DOI: 10.1111/cgf.14791

COMPUTER GRAPHICS *forum*
Volume 42 (2023), number 6, e14791

Two-Step Training: Adjustable Sketch Colourization via Reference Image and Text Tag

Dingkun Yan, Ryogo Ito, Ryo Moriai and Suguru Saito

Department of Computer Science, Tokyo Institute of Technology, Meguro-ku, Japan
{yan, ryogo, mori, suguru}@img.cs.titech.ac.jp

Abstract

Automatic sketch colourization is a highly interestinged topic in the image-generation field. However, due to the absence of texture in sketch images and the lack of training data, existing reference-based methods are ineffective in generating visually pleasant results and cannot edit the colours using text tags. Thus, this paper presents a conditional generative adversarial network (cGAN)-based architecture with a pre-trained convolutional neural network (CNN), reference-based channel-wise attention (RBCA) and self-adaptive multi-layer perceptron (MLP) to tackle this problem. We propose two-step training and spatial latent manipulation to achieve high-quality and colour-adjustable results using reference images and text tags. The superiority of our approach in reference-based colourization is demonstrated through qualitative/quantitative comparisons and user studies with existing network-based methods. We also validate the controllability of the proposed model and discuss the details of our latent manipulation on the basis of experimental results of multi-label manipulation.

Keywords: colour, image and video processing, image/video editing

CCS Concepts: • Applied computing → Fine arts; • Computing methodologies → Artificial intelligence; Image processing

1. Introduction

Anime illustration is a worldwide popular art form of image owing to its diverse colour composition and fascinating character design. However, colourizing a sketch image is a time-consuming and tedious process, even for professional artists. It is also challenging for neural networks owing to the absence of colour and texture in sketch images. Given that line art has enormous market demand, both research and industry can benefit from successfully developing a fully-/semi-automatic colourization system.

Existing sketch colourization approaches usually require additional hints to synthesize colours [ZLW*18, ZJLL17, LKL*20, HYES19]. In accordance with how hints are given, these methods can be categorized into three types: text-based [ZMG*19, HYES19], user-guided [SDC09, PMC22, SMYA14, FTR18, ZLW*18] and reference-based [ZJLL17, LKL*20]. Text-based methods use the binary attributes of tags, words and sentences to colourize images, but they are insufficient to adjust the degree of colours. User-guided methods require users to specify colours for regions with spots or sprays, so a basic knowledge of line art and an interactive system is necessary. In addition, user-guided approaches are inefficient in colourizing different sketch images as hints are designated in correspondence to each input image. Although reference-based methods overcome these limitations, developing such a method is more challenging as it requires an additional evaluation of the colour similarity with references and a semantically well-paired training dataset, which is currently unavailable and expensive to build.

To improve the quality and controllability of reference-based results, in this paper, we present a generative adversarial network (GAN) that adopts a pre-trained convolutional neural network (CNN) and two-step training. We design a reference-based channel-wise attention (RBCA) block and a self-adaptive multi-layer perceptron (MLP) to enable the proposed model to generate high-quality images through references and text tags, as shown in Figure 1. We develop spatial latent manipulation for our attention-based receiving block in the second training. Qualitative/quantitative comparisons and user studies with other methods are taken to show our advantages in reference-based colourization. Experiments on multi-label manipulation also demonstrate the controllability of our results. The main contributions of this paper are concluded as follows:

Figure 1: *Multi-label manipulated results generated by the proposed* Attention *and* M-Attention *models. Our method can colourize sketch images using reference images and text tags.*

- A two-step training system that can achieve state-of-the-art reference-based results and control the colours through reference images and text tags.
- Investigate spatial latent manipulation on the basis of a pre-trained CNN and propose a self-adaptive MLP for disentanglement.

The rest of this paper starts by reviewing related works in Section 2. Section 3 explains the loss functions used in the two-step training and important components of the proposed models. Section 4 includes the implementation details, experimental results and corresponding discussion. Section 5 concludes the paper.

2. Related work

Image generation with GANs:. GANs are one of the most prevalent generative models owing to their effectiveness in synthesizing high-quality images. Goodfellow *et al.* [GPM*14] first proposed the vanilla GAN to decode random noise into images [GPM*14], whose learning process is extremely unstable. Researchers then designed a series of improvements to resolve this issue from the network structure [IZZE17, KLA19, TMYTCJY19, KLA*20] and loss function [MLX*17, ACB17, GAA*17]. Many works explored the latent space inside a GAN and proposed effective algorithms to edit the outputs by latent manipulation [GAOI19, SGTZ20, YCW*21, VB20, GSZ20]. These methods introduced additional learnable modules to locate the specific visual attributes in the latent space $\mathcal{Z}$ of noise input. However, most of them are tailored for StyleGAN-based architecture [KLA19] and real photo images. Inspired by these works, we adopt a GAN-based architecture and propose the second training, which manipulates reference latent codes to adjust the colours in final outputs.

Style transfer:. Many network-based style transfer methods have been proven efficient in learning features from images. Gatys *et al.* [GEB16] adopted a pre-trained Visual Geometry Group (VGG) network [SZ15, HB17] to transfer style information from a predetermined image. Johnson *et al.* [JAF16] proposed a perceptual loss for training a real-time feed-forward network. GANs soon outperformed these types of networks in various style transfer tasks [IZZE17, XYH*21, WCZ*22, JYTPA17, CUYH20, HLBK18]. As sketch colourization can be regarded as multimodal style transfer, many related algorithms are applicable. To achieve pixel-level correspondence, we utilize the pixel-level L1 loss instead of the frequently used perceptual loss and cycle consistency loss. A feature-level L1 loss is also adopted for latent manipulation in our second training.

Attention in computer vision:. The attention mechanism has dominated the natural language processing (NLP) field [VSP*17, DCLT19] for a long time. Many works have demonstrated the effectiveness of extracting features using spatial and channel-wise attention [HSS18, DBK*21, WPLK18]. We adopt a cross-attention module as our receiving block, which can provide latent codes spatially for the GAN to improve the quality of reference-based results.

Sketch colourization:. Colourizing is very time-consuming in practice, so researchers have developed many assistance tools to accelerate this process, such as Lazybrush [SDC09]. However, traditional methods [SDC09, PMC22, SMYA14, FTR18] are usually developed on the basis of user-guided hints, so they are inappropriate for reference-based colourization. As neural networks have been proven effective in object recognition and colour rendering [ZIE16, ZZI*17, XHZ*20], many deep learning models have been proposed to solve the sketch colourization problem by encoding

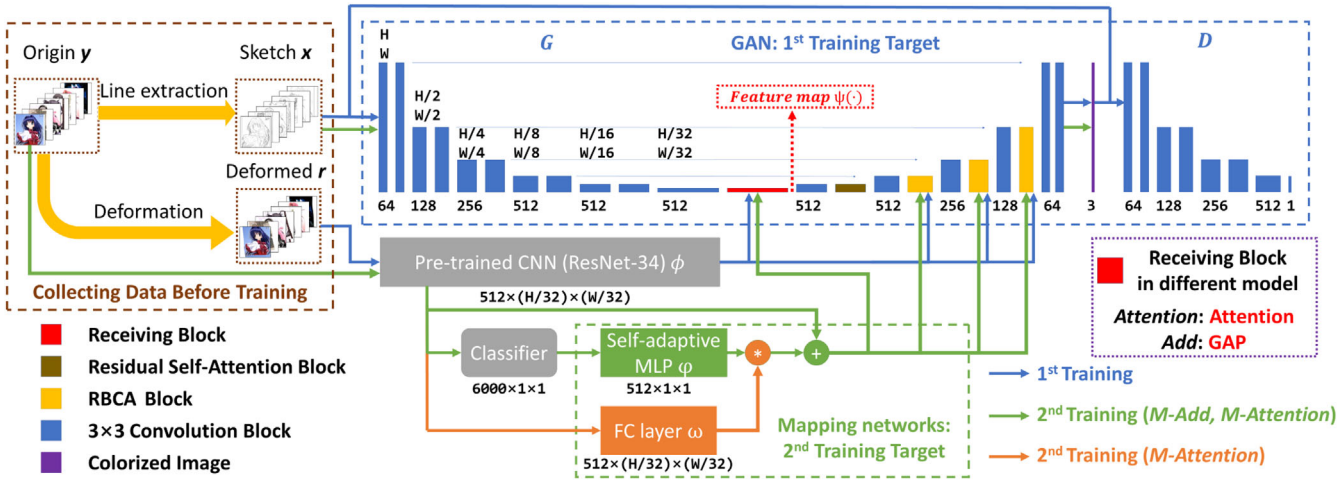


Figure 2: *Illustration of proposed network architecture and training flows. ω is a single fully connected layer, and the orange data flow additionally computes the position weight matrix (PWM) for the* M-Attention *model. **x**, **y** and **r** denote sketch, origin and reference images, respectively; H and W represent the height and width of the input images, respectively; we adopt ReLU and Tanh as the activation function in the intermediate layers and the final layer in the GAN, respectively.*

text [HYES19, ZMG*19, CMG21], user-guided hints [FHOO17, ZLW*18] and reference images [ZJLL17, LKL*20, SLWW19, AMT20] into colour information. Among the reference-based methods, Akita *et al.* proposed a method [AMT20] that can fill empty pupils with reference images, but their method is only effective for eyes in portraits; Style2paints [ZLW*18] can generate satisfactory results by introducing a pre-trained Inception network [CLJ*15], but the colours in their results differ from the reference image and cannot be edited using tags. Inspired by Style2paints, we investigate how a pre-trained encoder benefits reference-based colourization and propose the second training to enable tag-based manipulation. Compared to other two-stage methods [HYES19, ZLW*18], our model enables the spatial latent manipulation and can generate images that are both visually more appealing and editable with reference images and text tags.

3. Method

The pipeline of the proposed architecture is shown in Figure 2, whose training includes two steps: train the GAN for reference-based colourization, and train the mapping networks for tag-based manipulation. We mark the optimization targets of the first and second training with blue and green dotted rectangles, respectively. To obtain a sufficient number of semantically paired images, we generated sketch $\boldsymbol{x}$ and reference images $\boldsymbol{r}$ by applying line extraction [lll17, SSISI16] and deformation [SMW06] to colour images $\boldsymbol{y}$, respectively. We use ResNet-34 to extract latent codes from reference images. ResNet-34 was pre-trained on ImageNet [RDS*15] and Danbooru 2020 [AcB21] for image and multi-label classification, respectively. The top 6000 tags, according to frequency, were adopted for the multi-label training. Note that these tags were not cleaned, so most of them are useless for colourization as they are not colour-related, *e.g.*, 'solo' and '1_girl'. A part of the effective tags is included in the supplementary materials.

3.1. The first GAN training

We utilize conditional GAN (cGAN)

L1 and total variation losses [IZZE17, AD05] to train our colourization GAN. The target of the first GAN training is expressed as

$$\arg\min_{G}\max_{D}\mathcal{L}(G,D)=\mathcal{L}_{cGAN}(G,D)+\lambda_{L1}\mathcal{L}_{L1}(G)+\lambda_{tv}\mathcal{L}_{tv}(G) \tag{1}$$

where hyperparameters $\lambda_{L1}=100$ and $\lambda_{tv}=0.0001$ in accordance with our pre-experiments and Refs. [IZZE17, ZLW*18, JAF16]. A lower value of λ_{L1} was found to decrease the colour diversity of the results, leading to a greyish appearance. The total variation loss is adopted to decrease artifacts, but it is unnecessary for lower resolution images as the artifacts are usually not noticeable. The recommended threshold for cancelling the total variation loss is 384^2.

Conditional adversarial loss:. The cGAN adopts sketch images $\boldsymbol{x}$ as the condition for D [IZZE17]. Latent codes obtained from the reference image are introduced as hints for G. Our cGAN loss can be expressed as

$$\begin{aligned}\mathcal{L}_{cGAN}(G,D)=&\mathbb{E}_{x,y}[\log D(\boldsymbol{x},\boldsymbol{y})]\\&+\mathbb{E}_{x,r}[\log(\boldsymbol{1}-D(\boldsymbol{x},G(\boldsymbol{x},\phi(\boldsymbol{r}))))]\end{aligned} \tag{2}$$

G attempts to generate images to be as real as possible, while D should classify the real/fake image correctly.

Pixel-level L1 loss:. We adopt a pixel-level L1 loss to penalize the difference between ground-truth $\boldsymbol{y}$ and generated images $G(\boldsymbol{x},\phi(\boldsymbol{r}))$. The loss is given as

$$\mathcal{L}_{L1}(G)=\mathbb{E}_{x,y,r}[\|\boldsymbol{y}-G(\boldsymbol{x},\phi(\boldsymbol{r}))\|_1] \tag{3}$$

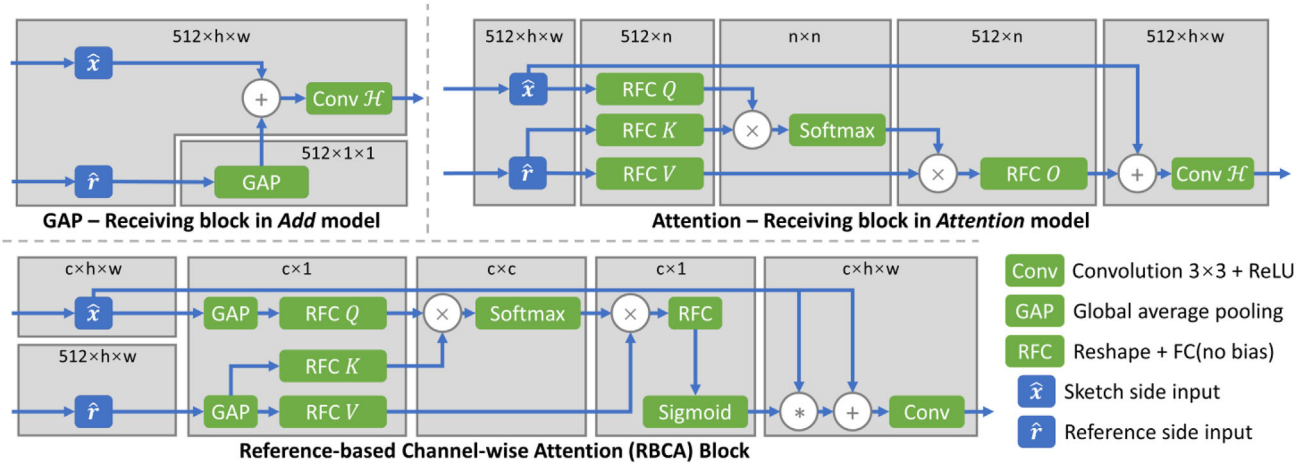


Figure 3: *Illustration of receiving blocks and RBCA blocks. The GAP and attention blocks are used in the* Add *and* Attention *models, respectively. We label the shape of the feature maps at the top of the corresponding grey rectangle, where $n = h \times w$ and c, h, w denote channel, height and width at the corresponding layer, respectively. GAP, global average pooling; RBCA, reference-based channel-wise attention.*

Total variation loss:. To encourage smoothness at high resolution, we follow former works [JAF16, ZYZH10] to utilize total variation regulation [AD05]. $\tilde{\boldsymbol{x}}$ is used to represent $G(\boldsymbol{x}, \phi(\boldsymbol{r}))$ to simplify the expression of the regulation, which is formulated as

$$\mathcal{L}_{tv}(G) = \sum_{i,j} (\|\tilde{\boldsymbol{x}}_{i,j+1} - \tilde{\boldsymbol{x}}_{i,j}\|^2 + \|\tilde{\boldsymbol{x}}_{i+1,j} - \tilde{\boldsymbol{x}}_{i,j}\|^2)^{\frac{\eta}{2}} \tag{4}$$

where $\tilde{\boldsymbol{x}}_{i,j}$ denotes the (i,j)th pixel in colourized result $G(\boldsymbol{x}, \phi(\boldsymbol{r}))$ and η is set to 1 in accordance with Mahendran and Vedaldi [MV15].

3.2. Receiving block

Reference latent codes are input to the GAN through the receiving block, marked in red in Figure 2. To investigate the latent manipulation in the proposed system for the second training, which will be introduced in Section 3.5, we propose two models with different receiving blocks. Details of the receiving blocks are shown in Figure 3 in which the *Attention* and *Add* models adopt an attention block and global average pooling (GAP) block, respectively. Given the sketch side input for the receiving block as $\hat{\boldsymbol{x}}$, its channel size as d_s, the convolution as $\mathcal{H}$ and the output of receiving block as $\psi(\cdot)$, the corresponding latent codes of the GAN $\psi(\boldsymbol{x}, \hat{\boldsymbol{r}})$ can be expressed as

$$\psi(\boldsymbol{x}, \hat{\boldsymbol{r}}) = \begin{cases} \mathcal{H}\left(\hat{\boldsymbol{x}} + O\left(\text{softmax}\left(\frac{Q(\hat{\boldsymbol{x}})K(\hat{\boldsymbol{r}})}{\sqrt{d_s}}\right)V(\hat{\boldsymbol{r}})\right)\right) & \text{for } \textit{Attention} \\ \mathcal{H}(\hat{\boldsymbol{x}} + \text{GAP}(\hat{\boldsymbol{r}})) & \text{for } \textit{Add} \end{cases} \tag{5}$$

where Q, K, V, O represent the linear transformations in the attention-based receiving block. Note that $\hat{\boldsymbol{r}}$ denotes $\phi(\boldsymbol{r})$ or $\boldsymbol{\delta}_b$ in accordance with the reference, and $\boldsymbol{\delta}_b$ will be introduced in Section 3.5.

Different from the *Add* model, which directly adds the globally averaged latent code GAP($\hat{\boldsymbol{r}}$), the *Attention* model indirectly modifies the GAN's latent codes $\psi(\boldsymbol{x}, \hat{\boldsymbol{r}})$ through the attention, where $\psi(\boldsymbol{x}, \hat{\boldsymbol{r}})$ is calculated on the basis of the spatial relationship between $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{r}}$. The *Attention* model performs better in reference-based colourization as attention preserves more local information, while the *Add* model is easier for latent manipulation, which will be discussed in Sections 3.5 and 4.5.

3.3. Reference-based channel-wise attention

Our pre-experiments show the deterioration of colour diversity and similarity caused by missing reference latent codes. A number of latent codes are unnecessary in the first decoding layers, so they are discarded before the corresponding visual attributes are synthesized in the middle layers. To solve this problem, we propose the RBCA block to receive hints in the intermediate upsampling layers, whose position and flow chart are shown in Figures 2 and 3, respectively.

Note that $\hat{\boldsymbol{x}}$ is re-weighted by sigmoid output instead of added. We also adopt a residual connection to ensure a straightforward back-propagation path.

3.4. Pre-trained reference encoder

The widely used perceptual loss [JAF16] and cycle consistency loss [JYTPA17] are insufficient for generating natural colours in sketch colourization, so a pixel-level restriction is necessary when training colourization network. However, training with pixel-level restriction requires pixel-level correspondence between (sketch, colour) pairs and semantic similarity between (reference, colour) pairs. As mentioned at the beginning of Section 3, reference images $\boldsymbol{r}$ were generated by applying deformation $\mathcal{D}$ to colour images $\boldsymbol{y}$, so we can re-write Equation (3) as

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y}[\|\boldsymbol{y} - G(\boldsymbol{x}, \phi(\mathcal{D}(\boldsymbol{y})))\|_1] \tag{6}$$

If G and ϕ are jointly trained, they can be viewed as a united generator G', and the optimization becomes the following process:

$$\arg\min_{G'} \mathcal{L}_{L1}(G') = \mathbb{E}_{x,y}[\|\boldsymbol{y} - G'(\boldsymbol{x}, \mathcal{D}(\boldsymbol{y}))\|_1] \tag{7}$$

We can determine the optimal G' for the re-organized loss to be $\mathcal{D}^{-1}$, the inverse transformation of $\mathcal{D}$ and ignores the sketch input $\boldsymbol{x}$. This leads to a substantial deterioration in inference.

Let F and E denote the decoder and trainable encoder(s), respectively. When jointly training the encoders, the reference-based colourization can be expressed as $\boldsymbol{y} = F(\boldsymbol{z}) \sim P(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{r})$, where $\boldsymbol{z} = E(\boldsymbol{x}, \boldsymbol{r})$, and the latent distribution of $\boldsymbol{z}$ is, therefore, $P(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{r})$. Adopting a pre-trained reference encoder stabilizes this process by dividing it into two steps. First, the sketch encoder generates $\boldsymbol{z}'$ by encoding the sketch image, expressed as $\boldsymbol{z}' = E(\boldsymbol{x})$ and the latent distribution of $\boldsymbol{z}'$ is $P(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})$. Then, the receiving block obtains the embeddings $\boldsymbol{z}$ by conditioning on the reference information, such that $\boldsymbol{z} = \psi(\hat{\boldsymbol{x}}, \hat{\boldsymbol{r}})$ according to Equation (5), where $\hat{\boldsymbol{x}} = \boldsymbol{z}' \sim P(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})$. As the reference encoder is fixed, the optimization target changes from the latent distribution $P(\boldsymbol{z}|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{r})$ to the distribution $P(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})$ and the receiving block. The latent distribution $P(\boldsymbol{z}'|\boldsymbol{y}, \boldsymbol{x})$ is irrelevant to the reference $\boldsymbol{r}$ and decides the image quality. Therefore, this change significantly improves the generated results, particularly compared to the cases when jointly trained encoders fail to match semantically corresponding regions between $\boldsymbol{x}$ and $\boldsymbol{r}$.

Choice for reference encoder:. We tested a series of frequently used networks for the reference encoder. Contrastive language-image pre-training (CLIP) encoders could colourize sketch images but were ineffective in adjusting colours using tags. CNNs other than ResNet-34 were less sensitive to colours because the number of colour-related channels will not increase as the networks become heavier, even though they perform better in segmentation and recognition. Considering the efficiency and quality, we chose ResNet-34 as our default reference encoder and left CLIP encoders for future work. All the networks are trained the same way, which will be explained in Section 4.1.

Implications of poor segmentation:. Our ResNet-34 was not well-trained according to the pre-experiment, as its recall and precision were unsatisfactory. This poor recognition decreases the GAN's segmentation ability and the controllability of results by missing reference latent codes. If a reference latent code is mostly missing during training, the GAN can hardly connect it with the corresponding visual attribute. For example, if the pre-trained CNN cannot precisely predict 'red_skirt', its corresponding visual attribute would be controlled by 'red_dress' or 'red_shirt', as they are all recognized as 'red_cloth'. The experiment in Section 4.5 will partially show this disadvantage.

3.5. The second mapping training

Motivated by latent manipulation research on StyleGAN [KLA19, KLA*20, GAOI19, YCW*21, WLS21], we propose the second training to manipulate the latent codes using probabilistic values of text tags. Previous work has demonstrated that probabilities given by a pre-trained CNN contain sufficient latent information to colourize sketch images, so our second training is tailored to connect these probabilities with the visual features we used in the first training through a network φ that satisfies the following equation:

$$\mathrm{GAP}(\phi(\boldsymbol{r}_t)) - \mathrm{GAP}(\phi(\boldsymbol{r}_a)) \approx \varphi(\boldsymbol{cls}_t) - \varphi(\boldsymbol{cls}_a) \tag{8}$$

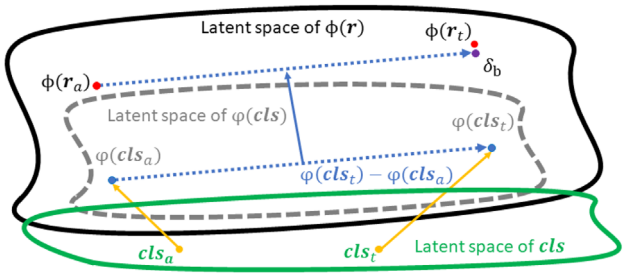


Figure 4: *Illustration of how to approximate $\phi(\boldsymbol{r}_t)$ using δ_b. Converting $\phi(\boldsymbol{r}_a)$ to $\phi(\boldsymbol{r}_t)$ on the basis of the vector distance is better than directly mapping $\varphi(\boldsymbol{cls}_t)$ to $\phi(\boldsymbol{r}_t)$ as it ignores the difference of latent space.*

where $\phi(\boldsymbol{r}_t)$ and $\phi(\boldsymbol{r}_a)$ are the visual features extracted from the target $\boldsymbol{r}_t$ and anchor $\boldsymbol{r}_a$ reference images, respectively, and $\boldsymbol{cls}_t$ and $\boldsymbol{cls}_a$ are their corresponding probabilities given by our pre-trained classifier. Using a neural network to approximate the latent codes makes the manipulation more 'linear', enabling the input probabilities to be larger than 1.

The *Attention* and *Add* models that go through the second training and are combined with the mapping networks φ will be called *M-Attention* and *M-Add* in the following sections, respectively. The *M-Attention* model additionally contains a fully connected layer ω.

Training objects:. The second optimization is defined as

$$\arg\min_{\varphi,\omega} \mathcal{L}(\varphi, \omega) = \mathcal{L}_{hybrid}(\varphi, \omega) + \mathcal{L}_{inv}(\varphi, \omega) \tag{9}$$

where $\mathcal{L}_{inv}$ and ω are tailored for the *M-Attention* model to solve the absence of spatial information in $\varphi(\boldsymbol{cls})$, so they are removed when training the *M-Add* model. Colourization performance is not significantly affected since the second training excludes G from optimization.

The key idea of the second training is to modify the reference latent codes on the basis of the mapped probabilities. To achieve this, we need to look back into the equation $\phi(\boldsymbol{r}) = \frac{\phi(\boldsymbol{r})}{\mathrm{GAP}(\phi(\boldsymbol{r}))} * \mathrm{GAP}(\phi(\boldsymbol{r}))$. We regard $\phi(\boldsymbol{r})$ as a combination of $(\frac{H}{32} \times \frac{W}{32})$ latent codes with H and W representing the height and width of input images, respectively. We can find that $\phi(\boldsymbol{r})$ are separately expressed by a spatial part $\frac{\phi(\boldsymbol{r})}{\mathrm{GAP}(\phi(\boldsymbol{r}))}$ and a content part $\mathrm{GAP}(\phi(\boldsymbol{r}))$. As $\mathrm{GAP}(\phi(\boldsymbol{r}))$ can be approached by $\varphi(\boldsymbol{cls})$, we use biased latent codes $\boldsymbol{\delta}_b$ to approximate the target $\phi(\boldsymbol{r}_t)$, and $\boldsymbol{\delta}_b$ is formulated as

$$\boldsymbol{\delta}_b = \phi(\boldsymbol{r}_a) + \mathcal{F}(\boldsymbol{r}_a) * (\varphi(\boldsymbol{cls}_t) - \varphi(\boldsymbol{cls}_a)) \tag{10}$$

where $\varphi(\boldsymbol{cls})$ is broadcasted to the same shape with $\phi(\boldsymbol{r}_a)$ by replicating the channel values, and $\phi(\boldsymbol{r}_a)$ is added to ensure the consistency between $\phi(\boldsymbol{r}_a)$ and $\boldsymbol{\delta}_b$. Figure 4 illustrates how to approximate the target latent code $\phi(\boldsymbol{r}_t)$ using δ_b. Here, $\mathcal{F}(\boldsymbol{r}_a)$ is calculated as

$$\mathcal{F}(\boldsymbol{r}_a) = \begin{cases} \omega\left(\frac{\phi(\boldsymbol{r}_a)}{\mathrm{GAP}(\phi(\boldsymbol{r}_a))}\right) & \text{for } \textit{M-Attention} \\ \mathbf{I} & \text{for } \textit{M-Add} \end{cases} \tag{11}$$

where $\omega(\frac{\phi(\boldsymbol{r}_a)}{\mathrm{GAP}(\phi(\boldsymbol{r}_a))}) = \boldsymbol{W}\frac{\phi(\boldsymbol{r}_a)}{\mathrm{GAP}(\phi(\boldsymbol{r}_a))} + \boldsymbol{b}$ as it is a linear layer in our design. The ResNet-34 adopts ReLU as the final layer, so $\phi(\boldsymbol{r}_a) \geq \mathbf{0}$

and we can assign $\frac{\phi(\boldsymbol{r}_a)^{(c)}}{\mathrm{GAP}(\phi(\boldsymbol{r}_a))^{(c)}} = \mathbf{I}$ when $\mathrm{GAP}(\phi(\boldsymbol{r}_a))^{(c)} = 0$. $\mathcal{F}(\boldsymbol{r}_a)$ provides spatial latent information for the *M-Attention* model as a position weight matrix (PWM). Accordingly, $\mathcal{F}(\boldsymbol{r}_a) = \mathbf{I}$ for the *M-Add* model since the *Add* model receives globally averaged latent codes, which can be inferred from Figure 3 and Equation (5).

We adopt ω to adjust the spatial part of the latent codes on the basis of its channel-wise relationship. We assume similar latent codes in $\phi(\boldsymbol{r})$, such as 'red_hair' and 'green_hair', should have the same weight when modified by $\varphi(\boldsymbol{cls})$ as they control the same object 'hair'. ω should be a linear transformation to prevent the spatial information in $\frac{\phi(\boldsymbol{r}_a)}{\mathrm{GAP}(\phi(\boldsymbol{r}_a))}$ from being destroyed.

Hybrid L1 loss:. The mapping networks are used to convert $\phi(\boldsymbol{r}_a)$ to $\phi(\boldsymbol{r}_t)$ using the mapped probability vectors, as expressed in Equation (10). To achieve this, we tailor the hybrid L1 loss to maintain the pixel- and feature-level consistency, which is written as

$$\begin{aligned}\mathcal{L}_{hybrid}(\varphi,\omega) = &\underbrace{\mathbb{E}_{\boldsymbol{x},\boldsymbol{r}_t,\delta_b}[\|G(\boldsymbol{x},\phi(\boldsymbol{r}_t)) - G(\boldsymbol{x},\delta_b)\|_1]}_{pixel-level} \\ &+ \underbrace{\mathbb{E}_{\boldsymbol{x},\boldsymbol{r}_t,\delta_b}[\|\psi(\boldsymbol{x},\phi(\boldsymbol{r}_t)) - \psi(\boldsymbol{x},\delta_b)\|_1]}_{feature-level}\end{aligned} \tag{12}$$

The feature-level L1 is the core component for the *M-Add* model as it encourages the generated latent codes to approximate the target $\phi(\boldsymbol{r}_t)$, which can be inferred by combining Equations (5), (10) and (12). The pixel-level L1 penalizes the differences in global attributes, such as 'sky' and 'theme' that are majorly controlled by RBCA blocks.

Inversion loss:. The feature-level L1 loss cannot propagate effective gradients for the mapping network φ in the *M-Attention* model due to the dot product $Q(\hat{\boldsymbol{x}})K(\hat{\boldsymbol{r}})$ in the attention-based receiving block, introduced in Equation (5). To optimize the mapping network φ for the *M-Attention* model, we tailor the inversion loss, which is formulated as

$$\begin{aligned}\mathcal{L}_{inv}(\varphi,\omega) = &\mathbb{E}_{\boldsymbol{r}_t,\boldsymbol{r}_a}[\|(\mathrm{GAP}(\phi(\boldsymbol{r}_t)) - \mathrm{GAP}(\phi(\boldsymbol{r}_a))) \\ &- (\varphi(\boldsymbol{cls}_t) - \varphi(\boldsymbol{cls}_a))\|_1]\end{aligned} \tag{13}$$

The inversion loss directly requires the mapping network to satisfy Equation (8). With the inversion loss, the hybrid L1 loss can optimize ω to modify $\frac{\phi(\boldsymbol{r}_a)}{\mathrm{GAP}(\phi(\boldsymbol{r}_a))}$ on the basis of its channel-wise relationships. However, the inversion loss is different from the feature-level L1 loss, which we will discuss in Section 4.5.

Mapping network φ :. In accordance with our pre-experiments, the visual attributes are controlled by latent codes generated in different layers. For example, 'hair' and 'eyes' are controlled by the first and second fully connected layers, but 'sky' and 'theme' are in the deeper ones, where the 'hair' and 'eyes' related codes will be entangled. The multi-layer MLP consequently loses control of 'hair' and 'eye' and results in entanglement. To solve this problem, we propose a specialized MLP called self-adaptive MLP. Outputs from different layers are concatenated and adaptively weighted, as illustrated in Figure 5.

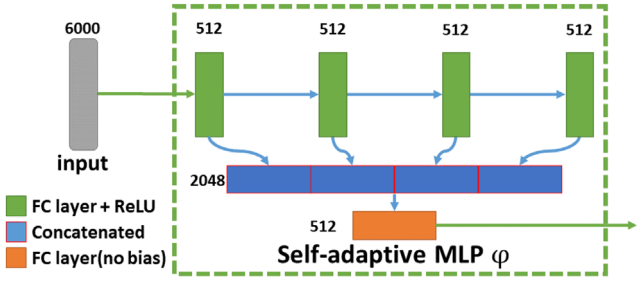


Figure 5: *Self-adaptive MLP used to generate latent codes from classification probabilities. It takes the classification probabilities as input. MLP, multi-layer perceptron.*

4. Experiments

In this section, we first introduce the implementation details of the proposed method in Section 4.1, and then justify our network design in Section 4.2 by an ablation study. In Section 4.3, qualitative and quantitative comparisons with baselines [ZLW*18, SLWW19, CUYH20, LKL*20, HLBK18] are taken to prove the superiority of our method in reference-based colourization. We conducted two user studies, which will be introduced in the same subsection, to investigate the users' preferences and subjectively evaluate the similarity of colours. Finally, we validate the controllability of the proposed *M-Attention* and *M-Add* models through experiments on multi-label manipulation and discuss the differences in latent manipulation between the two models using the experimental results.

We quantitatively evaluate the quality of generated images using the Fréchet inception distance (FID) [HRU*17, Sei20], as a lower FID indicates better image quality. Using the official PyTorch implementation, we computed the FID over the validation dataset 10 times and took the averages. The reference images were shuffled for each evaluation.

4.1. Implementation details

We retrained the CNNs on the multi-label classification dataset Danbooru2020 for two epochs. 855,876 images were collected as source data from Danbooru Figure2019, a subset of Danbooru2020 [AcB21] that only contains figure images. Colour images were resized to 512^2 and used to generate training and validation data, 766,454 and 89,422 triples, respectively. We implemented the framework using PyTorch and trained the proposed models on four Tesla P100s at a batch size of 64 and an NVIDIA GeForce 3090 at a batch size of 32 for nine epochs. Baselines were trained on the 3090 for the same epochs, but their batch sizes were accordingly lowered due to the higher cost of GPU memory. We adopted the Adam optimizer [KB15] with the settings $learning\_rate = 0.0001$, $betas = (0.5, 0.99)$. Input images were randomly rotated, flipped and resized to 384^2 before each iteration, where identical transformations were applied to (sketch, colour) pairs. We excluded validation images from all training sets to ensure that they were only used for evaluation. The second training was taken with the same settings for three epochs.

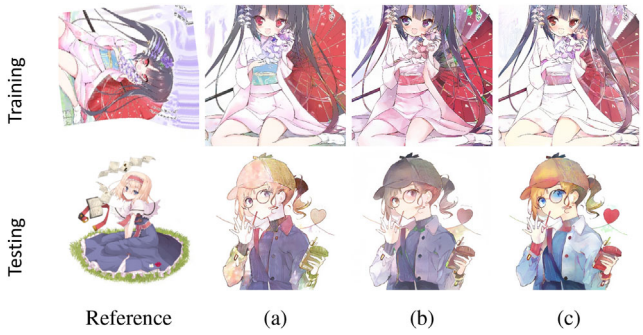


Figure 6: *Comparison of results generated by models using different reference encoders. The reference encoder was (a) jointly trained with GAN, (b) fixed and pre-trained on ImageNet and (c) fixed and pre-trained on ImageNet and Danbooru. GAN, generative adversarial network.*

4.2. Ablation study

Reference encoder:. To justify the adoption of a pre-trained CNN, we trained three models, where ResNet-34 was (a) jointly trained with GAN, (b) fixed and pre-trained on ImageNet and (c) fixed and pre-trained on Danbooru2020 and ImageNet. Samples included in Figure 6 show that (a) generated images with better quality and similarity than (b) and (c) during training but strongly deteriorated in evaluation, whereas the results of (c) are much better than (a) and (b) in both diversity and similarity of colours. In addition to the qualitative experiment, an FID evaluation was conducted and the results are shown in Table 1 for an objective comparison, where a significant improvement can be observed.

RBCA block:. To demonstrate the effectiveness of the RBCA blocks, we performed a quantitative evaluation using the FID. As shown in Table 1, the proposed models achieved better scores by adopting RBCA blocks.

Mapping network:. We objectively show the advantage of the self-adaptive MLP by comparing the feature-level L1 loss, labelled in Equation (6). A lower loss indicates better controllability of the results as it measures the distance between the target $\psi((x), \phi(\boldsymbol{r}))$ and the modified ones $\psi(\boldsymbol{x}, \boldsymbol{\delta}_b)$. To make a better comparison for the *M-Attention* models, we also show the inversion loss used in the second training. As shown in Figure 7, the models with the self-adaptive MLP were better optimized.

4.3. Comparison with baselines

To justify our method in reference-based colourization, we compare our results with those generated by the baselines in this subsection. Our baselines include StarGAN, MUNIT, IconGAN, SCFT and the most important one, Style2paints.

StarGAN [CUYH20] and MUNIT [HLBK18] encode the input images and decode their latent representations with style codes in accordance with the references. IconGAN [SLWW19] adopts separate discriminators for colour and structure. SCFT [LKL*20] obtains the references by deforming the input colour images and records the deformation before each iteration to enable the networks to be trained to find corresponding regions. These baselines jointly train multiple encoders for different styles of input images. Different from these methods, Style2paints [ZLW*18, ZJL20, ZLS*21] adopts two-stage training and a pre-trained InceptionNet. It is an integrated application that requires users to provide hints manually for each input and go through post-processing, so preparing sufficient results of Style2paints for FID evaluation is difficult. We instead conducted two user studies to explore users' preferences for comparison.

Computational cost:. It took 7 h to retrain ResNet-34 on Danbooru2020 for multi-label classification, and the proposed first training cost 45 h on an NVIDIA GeForce 3090. As shown in Table 2, our training time is much less than most baselines owing to the simpler architecture compared with Refs. [CUYH20, HLBK18] or less pre-processing to Lee *et al.* [LKL*20]. Though training IconGAN [SLWW19] is faster than our (CNN pre-training + first training), IconGAN cannot generate high-quality images as our model in accordance with the following comparisons. As the other methods are not capable of tag-based control, we excluded the time of our second training, which took another 8 h, in this comparison.

Table 1: *FID score evaluation for the ablation study and comparison with baseline methods.*

Full setting		w/o RBCA blocks		Ablation reference encoder			
Attention	Add	Attention	Add	D + Train	I + Fix	I + Train	Train
11.2390	11.8531	15.6212	13.6563	23.1408	30.5746	51.3302	57.0753
		Second training scores			Baseline methods		
		M-Attention	M-Add	[CUYH20]	[SLWW19]	[LKL*20]	[HLBK18]
		12.3845	11.9800	34.8503	59.6767	62.1494	121.5455

A lower FID score indicates better quality of the generated image. 'Fix' and 'Train' indicate that the reference encoder is fixed or trained in the colourization training, respectively, and 'D' and 'I' indicate that the reference encoder is pre-trained on Danbooru2020 [AcB21] + ImageNet [RDS*15] or ImageNet only, respectively.
Bold values highlight the best scores.

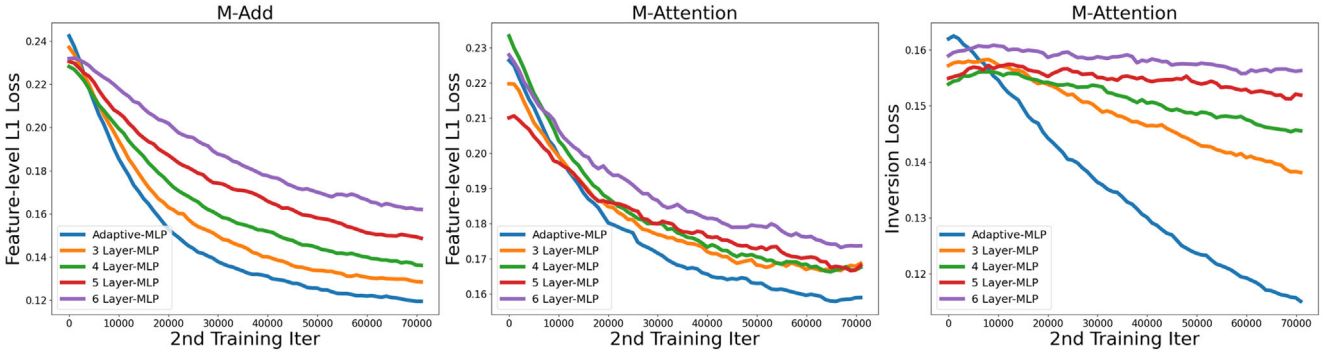


Figure 7: *Comparison of feature-level L1 and inversion losses during training. The losses are smoothed by exponential moving average with the smoothness weight set to 0.9.*

Table 2: *Comparison of training times (days) on an NVIDIA 3090 GPU for reference-based colourization.*

Ours	[CUYH20]	[SLWW19]	[LKL*20]	[HLBK18]
2.2(0.3)	15	2.1	3.2	3.5

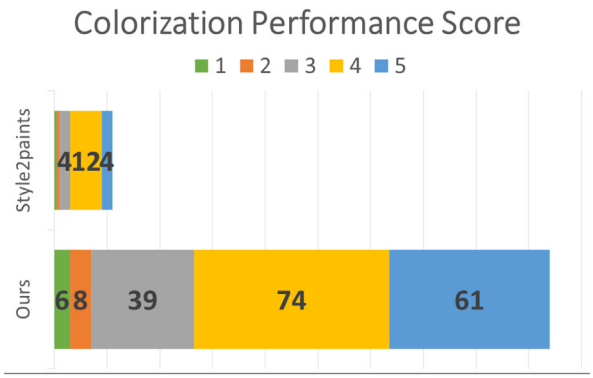
We spent 7 h re-training ResNet-34 for multi-label classification, which is bracketed in the table.

Qualitative comparison:. Colourized images generated by the proposed models and baselines are shown in Figure 8 to prove our advantage in reference-based colourization. It can be seen that only ours and Style2paints produce visually pleasant textures in the results. However, the synthesized colours in Style2paints's results are less similar to the references than ours, especially the eyes and skin. More samples are included in the supplementary materials.

Quantitative comparison:. In addition, we conduct a quantitative experiment using the FID. As shown in Table 1, our method achieved a much lower FID than the baselines. StarGAN v2 [CUYH20], IconGAN [SLWW19], SCFT [LKL*20] and MUNIT [HLBK18] rank from the second to the lowest as they are ineffective in generating visually pleasant colours and maintaining the structure of objects.

User study:. To investigate users' preferences, we used the *Attention* model to conduct two user studies. Since only ours and

Figure 8: *Qualitative comparison with the* Add *model and baseline methods. Sketch images used in the third and fourth rows are manually drawn by a human artist.*



	Ours	Style2paints
Preferred	**188**	22
Preferred (%)	**89.52%**	10.48%
Average Colorization Score	**3.9362**	3.7727

Figure 9: *User study results. The participants are invited to rate the quality of their preferred result from 1 to 5, with 5 as the best. The average colourization score is calculated as* $\sum score / \sum pt$*, where pt denotes the preferred time.*

Table 3: *Average scores in the second user study.*

	Colourization Performance	Similarity
Attention	**4.165**	**3.718**
[ZLW*18]	3.612	2.976
[CUYH20]	3.047	2.541
[SLWW19]	1.624	2.435

The participants needed to rate colourization performance and similarity for each group, which contains a sketch image, reference image and corresponding colourized result.
Bold values highlight the best scores.

Style2paints [ZLW*18] achieved satisfactory results, the first user study was taken for direct comparison between the two methods. The first user study included 10 groups of images, and each group contained a sketch, reference and two colourized images from ours and Style2paints [ZLW*18]. We invited 21 participants to select the preferred image and rate its colourization performance for each group. The first user study result, as shown in Figure 9, indicates that our results are preferred by most participants while achieving a higher score in colourization performance.

Another user study was conducted to compare the proposed method with all baselines. We arranged four questionnaires in the second user study. Each had 20 (sketch, reference, generated) image triples. For example, in questionnaire #1, the respective results used in groups [1–5], [6–10], [11–15] and [16–20] were from ours, Style2paints, StarGAN and IconGAN. We invited 17 participants to rate the quality and similarity of the result for each triple. The average scores and total counting are shown in Table 3 and Figure 10, respectively, and that ours *Attention* achieves the highest scores in

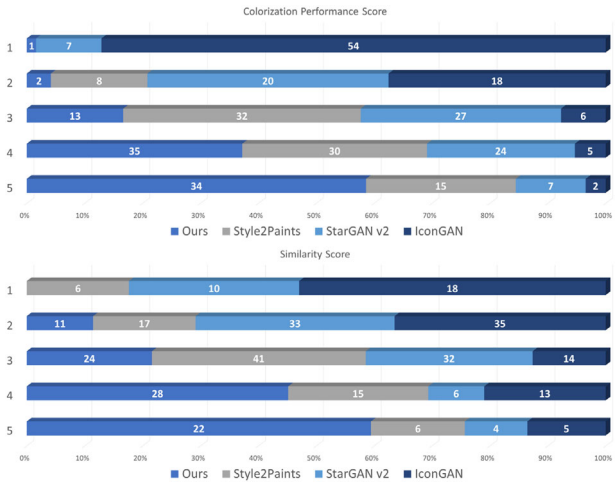


Figure 10: *Rating score distribution in the second user study. Higher score indicates better performance.*

both evaluations. Samples of the user studies are included in the supplementary materials.

4.4. Multi-label manipulation

To investigate the controllability of the proposed models, we performed multi-attribute manipulation by changing the values of hair-related tags, as shown in Figure 11. The values increase along the axes, where the progressive change of hair colour can be observed in the results. We then tested the disentanglement and effectiveness for global attributes as shown in Figure 12, where we can see that the global hue of the images is modified on the basis of the manipulation of 'sky' labels without influencing the eyes and hair. To explore the manipulation linearity, we generated two sets of results using the *M-Attention* and *M-Add* models, respectively, as shown in Figure 13. These samples demonstrate the effective control for values larger than 1. According to our experiments, the input values can be approximately [0, 5].

These experimental results qualitatively prove the controllability of our models; however, there are a number of differences between the *M-Attention* and *M-Add* models, which we will discuss in the next subsection.

4.5. Difference between two mapping models

As introduced in Equation (5), the *M-Attention* model modifies the $\psi(\boldsymbol{x}, \boldsymbol{\delta}_b)$ on the basis of the dot product $Q(\hat{\boldsymbol{x}})K(\boldsymbol{\delta}_b)$. To investigate the information included in $\hat{\boldsymbol{x}}$, we show a result generated without reference $\hat{\boldsymbol{r}}$ in the first column in Figure 14, which indicates that texture and identity are synthesized before receiving the references. Therefore, if a number of latent codes are missing in $\hat{\boldsymbol{x}}$, $\boldsymbol{\delta}_b$ would be ineffective in manipulating related visual attributes.

While the *Add* model ignores the spatial similarity, which is computed by the dot product in attention, the feature-level L1 loss in Equation (6) can map probabilities $\boldsymbol{cls}$ into the latent space of the

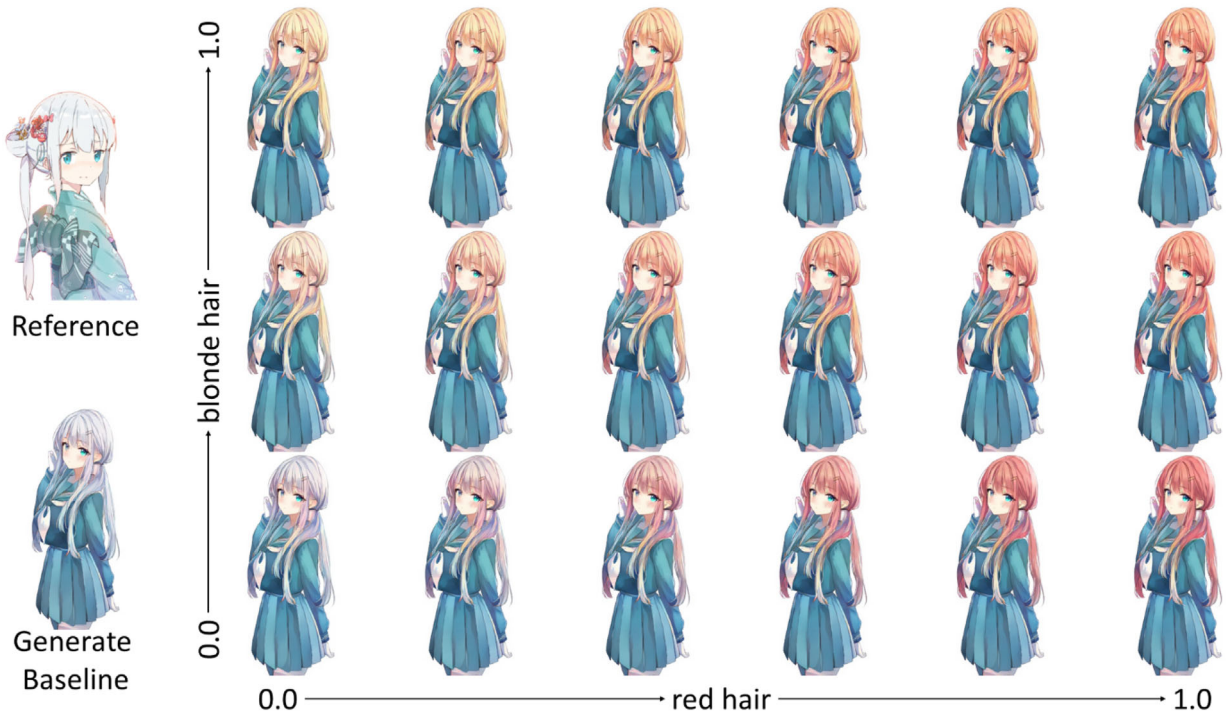


Figure 11: *Multi-attribute results rendered by the* M-Attention *model. From left to right, the respective 'red_hair' values are {0.0, 0.2, 0.4, 0.6, 0.8, 1.0}, and from bottom to top, the respective 'blonde_hair' values are {0, 0.5, 1.0}.*

Figure 12: *Multi-attribute results generated by respective models. All results are generated with 'red_hair' = 2.0 and 'yellow_eyes' = 2.0. The sky labels can control the global hue of the generated image. Background and theme labels have a similar effect, such as 'simple_background', 'red_background' and 'green_theme', 'red_theme', respectively.*

GAN. In other words, *M-Add* can directly recover the missing latent codes. The shirt colour clearly shows the difference in Figure 14, as most of the shirt in *M-Attention*'s result is not colourized as red. This entanglement can be diminished by improving the pre-trained CNN, as explained in Section 3.4.

Another difference caused by the dot product can be observed by comparing the hair colour vertically in Figure 13. Hair colours of the *M-Attention* model's results seem to be rendered by scaling the reference-based one. In contrast, the *M-Add* model's results are more likely to be shifted.

Figure 13: *Multi-attribute results generated by the proposed* M-Attention *and* M-Add *models, where the baseline columns show the respective reference-based results. The manipulated tags are 'blue_shirt', 'green_hair' and 'yellow_eyes'.*

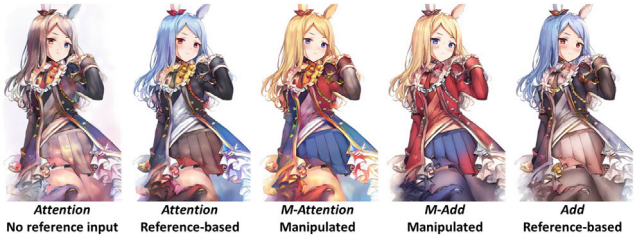


Figure 14: *Samples generated by the* M-Attention *and* M-Add *models, respectively. The values are given as 'purple_eyes' = 3.0, 'red_shirt' = 2.0, 'blue_skirt' =2.0.*

5. Conclusions

We have presented a novel system for reference-based sketch colourization, which can generate visually pleasant and adjustable results by adopting a pre-trained CNN as the reference encoder. We demonstrated the effectiveness of the proposed RBCA block and self-adaptive MLP in colourization through an ablation study. Qualitative/quantitative comparisons and user studies with the baselines were taken to show our advantages in reference-based colourization. We also showed experimental results on multi-label manipulation to demonstrate the controllability of our models and to investigate spatial latent manipulation.

However, there are still a number of limitations. First, the performance of colourization deteriorates as the line density of the input sketch images decreases, resulting in a loss of texture information. While most generative models rely on noise inputs to compensate for this missing information when applied to single-condition generation, our model is designed for dual-condition generation (sketch + reference image or sketch + text tags) and eschews this approach due to its negative impact on the stability of training and image quality. Second, our ResNet-34 is inefficient in multi-label classification as we found it performs much worse than DeepDanbooru [Kic], which is too heavy for our research, and this drawback decreases the GAN's segmentation ability. Finally, the proposed *M-Attention* model cannot directly manipulate the latent code in the GAN, which may degrade the controllability of the results, as discussed in Section 4.5. Improving the generative model and adopting well-trained CLIP encoders will be the key points of our future work.

Acknowledgements

The authors have nothing to report.

References

[ACB17] Arjovsky M., Chintala S., Bottou L.: Wasserstein generative adversarial networks. In *Proceedings of the International Conference on Machine Learning, ICML* (Aug. 2017), D. Precup and Y. W. Teh (Eds.), PMLR, vol. *70*, pp. 214–223.

[AcB21] Anonymous, Community D., Branwen G.: Danbooru2020: A large-scale crowdsourced and tagged anime illustration dataset. https://www.gwern.net/Danbooru2020 (2021). Accessed: 2021-03-13.

[AD05] Aly H. H., Dubois E.: Image up-sampling using total-variation regularization with a new observation model. *IEEE Transactions on Image Processing 14*, 10 (2005), 1647–1659.

[AMT20] Akita K., Morimoto Y., Tsuruno R.: Colorization of line drawings with empty pupils. *Computer Graphics Forum 39*, 7 (2020), 601–610.

[CLJ*15] Christian S., Liu W., Jia Y., Pierre S., Scott R., Dragomir A., Dumitru E., Vanhoucke V., Rabinovich A.:

Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2015), IEEE Computer Society, pp. 1–9. https://doi.org/10.1109/CVPR.2015.7298594

[CMG21] Cao R., Mo H., Gao C.: Line art colorization based on explicit region segmentation. *Computer Graphics Forum 40*, 7 (2021), 1–10.

[CUYH20] Choi Y., Uh Y., Yoo J., Ha J.: Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2020), Computer Vision Foundation/IEEE, pp. 8185–8194. https://doi.org/10.1109/CVPR42600.2020.00821

[DBK*21] Dosovitskiy A., Beyer L., Kolesnikov A., Weissenborn D., Zhai X., Unterthiner T., Dehghani M., Minderer M., Heigold G., Gelly S., Uszkoreit J., Houlsby N.: An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations ICLR* (2021), OpenReview.net.

[DCLT19] Devlin J., Chang M., Lee K., Toutanova K.: BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT* (2019), J. Burstein, C. Doran and T. Solorio (Eds.), Association for Computational Linguistics, pp. 4171–4186. https://doi.org/10.18653/v1/n19-1423

[FHOO17] Furusawa C., Hiroshiba K., Ogaki K., Odagiri Y.: Comicolorization: Semi-automatic manga colorization. In *SA'17: SIGGRAPH Asia 2017 Technical Briefs* (New York, NY, USA, 2017), Association for Computing Machinery. https://doi.org/10.1145/3145749.3149430

[FTR18] Fourey S., Tschumperlé D., Revoy D.: A fast and efficient semi-guided algorithm for flat coloring line-arts. In *Vision, Modeling and Visualization VMV* (2018), Eurographics Association, pp. 1–9. https://doi.org/10.2312/vmv.20181247

[GAA*17] Gulrajani I., Ahmed F., Arjovsky M., Dumoulin V., Courville A.: Improved training of wasserstein GANs. In *Proceedings of the International Conference on Neural Information Processing Systems, NeurIPS* (Red Hook, NY, USA, 2017), Curran Associates Inc., pp. 5769–5779.

[GAOI19] Goetschalckx L., Andonian A., Oliva A., Isola P.: Ganalyze: Toward visual definitions of cognitive image properties. In *Proceedings of the IEEE International Conference on Computer Vision, ICCV* (2019), IEEE, pp. 5743–5752. https://doi.org/10.1109/ICCV.2019.00584

[GEB16] Gatys L. A., Ecker A. S., Bethge M.: Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2016), IEEE Computer Society, pp. 2414–2423. https://doi.org/10.1109/CVPR.2016.265

[GPM*14] Goodfellow I. J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A. C., Bengio Y.: Generative adversarial nets. In *Advances in Neural Information Processing Systems* (2014), Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence and K. Weinberger (Eds.), Curran Associates, Inc., vol. *27*.

[GSZ20] Gu J., Shen Y., Zhou B.: Image processing using multi-code GAN prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2020), Computer Vision Foundation/IEEE, pp. 3009–3018. https://doi.org/10.1109/CVPR42600.2020.00308

[HB17] Huang X., Belongie S. J.: Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision, ICCV* (2017), IEEE Computer Society, pp. 1510–1519. https://doi.org/10.1109/ICCV.2017.167

[HLBK18] Huang X., Liu M., Belongie S. J., Kautz J.: Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision, ECCV. Lecture Notes in Computer Science* (2018), Springer, vol. *11207*, pp. 179–196. https://doi.org/10.1007/978-3-030-01219-9_11

[HRU*17] Heusel M., Ramsauer H., Unterthiner T., Nessler B., Hochreiter S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Proceedings of the International Conference on Neural Information Processing Systems, NeurIPS* (2017), pp. 6626–6637.

[HSS18] Hu J., Shen L., Sun G.: Squeeze-and-excitation networks. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 7132–7141. https://doi.org/10.1109/CVPR.2018.00745

[HYES19] Hyunsu K., Young J. H., Eunhyeok P., Sungjoo Y.: Tag2pix: Line art colorization using text tag with secat and changing loss. In *Proceedings of the IEEE International Conference on Computer Vision, ICCV* (2019), pp. 9055–9064. https://doi.org/10.1109/ICCV.2019.00915

[IZZE17] Isola P., Zhu J., Zhou T., Efros A. A.: Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2017), IEEE Computer Society, pp. 5967–5976. https://doi.org/10.1109/CVPR.2017.632

[JAF16] Johnson J., Alahi A., Fei-Fei L.: Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision, ECCV. Lecture Notes in Computer Science* (2016), B. Leibe, J. Matas, N. Sebe and M. Welling (Eds.), Springer, vol. *9906*, pp. 694–711. https://doi.org/10.1007/978-3-319-46475-6_43

[JYTPA17] Jun-Yan Z., Taesung P., Phillip I. A. E. A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International*

Conference on Computer Vision, ICCV (2017), pp. 2242–2251. https://doi.org/10.1109/ICCV.2017.244

[KB15] Kingma D. P., Ba J.: Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations, ICLR* (2015), Y. Bengio and Y. LeCun (Eds.).

[Kic] KichangKim: DeepDanbooru: Anime-style girl image tag estimation system. https://github.com/KichangKim/DeepDanbooru (2022). Accessed: 2022-01-12.

[KLA19] Karras T., Laine S., Aila T.: A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2019), pp. 4396–4405. https://doi.org/10.1109/CVPR.2019.00453

[KLA*20] Karras T., Laine S., Aittala M., Hellsten J., Lehtinen J., Aila T.: Analyzing and improving the image quality of StyleGAN. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2020), Computer Vision Foundation/IEEE, pp. 8107–8116. https://doi.org/10.1109/CVPR42600.2020.00813

[LKL*20] Lee J., Kim E., Lee Y., Kim D., Chang J., Choo J.: Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2020), Computer Vision Foundation/IEEE, pp. 5800–5809. https://doi.org/10.1109/CVPR42600.2020.00584

[lll17] lllyasviel: Sketchkeras. https://github.com/lllyasviel/sketchKeras (2017). Accessed: 2022-01-12.

[MLX*17] Mao X., Li Q., Xie H., Lau R. Y., Wang Z., Smolley S. P.: Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision, ICCV* (2017), IEEE Computer Society, pp. 2813–2821. https://doi.org/10.1109/ICCV.2017.304

[MV15] Mahendran A., Vedaldi A.: Understanding deep image representations by inverting them. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2015), IEEE Computer Society, pp. 5188–5196. https://doi.org/10.1109/CVPR.2015.7299155

[PMC22] Parakkat A. D., Memari P., Cani M.-P.: Delaunay painting: Perceptual image colouring from raster contours with gaps. *Computer Graphics Forum 41*, 6 (2022), 166–181.

[RDS*15] Russakovsky O., Deng J., Su H., Krause J., Satheesh S., Ma S., Huang Z., Karpathy A., Khosla A., Bernstein M. S., Berg A. C., Fei-Fei L.: Imagenet large scale visual recognition challenge. *International Journal of Computer Vision 115*, 3 (2015), 211–252.

[SDC09] Sýkora D., Dingliana J., Collins S.: LazyBrush: Flexible painting tool for hand-drawn cartoons. *Computer Graphics Forum 28*, 2 (2009), 599–608.

[Sei20] Seitzer M.: pytorch-fid: FID Score for PyTorch. Version 0.2.1. https://github.com/mseitzer/pytorch-fid (Aug. 2020). Accessed: 2022-01-12.

[SGTZ20] Shen Y., Gu J., Tang X., Zhou B.: Interpreting the latent space of GANs for semantic face editing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2020), Computer Vision Foundation/IEEE, pp. 9240–9249. https://doi.org/10.1109/CVPR42600.2020.00926

[SLWW19] Sun T., Lai C., Wong S., Wang Y.: Adversarial colorization of icons based on contour and color conditions. In *Proceedings of the ACM International Conference on Multimedia MM* (2019), L. Amsaleg, B. Huet, M. A. Larson, G. Gravier, H. Hung, C. Ngo and W. T. Ooi (Eds.), ACM, pp. 683–691. https://doi.org/10.1145/3343031.3351041

[SMW06] Schaefer S., McPhail T., Warren J. D.: Image deformation using moving least squares. *ACM Transactions on Graphics 25*, 3 (2006), 533–540.

[SMYA14] Sato K., Matsui Y., Yamasaki T., Aizawa K.: Reference-based manga colorization by graph correspondence using quadratic programming. In *SA'14: SIGGRAPH Asia 2014 Technical Briefs* (New York, NY, USA, 2014), Association for Computing Machinery. https://doi.org/10.1145/2669024.2669037

[SSISI16] Simo-Serra E., Iizuka S., Sasaki K., Ishikawa H.: Learning to simplify: Fully convolutional networks for rough sketch cleanup. *ACM Transactions on Graphics 35*, 4 (2016). https://doi.org/10.1145/2897824.2925972

[SZ15] Simonyan K., Zisserman A.: Very deep convolutional networks for large-scale image recognition. In *Proceedings of International Conference on Learning Representations, ICLR* (2015).

[TMYTCJY19] Taesung P., Ming-Yu L., Ting-Chun W., Jun-Yan Z.: Semantic image synthesis with spatially-adaptive normalization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2019), Computer Vision Foundation/IEEE, pp. 2332–2341. https://doi.org/10.1109/CVPR.2019.00244

[VB20] Voynov A., Babenko A.: Unsupervised discovery of interpretable directions in the GAN latent space. In *Proceedings of the International Conference on Machine Learning, ICML* (2020), PMLR, vol. *119*, pp. 9786–9796.

[VSP*17] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L., Polosukhin I.: Attention is all you need. In *Proceedings of the International Conference on Neural Information Processing Systems, NeurIPS* (2017), I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan and R. Garnett (Eds.), pp. 5998–6008.

[WCZ*22] Wei T., Chen D., Zhou W., Liao J., Tan Z., Yuan L., Zhang W., Yu N.: HairCLIP: Design your hair by text and reference image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2022), pp. 18051–18060. https://doi.org/10.1109/CVPR52688.2022.01754

[WLS21] Wu Z., Lischinski D., Shechtman E.: Stylespace analysis: Disentangled controls for StyleGAN image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2021), Computer Vision Foundation/IEEE, pp. 12863–12872. https://doi.org/10.1109/CVPR46437.2021.01267

[WPLK18] Woo S., Park J., Lee J., Kweon I. S.: CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision, ECCV. Lecture Notes in Computer Science* (2018), Springer, vol. *11211*, pp. 3–19. https://doi.org/10.1007/978-3-030-01234-2_1

[XHZ*20] Xiao C., Han C., Zhang Z., Qin J., Wong T., Han G., He S.: Example-based colourization via dense encoding pyramids. *Computer Graphics Forum 39*, 1 (2020), 20–33.

[XYH*21] Xiao C., Yu D., Han X., Zheng Y., Fu H.: SketchHairSalon: Deep sketch-based hair image synthesis. *ACM Transactions on Graphics 40*, 6 (2021), 216:1–216:16.

[YCW*21] Yang H., Chai L., Wen Q., Zhao S., Sun Z., He S.: Discovering interpretable latent space directions of GANs beyond binary attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2021), Computer Vision Foundation/IEEE, pp. 12177–12185. https://doi.org/10.1109/CVPR46437.2021.01200

[ZIE16] Zhang R., Isola P., Efros A. A.: Colorful image colorization. In *Proceedings of the European Conference on Computer Vision, ECCV* (2016), Springer, vol. *9907*, pp. 649–666. https://doi.org/10.1007/978-3-319-46487-9_40

[ZJL20] Zhang L., Ji Y., Liu C.: DanbooRegion: An illustration region dataset. In *Proceedings of the European Conference on Computer Vision, ECCV. Lecture Notes in Computer Science* (2020), A. Vedaldi, H. Bischof, T. Brox and J. Frahm (Eds.), Springer, vol. *12358*, pp. 137–154. https://doi.org/10.1007/978-3-030-58601-0_9

[ZJLL17] Zhang L., Ji Y., Lin X., Liu C.: Style transfer for anime sketches with enhanced residual U-net and auxiliary classifier GAN. In *Proceedings of the ACPR* (Nanjing, China, 2017), IEEE Computer Society, pp. 506–511. https://doi.org/10.1109/ACPR.2017.61

[ZLS*21] Zhang L., Li C., Simo-Serra E., Ji Y., Wong T., Liu C.: User-guided line art flat filling with split filling mechanism. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (2021), Computer Vision Foundation/IEEE, pp. 9889–9898. https://doi.org/10.1109/CVPR46437.2021.00976

[ZLW*18] Zhang L., Li C., Wong T., Ji Y., Liu C.: Two-stage sketch colorization. *ACM Transactions on Graphics 37*, 6 (2018), 261:1–261:14.

[ZMG*19] Zou C., Mo H., Gao C., Du R., Fu H.: Language-based colorization of scene sketches. *ACM Transactions on Graphics 38*, 6 (2019). https://doi.org/10.1145/3355089.3356561

[ZYZH10] Zhang H., Yang J., Zhang Y., Huang T. S.: Non-local kernel regression for image and video restoration. In *Proceedings of the European Conference on Computer Vision, ECCV* (2010), Springer, pp. 566–579. https://doi.org/10.1007/978-3-642-15558-1_41

[ZZI*17] Zhang R., Zhu J.-Y., Isola P., Geng X., Lin A. S., Yu T., Efros A. A.: Real-time user-guided image colorization with learned deep priors. *ACM Transactions on Graphics 36*, 4 (2017). https://doi.org/10.1145/3072959.3073703

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Github: https://github.com/ydk-tellurion/sketch_colorizer