; Xueting LIU, Caritas Institute of Higher Education, tliu@cihe.edu.hk; Tien-Tsin WONG, The Chinese University of Hong Kong, ttwong@cse.cuhk.edu.hk.

0730-0301/2020/12-ART226 $15.00
https://doi.org/10.1145/3414685.3417873

1 INTRODUCTION

Comics is a worldwide popular visual art form. While Japanese-style manga (Fig. 1(c)) is mainly produced in black-and-white (B/W), western comics (Fig. 1(a)) are commonly produced in color, due to the printing cost consideration, as well as cultural acceptance. Localization (colorization or decolorization) sometimes may be needed when the comic is publicized in non-originated countries. For example, the bitonal manga *Akira* was later reproduced in color version

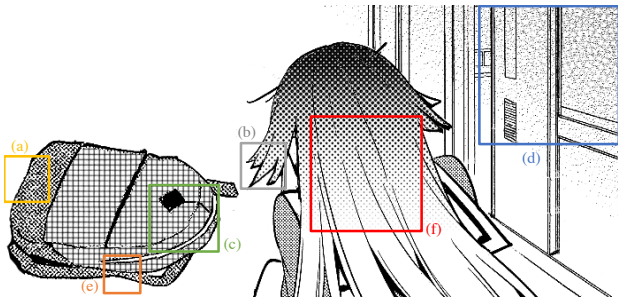


Fig. 2. Screened manga imposes many challenges to traditional texture analysis, especially at (a) region boundary, (b)&(e) narrow structure, (c) overlapping structure, (d) area across multiple objects, and (f) gradually changed regions.

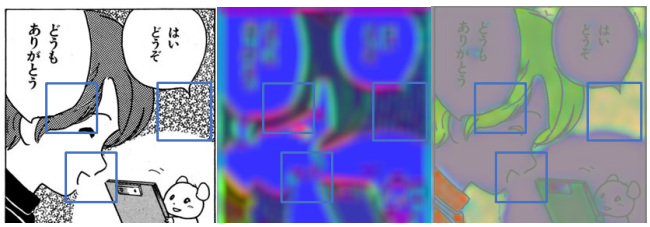


Fig. 3. Visual comparison between our ScreenVAE map and traditional texture analysis. (a) Screened manga, (b) Gabor Wavelet features (3 major components), (c) Our ScreenVAE map (3 major components). TapkunNo-Tanteisitsu ©Hukuyama Kei, from the Manga109 dataset [Matsui et al. 2017]

and sold in western countries[1]. The classic color comic *Tintin* is also produced in both color version and B/W version aiming for readers with different cultural backgrounds. However, the productions of color and B/W comics are substantially different, and thus the conversion is usually manual, costly and non-trivial. This is why such conversion was only done for best-selling comics/manga. A key difference between color and B/W productions is on how regions are filled. While regions of color comics are usually filled with flat or smoothly changing colors, regions of B/W manga are usually filled with rich screentone patterns to compensate for the lack of color (Fig. 2).

In this paper, we study the feasibility of automatic bidirectional conversion of region-filling styles between screening (screentone-filling) and color-filling, so that localization and electronic migration can be significantly automated. However, the fundamental properties of colors and screentones are significantly different. While a color is characterized by a single pixel, a screentone is characterized by a local neighborhood of pixels. This is why analyzing a screentone is usually based on a local window, but usually fails at region boundary (Fig. 2(a)), narrow structure (Fig. 2(b)&(e)), screened area with overlapping structure (Fig. 2(c)), and screened area across multiple objects (Fig. 2(d)). In contrast, humans can unconsciously identify these screened regions regardless of the interference of region boundary, narrow structure, or overlapping fine details. Even when the screentone is gradually changed (interpolated) over a larger spatial region as in Fig. 2(f), human vision can easily link them up together. This suggests we need to design a more sophisticated feature that resembles the human vision, so that a screentone can also be characterized at a single pixel and interpolative to facilitate the conversion between screening and color-filling.

Existing works in comic filling style conversion are mainly unidirectional, mostly solving the easier problem from color to screentone [Pang et al. 2008; Qu et al. 2008; Ulichney 1987]. Existing learning-based methods [Johnson et al. 2016; Liu et al. 2017; Zhu et al. 2017a] may also be adopted, but the results are usually unsatisfactory, due to the lack of consideration on the fundamental difference between color and screentone.

To unify the property of screentone to that of color and resemble the ability of human perception, we propose a novel learning-based model, *screentone variational autoencoder* (ScreenVAE), to summarize the texture characteristic of the local neighborhood into an intermediate representation. Unlike traditional window-based texture analyzer such as Gabor wavelet (Fig. 3(b)), our ScreenVAE is not interfered by boundary or overlapping fine details, and propagates through narrow structures (Fig. 3(c)). More importantly, just like the color, our ScreenVAE feature is *interpolative* as demonstrated by the generated interpolation of multiple substantially different screentones in Fig. 14.

With ScreenVAE, screentone and color are unified, and this facilitates the subsequent filling style conversion between western comics and Japanese-style manga. To convert, we train another bidirectional cycle-consistent neural network in an unsupervised manner, requiring no paired training data. The trained model can translate between color and screentone with consistency, i.e. the same screentone is consistently translated to the same color, and vice versa. This is especially useful for translating not only a single panel, but also a whole comic book. To achieve better results, we also design tailored objectives and adopt the adversarial learning to regularize the output comics. While this framework is currently trained for the conversion between screening and color-filling, it can also be trained to convert between two screening styles, or between two color-filling styles. To validate the effectiveness of our method, we apply our filling style conversion on various real-world comics of both western and manga styles. Convincing results are obtained. With our ScreenVAE model, manga inpainting is also simplified and achievable by performing the state-of-the-art learning-based inpainting on our generated ScreenVAE feature map, instead of on the original manga image. Satisfactory inpainting results are demonstrated. Our contributions can be summarized as follows.

- We propose a novel variational model, *ScreenVAE*, to characterize the local texture property at a single point, without the interference of boundary and overlapping fine details, as an interpolative ScreenVAE feature map.
- With the ScreenVAE unifying the property of screentone and color, we propose to learn and convert between screening and color-filling styles.
- The proposed ScreenVAE effectively simplifies the complex patterns in manga, and assists manga inpainting effectively.

[1]https://www.pigboom.com/steve-oliff-coloring-akira-manga/

2 RELATED WORK

Before introducing our method, we first review and discuss some existing works in the related fields.

Texture analysis. Texture analysis is a common technique analyzing screentone patterns. It summarizes local features to simpler representations to facilitate the identification of regions or objects in images, and combines with learning algorithms to perform required tasks in the feature domain, such as texture classification [Randen and Husoy 1999] and texture segmentation [Jain and Farrokhnia 1990; Liu and Wang 2006]. However, these texture features are all windowed-based and usually fail at boundary, thin structure, and overlapping structure. In comparison, our ScreenVAE can tackle these limitations with tailored design (Fig. 3).

Recently, convolutional neural network (CNN) has been demonstrated as a powerful feature extractor for texture analysis [Andrearczyk and Whelan 2016; Cimpoi et al. 2016]. Some research studies use low-level image cues and CNN-based descriptors to make region suggestions and facilitate tasks including texture classification [Cimpoi et al. 2014], texture segmentation [Cimpoi et al. 2015], and semantic segmentation [Cimpoi et al. 2016]. However, all these methods could only extract texture features for certain image types with tailor-made models, and usually fail for bitonal screened manga input. Although there exist research targeting for removing screentones from manga via CNN [Li et al. 2017a], the proposed method did not study how screentones are used for filling the regions. Differently, our method attempts to learn how regions are screened, and then generate a summarized intermediate representation based on the screening style.

Comic style conversion. A few attempts have been made to tackle a unidirectional conversion of the filling style. To translate color images to screentoned manga, one may adopt halftoning [Jarvis et al. 1976; Pang et al. 2008; Ulichney 1987] or hatching [Winkenbach and Salesin 1994] to shade the image regions. But the primary goal of these methods is to reproduce the intensity tone with a single kind of screentone. The results are unlike manga due to the lack of screentone variation and content-aware filling strategy. Qu et al. [2008] matched user-specified screentones with clustered color segments based on the similarity between screentones and colors to preserve visual richness. However, this method is highly dependent on the accuracy of image segmentation and user-specified screentone set. It is also unable to generate smooth transitions between two screentones for smoothing-changing color input. In comparison, we propose to directly project the dense color distribution in color comics to the bitonal screentone patterns. With our generated ScreenVAE map, the screentones can be spatially interpolated in terms of both type and tonal intensity. In addition, our method is fully automatic with content-awareness.

For translating B/W images into color comics, image colorization techniques [Levin et al. 2004; Sỳkora et al. 2004] may be applied. However, these methods mainly rely on the grayness continuity for smooth colorization, but screentones are bitonal in nature, which dissatisfies the grayness continuity assumption. Qu et al. [2006] proposed to propagate colors relying on pattern-continuity. However, it requires intensive user interaction for providing the color hints. Besides, the colorization is usually stuck at region boundary and thin structure, due to the inability of window-based texture analysis. In comparison, our method achieves automatic filling style conversion without any user hint. More importantly, our method provides a unified framework for bidirectional comic style conversion.

Learning-based image translation. Deep learning has been demonstrated effective in solving image-to-image translation [Isola et al. 2017]. It has a high potential for comic filling style translation. However, it usually requires a very large number of annotated training images, but paired data of manga and color comics is usually unavailable. Style transfer is a technique that aims at rendering images with the desired artistic styles [Gatys et al. 2016; Huang and Belongie 2017; Johnson et al. 2016] without annotated data. However, they fail to generate fine bitonal screentones as they only modify the statistics of higher-level perceptual features. Currently, there exists no deep-learning-based method that can directly convert color comics to screened manga, as generic deep image synthesis only captures prominent structures like outlines and object boundaries, but is less effective in handling very fine details and textures like screentones. For translating screened manga to color comics, a few attempts have been made to colorize manga or line drawings. Hensman et al. [2017] trained a deep model feeding only a single image to colorize manga and remove the screentones by averaging the colors of each closed region. However, it also eliminates tonal variation and shading within each region. Another possible solution is to first extract the structural lines [Li et al. 2017a] and then apply learning-based sketch colorization [Furusawa et al. 2017; Zhang et al. 2018]. However, it does not make use of the screentone information to colorize, as the screentones are removed. Instead, our method utilizes the original screentone information laid by the manga artist, to generate color comics with shading and vivid colors that align with the artist's intention.

All the above methods are unidirectional. In sharp comparison, we are interested in a unified framework for bidirectional comic filling style conversion with color and screentone consistency, i.e. the same screentone is consistently translated to the same color, and vice versa. Such consistency is necessary for translating a whole comic book. Recently, some learning-based methods [Yi et al. 2017; Zhu et al. 2017a] proposed bidirectional image translations that build bidirectional projections and predict image style translation between two domains based on unpaired data. Liu et al. [2017] proposed the UNIT framework, which encodes images from two different domains to a shared latent space. However, these methods are not effective in manga filling style translation due to the insufficient learning ability for screened manga input with a wide variation of screentones. Follow-up works like MUNIT [Huang et al. 2018] and DRIT [Lee et al. 2018], are proposed to disentangle content and style. However, these methods can only obtain a summarized texture style for the whole image, but are unable to perform a region-wise texture filling conversion as our method. We identify that the problem mainly stems from the difference of fundamental properties between color and screentone. By introducing ScreenVAE, we are enabled to effectively learn the filling style of both manga and color comics, which in turn enables a unified framework for bidirectional filling style translation.

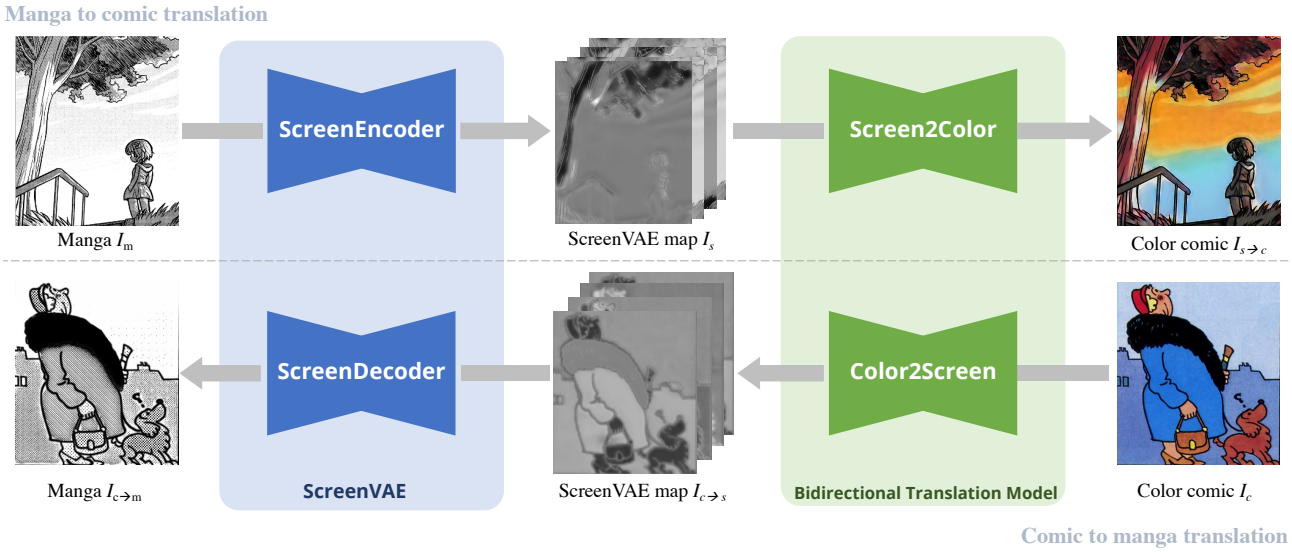


Fig. 4. Overview diagram of our whole system. When translating manga to western comics, we first convert the manga image I_m into a ScreenVAE map I_s, and then translate it to a western-style color comic $I_{s\to c}$. In reverse, when translating western comics to manga, we first convert the color comic image I_c to a ScreenVAE map $I_{c\to s}$, and then translate it to a manga image $I_{c\to m}$. The ScreenVAE and the bidirectional translation model are trained in sequence.

3 OVERVIEW

The framework of our comic style conversion system is illustrated in Fig. 4, which consists of two models: a variational model *ScreenVAE* for converting between manga and intermediate representation, and a bidirectional translation model for converting between color comics and intermediate representation. The screened manga is 1-channel, while the color comics is 3-channel. The intermediate representation is a 4-channel feature map (ScreenVAE map) which can both summarize the texture characteristics of the local neighborhood in a screened manga and capture the color characteristics of the local neighborhood in a color comic.

The ScreenVAE is a variational autoencoder that consists of a *ScreenEncoder* (*SE*) and a *ScreenDecoder* (*SD*). ScreenEncoder takes a screened manga as input (I_m) to generate an intermediate ScreenVAE map (I_s) in which each single point in this map *summarizes* the texture characteristics of the local neighborhood surrounding this point in I_m. ScreenDecoder decodes the ScreenVAE map (I_s) back to a screened manga (I'_m). The variational design of ScreenVAE ensures the intermediate representation to be interpolative. Hence, ScreenDecoder can interpolate over the whole screentone space in the dataset, and synthesize smooth transitions among screentones in the resultant screened manga. In order to allow our *ScreenVAE* to summarize and synthesize screentones with the awareness of content and region semantics, we adopt a multi-scale design for both ScreenEncoder and ScreenDecoder. Moreover, this also enforces constant-tone regions to contain stable values in the generated ScreenVAE map (Section 4.1).

Instead of naively performing bidirectional translation between screened manga domain and color comic domain, we translate between the intermediate ScreenVAE domain and color comic domain. It also consists of two networks, a Screen2Color generator that translates a ScreenVAE map (I_s) to a color comic ($I_{s\to c}$), and a Color2Screen generator that translates a color comic (I_c) to a ScreenVAE map ($I_{c\to s}$). To translate between the two domains with cycle-consistency, we propose a tailored bidirectional style translation model with adversarial loss based on [Zhu et al. 2017a]. Since screened manga usually contains a number of blank (solid-white) regions, pixels within these regions in the ScreenVAE map also exhibit similar features, which narrows the style diversity after converting to a color comic. Therefore, we allow the user to provide an optional reference image as the style target during conversion (detailed in Section 4.2).

The ScreenVAE and the bidirectional translation model are trained in sequence. All networks, including *SE*, *SD*, $G_{s\to c}$, and $G_{c\to s}$ are conditioned on the line drawing, as an extra input for better awareness of boundaries and regions, throughout the training and testing. We have also conducted a number of experiments to validate the effectiveness of our comic style translation. Implementation details and experiments are discussed in Section 5.

4 APPROACH

4.1 ScreenVAE

Due to the difference of fundamental properties of screening and color-filling, it is hard to directly convert between them even with the state-of-the-art learning-based methods. The main purpose of our ScreenVAE is to map a screened manga to an intermediate representation (ScreenVAE map) that owns similar properties with a color comic, so that this ScreenVAE map can be effectively translated to and from a color comic.

4.1.1 Network Architecture. The network structure of our ScreenVAE model is presented in Fig. 5. It is a variational autoencoder

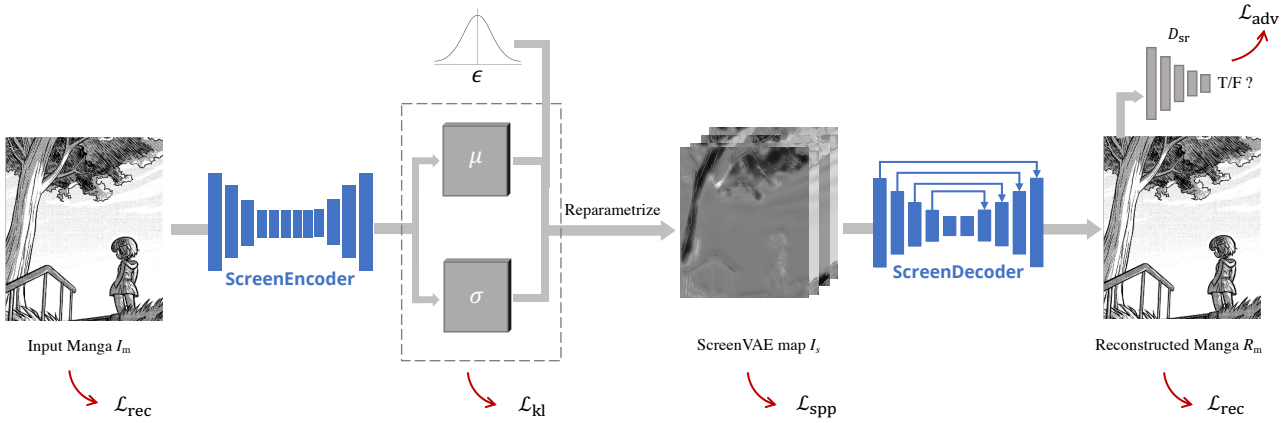


Fig. 5. Network architecture of our screentone variational autoencoder (ScreenVAE). Structure line of the manga is also fed to the both SE and SD to be aware of screentone boundaries and regions.

consisting of two jointly-trained networks, the parametric ScreenEncoder that encodes a screened manga into an intermediate ScreenVAE map and the ScreenDecoder that decodes a ScreenVAE map back to a screened manga. The ScreenVAE map has the same resolution as the input manga. Each pixel in the ScreenVAE map summarizes the texture characteristics of a local neighborhood in the input manga within the receptive field. From experiments, we find that 4 channels are already sufficient to capture most of the texture characteristics of the local neighborhood. We employ variational inference [Kingma and Welling 2013] to ensure the ScreenVAE map to be interpolative, just like the color space. In order to summarize the local texture characteristics with more attention on content and region semantics, we adopt a downscaling-upscaling network design for both ScreenEncoder and ScreenDecoder so as to enlarge the receptive fields. In particular, the ScreenEncoder utilizes a downscaling-upscaling network architecture with 6 residual blocks [He et al. 2016], while the ScreenDecoder utilizes a 5-level U-net structure [Ronneberger et al. 2015] with strided deconvolution operations to preserve more structures and generate bitonal screentones of different scales.

Our ScreenVAE model can be factorized as

$$\begin{aligned} I_s &= SE(\epsilon, I_m) \\ R_m &= SD(I_s) \end{aligned} \tag{1}$$

where SE and SD are the ScreenEncoder and ScreenDecoder, respectively; SE converts an input manga I_m to a parametric representation μ and σ; I_s is the ScreenVAE map sampled from μ and σ over the standard normal distribution $p(\epsilon) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ using the reparameterization trick [Kingma and Welling 2013; Rezende et al. 2014]; R_m is the screened manga reconstructed from I_s using SD.

4.1.2 Objectives. The optimization objective of our ScreenVAE consists of four losses, the reconstruction loss $\mathcal{L}_{rec}$, the superpixel difference loss $\mathcal{L}_{spp}$, the KL regularization loss $\mathcal{L}_z$, and the adversarial loss $\mathcal{L}_{adv}$.

Reconstruction Loss. The reconstruction loss $\mathcal{L}_{rec}$ ensures that the reconstructed manga R_m is similar to the input manga I_m and satisfies the the MAP criteria for variational inference. This loss enables the decoder SD to learn to generate the original manga. We adopt pixel-wise mean square error (MSE) to regularize the similarity. The reconstruction loss can be formally defined as

$$\mathcal{L}_{rec} = \mathbb{E}_{I_m \sim \mathcal{I}_m}\{\|R_m - I_m\|_2\} \tag{2}$$

where $\|\cdot\|_2$ denotes the L_2 norm, and $\mathbb{E}$ denotes the average operator over all images in the training manga dataset $\mathcal{I}_m$.

Superpixel Loss. The superpixel difference loss $\mathcal{L}_{spp}$ ensures that the ScreenVAE map I_s can well summarize the texture characteristics of constant-tone regions in the input manga I_m composed by either regular or irregular screentones. To extract constant-tone regions, we first acquire the tonal intensity map I_t of the input manga by total-variation based smoothing [Xu et al. 2011] and apply a simple linear iterative clustering (SLIC) [Achanta et al. 2012] to the tonal intensity map I_t to obtain a superpixel map I_{spp}. Then we remove the varying-toned regions from I_{spp} by estimating regional texture feature variances. In particular, if two regions are of the same tone but with different textures, we separate the regions into two superpixels in the superpixel map I_{spp}.

With the obtained superpixel map I_{spp}, we adopt the superpixel pooling network (SPN) [Kwak et al. 2017] to encourage a uniform region representation in the ScreenVAE map I_s within each superpixel.

$$\mathcal{L}_{spp} = \mathbb{E}_{I_m \sim \mathcal{I}_m}\{w_l \|I_s - \text{Superpixel}(I_s, I_{spp})\|^2\} \tag{3}$$

Here, w_l is the binary mask of structure lines (0 for structural lines, 1 for non-structural lines). Superpixel(I_s, I_{spp}) is a map in which each pixel is replaced by the average value of the corresponding superpixel region indexed by I_{spp} in the ScreenVAE map I_s. The value of each pixel in the superpixel map Superpixel(I_s, I_{spp}) is obtained by $(\sum_{k \in r^s} I_s^k)/k$ where I_s^k is the response of pixel k ($\in r^s$) in I_s, and K is the number of pixels in the superpixel region r^s indexed by I_{spp}.

KL Regularization Loss. The KL regularization loss ensures that the statistics of the ScreenVAE map is normally distributed. Given

(a) Screened manga (b) Tone intensity (c) Superpixel map

Fig. 6. Example of our superpixel map. Each region with same superpixel index should share the same intensity. ©Minamoto Tarou

an input manga I_m and the encoded ScreenVAE map $I_s = SE(z|I_m)$, we compute the KL regularization loss $\mathcal{L}_z$ as the summed Kullback-Leibler divergence [Kullback and Leibler 1951] of μ and σ over the standard normal distribution $p(z) = \mathcal{N}(\mathbf{0}, \mathbf{I})$:

$$\begin{aligned} \mathcal{L}_z &= \mathbb{E}_{I_m \sim \mathcal{I}_m}\{KL(\mathcal{N}(\mu, \sigma)|\mathcal{N}(\mathbf{0}, \mathbf{I}))\} \\ KL(\mathcal{N}(\mu, \sigma), \mathcal{N}(\mathbf{0}, \mathbf{I})) &= \tfrac{1}{2}\sum(\sigma^2 + \mu^2 - \log(\sigma^2) - 1) \end{aligned} \tag{4}$$

Here, $KL(\cdot, \cdot)$ denotes the KL divergence between two probability distributions.

Adversarial Loss. We find that the decoder tends to generate blurry results when only the above three losses are imposed. To generate manga with clear and visually pleasant screentones, we introduce the adversarial loss [Yu et al. 2019] for our variational model. A discriminator $D_{sr}(R_m)$ with 4 strided downscaling blocks is introduced. We adopt the WGAN-gp flavor of adversary [Gulrajani et al. 2017], which improves the training stability of GAN by applying the Wasserstein distance as value function and imposing gradient norm regularity, as

$$\begin{aligned} \mathcal{L}_{\text{adv}} = \quad & \mathbb{E}_{I_m \sim \mathcal{I}_m}\{D_{sr}(I_m) - D_{sr}(R_m)\} \\ & + \mathbb{E}_{\hat{I}_m \sim \hat{\mathcal{I}}_m}\{(\|\nabla_{\hat{I}_m} D_{sr}(\hat{I}_m)\|_2 - 1)^2\} \end{aligned} \tag{5}$$

where $\hat{I}_m$ is an image linearly interpolated from I_m and R_m with a random factor between 0 to 1. λ is the coefficient of gradient penalty and is empirically set to 10 in all our experiments.

With the above defined losses, the loss function of our ScreenVAE is defined as the weighted sum of the four losses as

$$\mathcal{L}_{\text{scr}} = \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{spp}}\mathcal{L}_{\text{spp}} + \lambda_z\mathcal{L}_z + \lambda_{\text{adv}}\mathcal{L}_{\text{adv}} \tag{6}$$

where λ_{rec}, λ_{spp}, λ_z, and λ_{adv} are the weights for balancing the loss terms. We empirically set $\lambda_{\text{rec}} = 5$, $\lambda_{\text{spp}} = 20$, $\lambda_z = 1$, and $\lambda_{\text{adv}} = 1$ in our experiments.

Our ScreenVAE successfully translates a manga image to a dense pixel-wise ScreenVAE map, in which each point summarizes the texture characteristics of a local neighborhood, without interfered by region boundary and overlapping fine structures. Moreover, the generated ScreenVAE map is interpolative, which allows the synthesis of high-quality screentones, some of which are even unseen from the dataset. Fig. 14 demonstrates such interpolative ability by generating smooth transitions across multiple screentones of different patterns and tonal intensities.

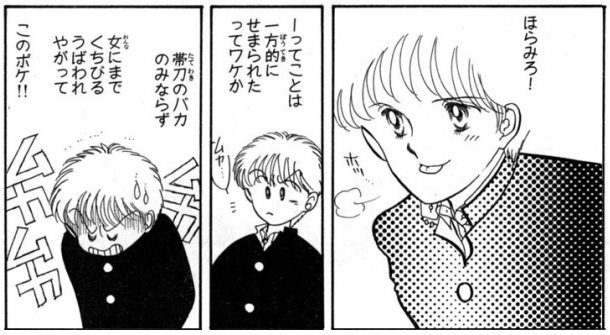


Fig. 7. Screened manga may have many blank regions which are lack of information. SonokiDeABC ©Tashiro Kimu, from the Manga109 dataset [Matsui et al. 2017]

4.2 Bidirectional Style Translation

While the ScreenVAE unifies the properties of screentone and color by converting between screened manga and ScreenVAE maps, our bidirectional style translation model learns to translate between ScreenVAE maps and color comics.

4.2.1 Network Architecture. To translate between two domains without paired data, we need unsupervised learning with sufficient constraints on the translating function. Similar with CycleGAN [Zhu et al. 2017a], we propose a bidirectional translation model consisting of two generators, a Screen2Color generator $G_{s\to c}$ that learns to translate a ScreenVAE map to color comic, and a Color2Screen generator $G_{c\to s}$ that learns to translate a color comic back to a ScreenVAE map.

We use a 7-level U-net structure [Ronneberger et al. 2015] for the Screen2Color generator $G_{s\to c}$ to capture the structures of a ScreenVAE map with a relatively large receptive field. Each level contains two convolution blocks, where each block consists of a convolution layer, a layer normalization layer, and a ReLU activation layer. The Color2Screen generator $G_{c\to s}$ also adopts a 7-level U-net structure, similar as $G_{s\to c}$. However, with this symmetric generator design, color comics generated by the Screen2Color generator usually contain poor color diversity and less vivid colors. We believe this may due to the fact that screened manga commonly contains blank (solid-white) region, so it results in inevitable loss of information in the corresponding pixels in the ScreenVAE map, as demonstrated in Fig. 7.

To provide essential information for blank regions, we follow the idea of style transfer by injecting a style vector to the Screen2Color generator $G_{s\to c}$. With a reference image I_r of the target style, we can fist extract its style vector v_r using a convolutional style extractor E_{st}, and then use the style vector as a hint, to encourage the output color comic $G_{m2c}(I_s, v_r)$ to have a similar color composition with the reference I_r. Note that, the Screen2Color generator $G_{s\to c}$ can still output colorful comics without reference image, by feeding a random vector as the style. Specifically, The style extractor E_{st} is a variational encoder consisting of 5 strided downscaling blocks and a fully connected layer, to map the reference input to a style vector. To manipulate image styles with the style vector v_r, we employ the

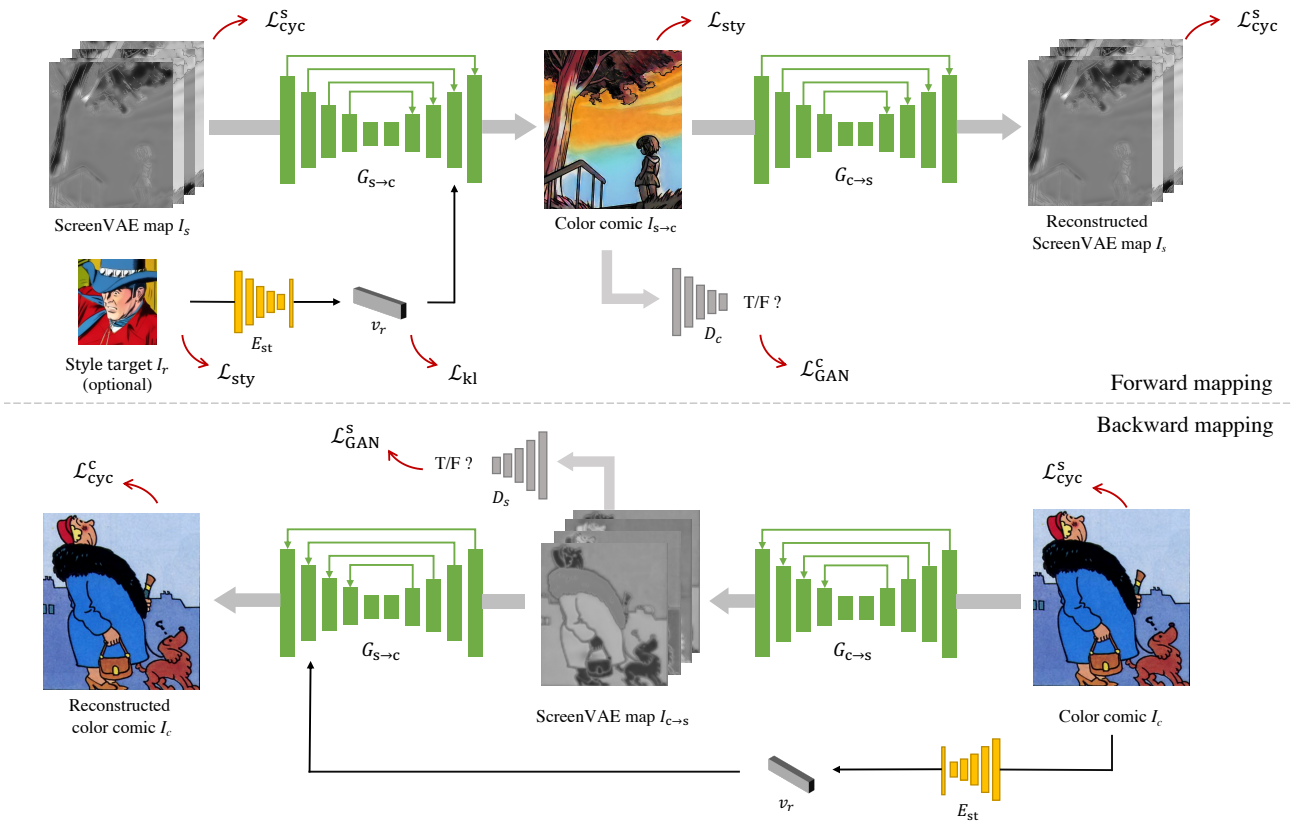


Fig. 8. Network architecture of our bidirectional translation model. Both $G_{s\to c}$, and $G_{c\to s}$ are conditioned by feeding the line drawing as an extra input, for better awareness of boundaries and regions.

adaptive instance normalization layer (AdaIN) [Huang and Belongie 2017] to the layer before each upscaling block to encourage the color diversity during the colorization of manga.

4.2.2 Objectives. We define four losses in our bidirectional translation model, including the bidirectional cycle-consistency loss ($\mathcal{L}^c_{cyc}$ and $\mathcal{L}^s_{cyc}$), the adversarial loss ($\mathcal{L}^c_{GAN}$ and $\mathcal{L}^s_{GAN}$), the style loss $\mathcal{L}_{sty}$, and the style regularization loss $\mathcal{L}_v$.

Bidirectional Cycle-consistency Loss. The bidirectional cycle-consistency loss, borrowed from CycleGAN, is to constrain the image to get back after forward and backward mappings. The idea is that, if we translate a color comic to a screened manga and then translate the generated manga back, the reconstructed color comic should be the same as the input color comic. The consistency for the backward translation from a screened manga to a color comic is similar. The loss is therefore defined as

$$\begin{aligned}\mathcal{L}^c_{cyc} &= \mathbb{E}_{I_c\sim\mathcal{I}_c}\|G_{s\to c}(G_{c\to s}(I_c)) - I_c\|_1\\ \mathcal{L}^s_{cyc} &= \mathbb{E}_{I_s\sim\mathcal{I}_s}\|G_{c\to s}(G_{s\to c}(I_s)) - I_s\|_1\end{aligned} \tag{7}$$

where $\|\cdot\|_1$ is the L_1 norm operator.

Adversarial Loss. The adversarial loss is used to encourage the network to generate high-quality color comics and screened manga. To do so, we introduce two discriminators D_s and D_c, each with 5 downscaling blocks, and train them with WGAN-gp[Gulrajani et al. 2017]:

$$\begin{aligned}\mathcal{L}^c_{GAN} = \;& \mathbb{E}_{I_c\sim\mathcal{I}_c}\{D_c(I_c)\} - \mathbb{E}_{I_s\sim\mathcal{I}_s}\{D_c(G_{s\to c}(I_s))\}\\ &+\mathbb{E}_{\hat{I}_c\sim\hat{\mathcal{I}}_c}\{(|\nabla_{\hat{I}_c}D_c(\hat{I}_c)|_2 - 1)^2\}\\ \mathcal{L}^s_{GAN} = \;& \mathbb{E}_{I_s\sim\mathcal{I}_s}\{D_s(I_s)\} - \mathbb{E}_{I_c\sim\mathcal{I}_c}\{D_s(G_{c\to s}(I_c))\}\\ &+\mathbb{E}_{\hat{I}_s\sim\hat{\mathcal{I}}_s}\{(|\nabla_{\hat{I}_s}D_s(\hat{I}_s)|_2 - 1)^2\}\end{aligned} \tag{8}$$

Here, $\hat{I}_c$ is an image linearly interpolated from I_c and $G_{s\to c}(I_s)$ and $\hat{I}_s$ is an image linearly interpolated from I_s and $G_{c\to s}(I_c)$.

Style Loss. To encourage the generated color comic to have a similar style with the reference image, we propose the style loss that measures the statistics of the style features between the generated color comic and the reference image. The style features are extracted using the illustration2vec network ϕ [Saito and Matsui 2015]. We further define the style loss based on the style features as that of [Li et al. 2017b]:

$$\begin{aligned}\mathcal{L}_{sty} = \;& \mathbb{E}_{(I_s,I_c)\sim(\mathcal{I}_f,\mathcal{I}_j)}(\textstyle\sum_l \|\text{mean}(\phi^l(I_c))\\ &-\text{mean}(\phi^l(G_{s\to c}(I_s, E_{st}(z|I_c))))\|_2\\ &+\textstyle\sum_l \|\text{std}(\phi^l(I_c) - \text{std}(\phi^l(G_{s\to c}(I_s, E_{st}(z|I_c))))\|_2)\end{aligned} \tag{9}$$

Here, mean$(\cdot)$ is the mean operator; std$(\cdot)$ is the standard deviation operator; ϕ^l denotes the layer l in ϕ. Since we focus more on the

color compositions of the reference image, we choose the relu1_1, relu2_1, relu3_2, relu4_2 layers of the illustration2vec network for measuring style loss in our experiments.

Style Regularization Loss. To make the style vector space to be normally distributed [Kingma and Welling 2013], we compute the style regularization loss by estimating the summed Kullback-Leibler divergence [Kullback and Leibler 1951] of the extracted style vector $v_r = E_{st}(z, I_c)$ over a standard normal distribution $p(z) = \mathcal{N}(\mathbf{0}, \mathbf{I})$.

$$\mathcal{L}_{\text{kl}} = \mathbb{E}_{I_c \sim I_c} KL(E_{st}(z, I_c) | \mathcal{N}(\mathbf{0}, \mathbf{I})) \tag{10}$$

The overall loss function is hence defined as

$$\begin{aligned} \mathcal{L}_{\text{bi}} = \quad & \alpha_{\text{cyc}}(\mathcal{L}^c_{\text{cyc}} + \mathcal{L}^s_{\text{cyc}}) + \alpha_{\text{GAN}}(\mathcal{L}^c_{\text{GAN}} + \mathcal{L}^s_{\text{GAN}}) \\ & + \alpha_{\text{sty}}\mathcal{L}_{\text{sty}} + \alpha_{\text{kl}}\mathcal{L}_{\text{kl}} \end{aligned} \tag{11}$$

where α_{cyc}, α_{GAN}, α_{sty} and α_{kl} are the weights for each loss. We empirically set $\alpha_{\text{cyc}} = 20$, $\alpha_{\text{GAN}} = 1$, $\alpha_{\text{sty}} = 20$, and $\alpha_{\text{kl}} = 0.1$ in our experiments. With these losses, our bidirectional translation model can successfully learn to map between ScreenVAE maps and color comics. Given a ScreenVAE map for a screened manga, our bidirectional translation model can convert it to a color comic with color diversity. Meanwhile, the generated ScreenVAE map from a color comic can be used to generate a screened manga via the ScreenVAE which favors the use of a rich set of screentones.

5 EXPERIMENTS AND DISCUSSIONS

5.1 Implementation Details

5.1.1 Data Preparation. Manga109[Matsui et al. 2017] is the only public manga dataset. However, the resolution of the images is not high enough and most of the screentones are blurred. As there is no public dataset tailored for high-resolution screened manga and color comics, we manually collect 6201 screened manga of resolution 1500×1046 and 1934 color comics of resolution 1528×994 to train our models. For each screened manga and color comic, we extract the structural lines using the line extraction model by Li et al. [2017a].

5.1.2 Training. We trained the model in the PyTorch framework [Paszke et al. 2017]. The network weights are randomly initialized using the method of [He et al. 2015]. During the training of the ScreenVAE model, we empirically set parameters as $\lambda_z = 1$, $\lambda_{\text{rec}} = 5$, $\lambda_{\text{adv}} = 1$ and $\lambda_{\text{spp}} = 20$. Adam solver [Kingma and Ba 2014] is used with a batch size of 4 and an initial learning rate of 0.0002. To train the bidirectional translation model, we set $\alpha_{\text{cyc}} = 20$, $\alpha_{\text{GAN}} = 1$, $\alpha_{\text{kl}} = 0.1$, and $\alpha_{\text{sty}} = 20$. Adam solver [Kingma and Ba 2014] is used with a batch size of 2 and an initial learning rate of 0.0002.

5.2 Comparisons on Color-filling to Screening

To evaluate our method, we first evaluate the quality of our screened manga converted from color comics. We visually compare our results to the state-of-the-art traditional manga screening method [Qu et al. 2008] and CycleGAN [Zhu et al. 2017a] trained with our training data. Fig. 9 shows the results. When converting color comics to screened manga, we need to guarantee both the overall grayness conformity and the inter-region visual distinguishability. From the results, we can see that our method not only preserves the tonal intensity, but also generates a rich variety of screentones including dots, regular line stripes, and stochastic patterns. In the last row of Fig. 9, our method even generates line stripes with smoothly changing tonal intensities on the face of the man, to ensemble the gradual intensity change.

The third column of Fig. 9 shows the results of Qu et al. [2008], which determines the mapping from color segments to screentones by optimizing mainly for color distinguishability. So it may fail to preserve tonal intensity, as shown in the tree example (second row). Moreover, their method fails to generate consistent screentones with smoothly changing tonal intensities for smoothly changing colors, as shown in the face region and the clothes regions in the man example (last row). In contrast, our method preserves the tonal intensity in the tree example, and generates visually pleasant tone-changing screentone patterns for the face and clothes regions in the man example.

The third column of Fig. 9 shows the results of CycleGAN [Zhu et al. 2017a]. CycleGAN tends to generate simple and low-contrast screentones with little variation of patterns. It fails to preserve the tonal intensity as shown by the dress of the old lady example (first row). It never generates regular line strips (and some other screentones) as our method does, even if the training data contains such screentones. In sharp comparison, our method preserves the tone and generates manga images with a rich variety of screentones (dots, lines strip, stochastic screentones, etc).

5.3 Comparisons on Screening to Color-filling.

We further evaluate our method on translating screened manga to color comics. Two existing methods are selected for comparisons, including sketch colorization [Zhang et al. 2018] (pretrained) and CycleGAN [Zhu et al. 2017a] (trained with our training dataset). We directly use the pretrained sketch colorization model since it requires paired training dataset, which is not available in our case. To generate results using sketch colorization, we first extract the structural line map from the tested screened manga using [Li et al. 2017a] and feed the line drawing to their model for colorization, without providing user hints. We did not compare with the traditional manga colorization method [Qu et al. 2006] since it is not automatic and requires users to provide manual color hints. Fig. 11 compares the colorized results of all methods. Fig. 10 shows more of our results with various reference hints.

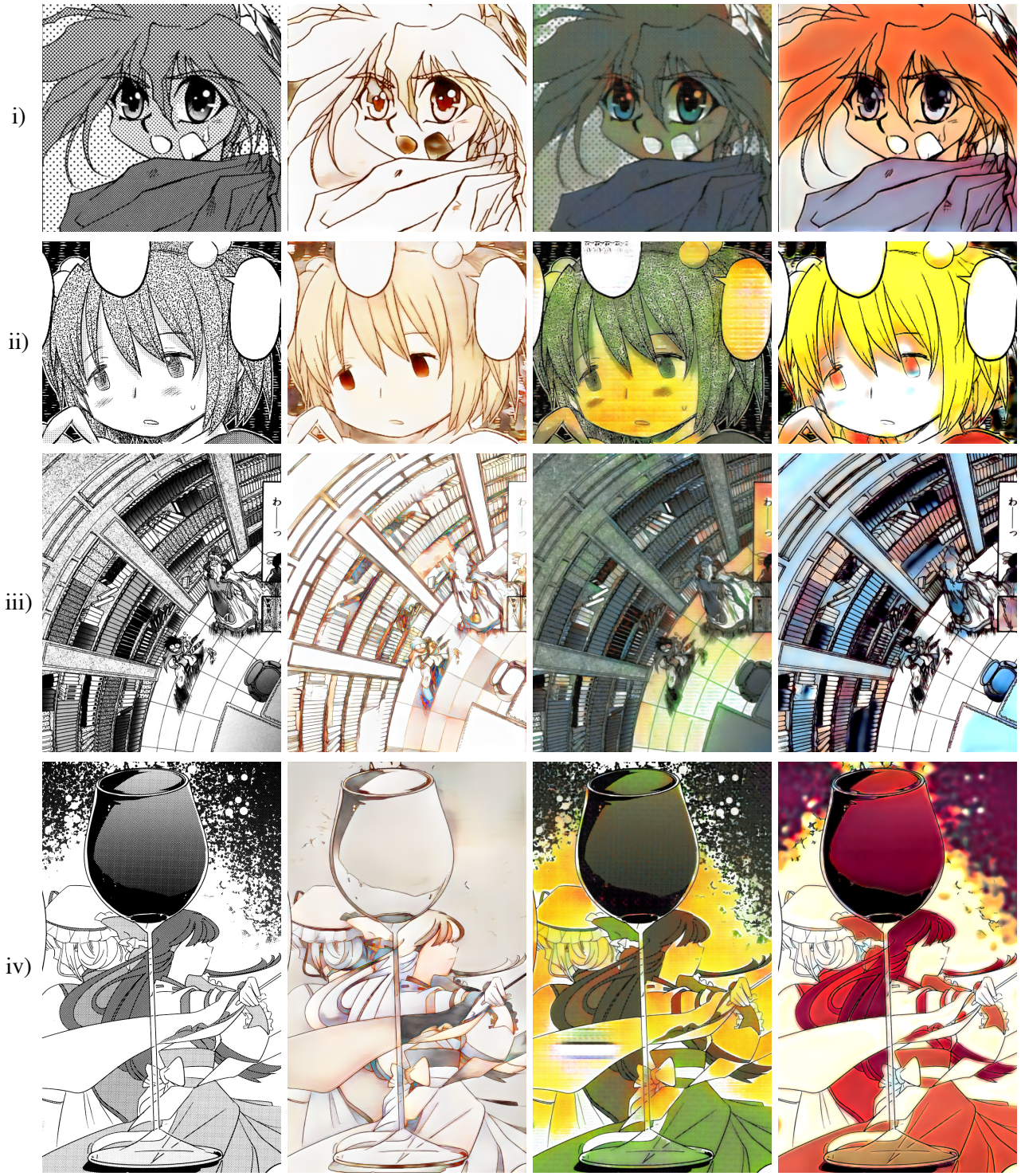
From Fig. 11, we can see that the sketch colorization method fails to generate plausible results. One reason is that we provide no color hint to their method during the testing. The other reason is that it is not trained with our training data, since no paired data is available. Moreover, it can only colorize sketches, but cannot perform color-filling to screening as our method does. In contrast, our model utilizes the screening information laid by manga artists, and tries to retain the original intention during the colorization. Our results exhibit color of better diversity and are aligned to the original screening in all examples.

Results generated by CycleGAN tend to retain the original screentones during the colorization (first and second rows), and sometimes exhibit severe color bleeding (third row). This may be due to failure of CycleGAN to recognize the rich variety of screentones, which results in the mix-up with the semantic structures. This evidences the importance of our ScreenVAE design. With our ScreenVAE, our

(a) Western comic (b) [Qu et al. 2008] (c) CycleGAN [Zhu et al. 2017b] (d) Our method

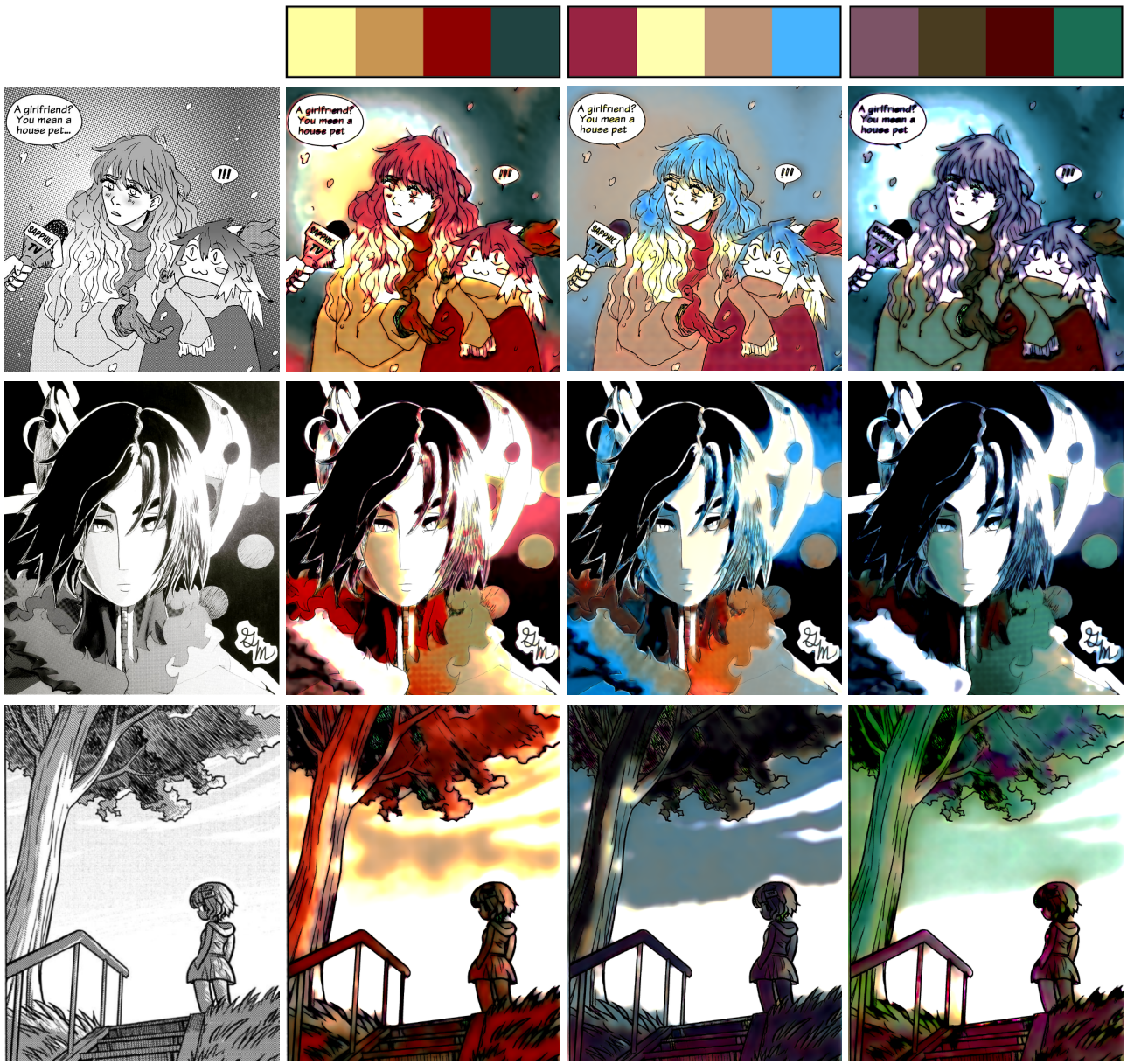
Fig. 9. Comparison of screened manga converted by two competitors and our method.

method can better distinguish screentones and semantic structures, and will not be confused by the rich variety of screentones. Hence,



(a) Screened manga (b) [Zhang et al. 2018] (c) CycleGAN [Zhu et al. 2017b] (d) Our method

Fig. 10. Comparison of colored comic converted by two competitors and our method.



(a) Screened manga (b) Our results with reference 1 (c) Our results with reference 2 (d) Our results with reference 3

Fig. 11. Colored comic generated by our method with different reference input. The color palettes of the references are shown on the top.

we can replace the screening with diversified color-filling. Smooth change of screentones is replaced by smooth change of colors, as shown in the wine glass example on the fourth row.

Optionally, our style translation model accepts reference image to generate multi-modal results with enhanced color diversity. This is done by injecting the style code extracted from the reference image. The color comics generated with three different references are shown in Fig. 10. The major color palettes of the reference images are shown on the top of the corresponding columns. The generated color comics mostly confirm to the major colors of the given reference images.

5.4 Analysis on ScreenVAE

5.4.1 Ablation study. To verify the effectiveness of individual loss term, we conduct an ablation study (except for the reconstruction loss $\mathcal{L}_{\text{rec}}$) by visually comparing the generated intermediate representation and reconstructed results. The results are shown in Fig. 13. Without the adversarial loss, blurry results may be generated (top

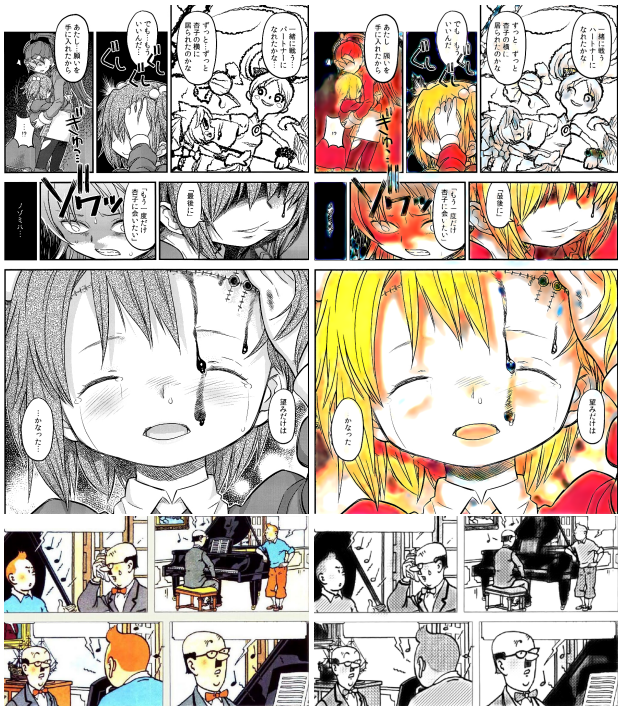


Fig. 12. Our colorization and screening are consistent for the same objects through frames. KaerimichiNoMajo ©A-10

row of Fig. 13(c)). Without the KL regularization loss, the model degrades into an AutoEncoder as the network tries to preserve itself and the σ decreases to 0. As we can see in bottom row of Fig. 13(b), the reconstructed background may exhibit seam within the screentone and cannot generate gradually changed effect. Without the superpixel loss, the network may fail to recognize multi-scale screentones and irregular screentones, so the generated representations for these regions are usually not smooth (bottom row of Fig. 13(d)). In comparison, the combined loss can help the network to recognize different types of clear screentones and generate a consistent representation for the screentones (Fig. 13(e)).

5.4.2 ScreenVAE v.s. Classical Texture Analysis. ScreenVAE can be regarded as a more intelligent texture analysis tool. In Fig. 3, we visually compare it to the classical Gabor wavelet texture analysis [Manjunath and Ma 1996]. We visualize both the Gabor feature and our ScreenVAE map by regarding the three major components as color values. As observed, both Gabor feature and our ScreenVAE map have the capability of summarizing the texture characteristics of the local neighborhood in the input manga. Both features can distinguish different types of screentones. However, the Gabor feature exhibits severe artifacts near region boundaries with blurry double edges due to its window-based analysis. In sharp contrast, our ScreenVAE map exhibits tight boundaries with no double edges, and is able to identify textures in narrow structures and overlapping structure. Our ScreenVAE recognizes irregular patterns and projects it to smooth values in the ScreenVAE map. It is because that the network design of a series of convolutions can better solve the windowing problem than the Gabor wavelet features which are single-window features. Meanwhile, our superpixel loss further encourages the network to push towards the boundary.

5.5 Interpolativity of ScreenVAE Feature

Artists of both screened manga and color comics commonly use smoothly changing screentones or colors to express shading or atmosphere, e.g. the smooth color shading of the man's face (Fig. 1 left) and the smoothly changing screentones of the girl's hair and background (Fig. 2). This is why interpolativity is needed for our ScreenVAE feature. With our end-to-end variational autoencoder design, our latent ScreenVAE space is interpolative. Our model can generate screentones that may or may not be seen in the training data, and also screentones that are close neighbors in our latent ScreenVAE space. This allows us to generate not only high-quality screentones, but also smoothly changing screentones to maintain various characteristics, such as gradients and highlights. Fig. 14 shows the synthesized interpolation over multiple screentone types as well as multiple screentone intensities. Note that the generated screentones are gradual, of high-quality, and sometimes unseen-but-reasonable.

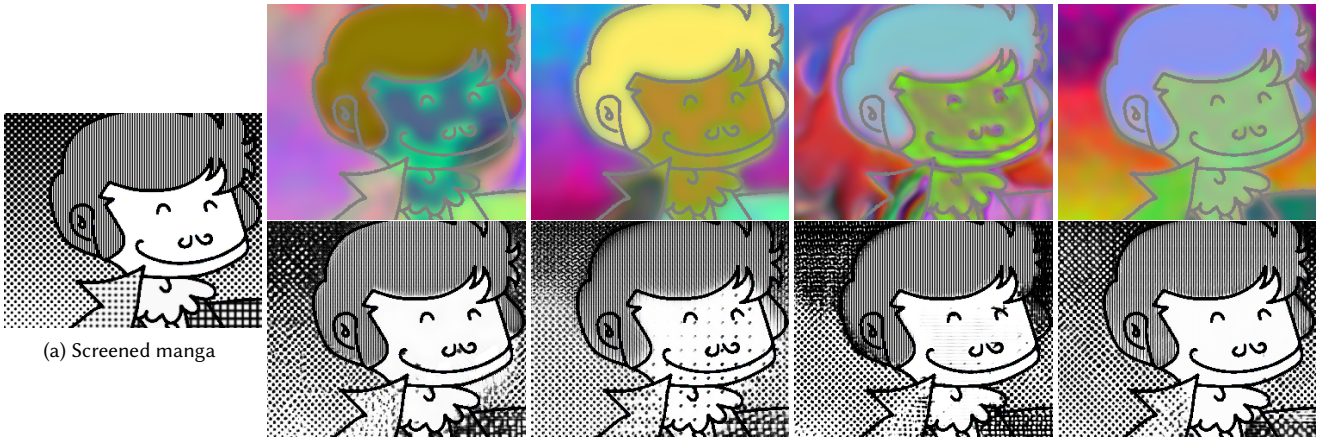
5.6 User study

We conduct two user studies to evaluate our generated results in terms of visual quality. For converting color comics to manga, we compare our generated manga images with the results achieved by [Qu et al. 2008] and CycleGAN [Zhu et al. 2017a]. For converting manga to color comics, we compare our results with the results generated by [Zhang et al. 2018] and CycleGAN [Zhu et al. 2017a]. For each user study, 30 samples are randomly picked from our dataset. The detailed questionnaire examples are provided in the supplementary materials.

The first user study evaluates on the screened manga. Participants are asked to rate each example of screened manga in terms of visual similarity, screentone diversity and screentone consistency respectively. The score ranges from 0 to 5, where the higher score indicates better quality. For each method, 300 scores are collected and the statistics are shown in Fig. 15(a). As we can see, the results generated by our method are consistently preferred by the participants over the other two methods. The second user study evaluates on the color comics generated by all methods. Participants are asked to select their preferred color image and rate each color comic in terms of visual richness, color consistency and color harmony respectively. Still, the score ranges from 0 to 5. The results are shown in Fig. 15, which tells that our method generally receives higher ratings due to the capability of preserving the image content and the contrast. Besides, the T-test analysis is available in the supplementary material.

5.7 Timing

Table 1 shows the average running time of our method on different resolution of input images. All timing statistics are recorded on a PC with Nvidia GeForce GTX Titan X GPU installed. Our method only takes about 0.5 second to process a 1024×1024 image, facilitating real-time applications.



(a) Screened manga (b) w/o $\mathcal{L}_z$ (c) w/o $\mathcal{L}_{adv}$ (d) w/o $\mathcal{L}_{spp}$ (e) $\mathcal{L}_{scr}$

Fig. 13. Effects of individual losses on ScreenVAE. The top row shows the encoded features and the bottom row are the reconstructed images of each network.

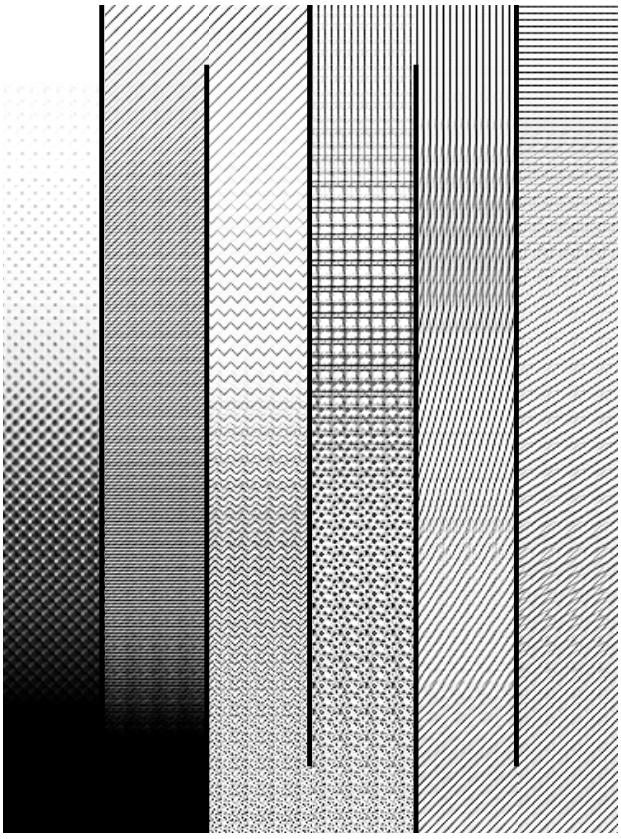
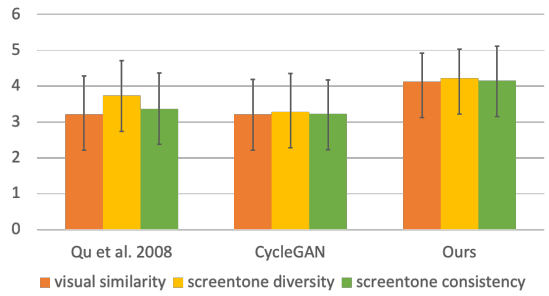
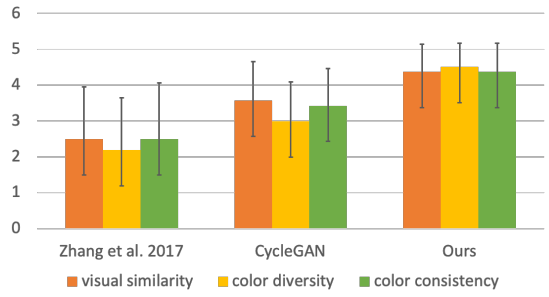


Fig. 14. Smooth interpolation over multiple screentones with various pattern types and tonal intensities. It is generated by interpolating in the ScreenVAE feature domain.



(a) Translation from color comic to screened manga



(b) Translation from screened manga to color comic

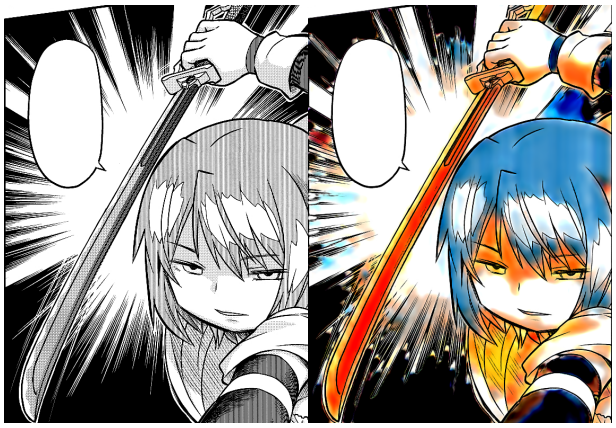
Fig. 15. The results of user studies.

5.8 Limitations

Our framework still suffers from some limitations. The current trained model cannot generate screentones of large-scale patterns. This is mainly due to the lack of such data in our training set. Our model may accidentally regard some non-screened regions

resolution	western comic to manga	manga to color comic
256 × 256	0.040 sec	0.045 sec
512 × 512	0.137 sec	0.160 sec
1024 × 1024	0.527 sec	0.598 sec

Table 1. Running time with different resolution of input images.



(a) Screened manga (b) Our result

Fig. 16. Speed lines may be accidentally interpreted as screentones and then colorized. KaerimichiNoMajo ©A-10

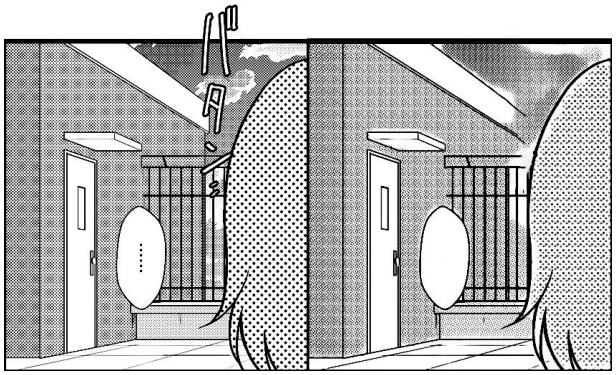
as screened regions and try to colorize them. For example, in Fig. 16, the speed lines are occasionally colorized by blue and orange colors.

6 APPLICATIONS

6.1 Manga Inpainting

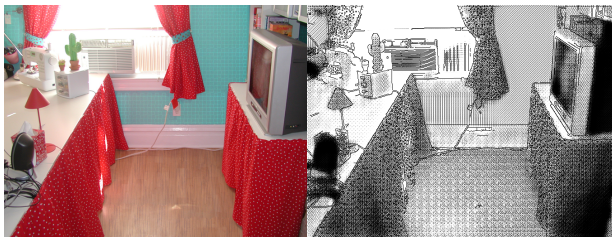
The ScreenVAE map is not just useful for bidirectional translation of comic filling styles. It can also be used to ease the manga inpainting application. Manga usually contains speech balloons, huge onomatopoeia text, etc. Electronic publishing of manga will commonly remove such balloons and onomatopoeia text, and add them back to produce simple animation effect later for more entertaining reading experience. However, disoccluded regions in Fig. 17(a) have to be manually inpainted currently, due to difficulty in inpainting manga with screened region. Note that multiple screentones and structures may appear in the disoccluded regions. The bitonal nature of screentones further complicates the inpainting. All these lead to technical challenges.

Since our ScreenVAE feature summarizes local texture characteristics at each point and is interpolative, it shares the same nice property as color. This means all inpainting techniques designed for color images, may also be applicable to our ScreenVAE map. So instead of directly inpaint the bitonal manga, we can instead inpaint the ScreenVAE map and synthesize manga based on the ScreenVAE map. To demonstrate such feasibility, we apply the image inpainting method proposed by Liu et al. [2018] to inpaint our ScreenVAE map. The inpainted ScreenVAE map is then converted back to screened manga with our ScreenVAE model. Fig. 17 presents our inpainted



(a) Input manga image (b) Our result

Fig. 17. Manga inpainting under a rectangular mask.



(a) Input photograph (b) Our result

Fig. 18. Screened manga generated from a natural photograph.

results. High-quality screentones and major structures are naturally inpainted (Fig. 17).

6.2 Natural Photographs to Manga

The property of ScreenVAE feature is not only close to that of color comics, but also close to that of natural photographs. In other words, we can naturally extend our method to convert natural photographs to screened manga. Fig. 18 shows the screened manga generated from the input natural photograph using our method. Our result exhibits tone-varying screentones for color-varying region, e.g. the floor.

7 CONCLUSION

In this paper, we proposed a learning-based framework for converting comic filling styles in an automatic and consistent fashion. It frees artists from the labor-intensive and time-consuming process of converting from screened manga to color comics, or from the color comics to screened manga. A core contribution is the awareness of the difference of fundamental properties of screening and color-filling. By proposing our ScreenVAE model, we are able to map the screening style to an intermediate domain, which has similar representability as the color-filling style. This effectively unifies the properties of screening and color-filling, which in turn eases the bidirectional translation between screened manga and color

comics. Our results show superior screening quality given the color comics, and the generated color comics align with the artists' intention via recognizing the original screentone information from the input screened manga. We also demonstrated the potential applications of our ScreenVAE for manga inpainting and photo-to-manga conversion.

8 ACKNOWLEDGMENT

This project is supported by the Research Grants Council of the Hong Kong Special Administrative Region, under RGC General Research Fund (Project No. CUHK14217516).

REFERENCES

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence* 34, 11 (2012), 2274–2282.

Vincent Andrearczyk and Paul F Whelan. 2016. Using filter banks in convolutional neural networks for texture classification. *Pattern Recognition Letters* 84 (2016), 63–69.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3606–3613.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, and Andrea Vedaldi. 2016. Deep filter banks for texture recognition, description, and segmentation. *International Journal of Computer Vision* 118, 1 (2016), 65–94.

Mircea Cimpoi, Subhransu Maji, and Andrea Vedaldi. 2015. Deep filter banks for texture recognition and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3828–3836.

Chie Furusawa, Kazuyuki Hiroshiba, Keisuke Ogaki, and Yuri Odagiri. 2017. Comicolorization: semi-automatic manga colorization. In *SIGGRAPH Asia 2017 Technical Briefs*. ACM, 12.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2016. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2414–2423.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. In *Advances in neural information processing systems*. 5767–5777.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*. 1026–1034.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

Paulina Hensman and Kiyoharu Aizawa. 2017. cGAN-based manga colorization using a single training image. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, Vol. 3. IEEE, 72–77.

Xun Huang and Serge Belongie. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*. 1501–1510.

Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. 2018. Multimodal Unsupervised Image-to-image Translation. In *ECCV*.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-Image Translation with Conditional Adversarial Networks. *CVPR* (2017).

Anil K Jain and Farshid Farrokhnia. 1990. Unsupervised texture segmentation using Gabor filters. In *1990 IEEE international conference on systems, man, and cybernetics conference proceedings*. IEEE, 14–19.

John F Jarvis, C Ni Judice, and WH Ninke. 1976. A survey of techniques for the display of continuous tone pictures on bilevel displays. *Computer graphics and image processing* 5, 1 (1976), 13–40.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*. Springer, 694–711.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.

Suha Kwak, Seunghoon Hong, and Bohyung Han. 2017. Weakly supervised semantic segmentation using superpixel pooling network. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. 2018. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European conference on computer vision (ECCV)*. 35–51.

Anat Levin, Dani Lischinski, and Yair Weiss. 2004. Colorization using optimization. In *ACM transactions on graphics (tog)*, Vol. 23. ACM, 689–694.

Chengze Li, Xueting Liu, and Tien-Tsin Wong. 2017a. Deep Extraction of Manga Structural Lines. *ACM Transactions on Graphics (SIGGRAPH 2017 issue)* 36, 4 (July 2017), 117:1–117:12.

Yanghao Li, Naiyan Wang, Jiaying Liu, and Xiaodi Hou. 2017b. Demystifying neural style transfer. *arXiv preprint arXiv:1701.01036* (2017).

Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. 2018. Image Inpainting for Irregular Holes Using Partial Convolutions. In *The European Conference on Computer Vision (ECCV)*.

Ming-Yu Liu, Thomas Breuel, and Jan Kautz. 2017. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems*. 700–708.

Xiuwen Liu and DeLiang Wang. 2006. Image and texture segmentation using local spectral histograms. *IEEE Transactions on Image Processing* 15, 10 (2006), 3066–3077.

Bangalore S Manjunath and Wei-Ying Ma. 1996. Texture features for browsing and retrieval of image data. *IEEE Transactions on pattern analysis and machine intelligence* 18, 8 (1996), 837–842.

Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2017. Sketch-based Manga Retrieval using Manga109 Dataset. *Multimedia Tools and Applications* 76, 20 (2017), 21811–21838.

Wai-Man Pang, Yingge Qu, Tien-Tsin Wong, Daniel Cohen-Or, and Pheng-Ann Heng. 2008. Structure-aware halftoning. In *ACM Transactions on Graphics (TOG)*, Vol. 27. ACM, 89.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. (2017).

Yingge Qu, Wai-Man Pang, Tien-Tsin Wong, and Pheng-Ann Heng. 2008. Richness-Preserving Manga Screening. *ACM Transactions on Graphics (SIGGRAPH Asia 2008 issue)* 27, 5 (December 2008), 155:1–155:8.

Yingge Qu, Tien-Tsin Wong, and Pheng-Ann Heng. 2006. Manga colorization. In *ACM Transactions on Graphics (TOG)*, Vol. 25. ACM, 1214–1220.

Trygve Randen and John Hakon Husoy. 1999. Filtering for texture classification: A comparative study. *IEEE Transactions on pattern analysis and machine intelligence* 21, 4 (1999), 291–310.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082* (2014).

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.

Masaki Saito and Yusuke Matsui. 2015. Illustration2vec: a semantic vector representation of illustrations. In *SIGGRAPH Asia 2015 Technical Briefs*. ACM, 5.

Daniel Sỳkora, Jan Buriánek, and Jiří Žára. 2004. Unsupervised colorization of black-and-white cartoons. In *Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*. ACM, 121–127.

Robert Ulichney. 1987. *Digital halftoning*. MIT press.

Georges Winkenbach and David H Salesin. 1994. Computer-generated pen-and-ink illustration. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. ACM, 91–100.

Li Xu, Cewu Lu, Yi Xu, and Jiaya Jia. 2011. Image smoothing via L 0 gradient minimization. In *ACM Transactions on Graphics (TOG)*, Vol. 30. ACM, 174.

Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. 2017. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*. 2849–2857.

Xianwen Yu, Xiaoning Zhang, Yang Cao, and Min Xia. 2019. VAEGAN: a collaborative filtering framework based on adversarial variational autoencoders. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. 4206–4212.

Lvmin Zhang, Chengze Li, Tien-Tsin Wong, Yi Ji, and Chunping Liu. 2018. Two-stage sketch colorization. In *SIGGRAPH Asia 2018 Technical Papers*. ACM, 261.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017a. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.

Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. 2017b. Toward Multimodal Image-to-Image Translation. In *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). Curran Associates, Inc., 465–476. http://papers.nips.cc/paper/6650-toward-multimodal-image-to-image-translation.pdf