Line Art Colorization with Offset Prior-based Diffusion Model

Xuan Zhu[1], Miao Cao[2, 3], Fang-Lue Zhang[1, *], Yu-Kun Lai[4], Paul L Rosin[4]
[1]School of Engineering and Computer Science, Victoria University of Wellington
[2]State Key Lab of Multimedia Info. Processing, School of Computer Science, Peking University
[3]National Eng. Research Ctr. of Visual Technology, School of Computer Science, Peking University
[4]School of Computer Science and Informatics, Cardiff University

Figure 1. Our method uses two keyframe reference images and target line arts to colorize line art sequences. The produced results preserve consistent colors and fine details across all target frames.

Abstract

Reference-based line art video colorization colorizes the target line art according to reference images, which is an essential stage for the cartoon production workflow. However, the manual colorization process is time-consuming and repetitive, making automatic video colorization highly desirable. Existing cartoon colorization methods struggle with domain misalignment between the reference and line art images and the loss of details caused by compression into a low-dimensional space in the existing video diffusion models, reducing colorization quality. In this paper, we propose an Offset Prior-based Diffusion Model (OPDM) for cartoon video colorization, which utilizes the powerful generation capability of the diffusion model and cross-domain matching priors to generate high-quality colorization results. Specifically, we design a simple and effective Offset-Adapter that leverages the idea of sampling offsets in deformable convolution to estimate the cross-domain spatial offset features between the target line arts and reference images. We further introduce a new training strategy that combines forward diffusion and reverse denoising in the training stage to ensure content consistency. Experiments on a public cartoon dataset and our newly constructed long cartoon video dataset demonstrate that our proposed method outperforms the existing state-of-the-art line art coloring methods. Code is available at https://github.com/xzh976/OPDM.

1. Introduction

Cartoon video colorization is an important process in the cartoon production workflow. Artists colorize a series of line art images based on the color keyframes, which is a process that needs to be repeated for each frame. Thus, colorization is a time-consuming and labor-intensive task. Automatic colorization methods based on reference frames are desired to reduce animation production costs.

Existing methods use Generative Adversarial Network (GAN) models [11] and diffusion models [13, 14] for the colorization task. GAN-based methods [33, 38, 40, 47] usually combine similarity calculation modules to align the target line arts and the reference images, which may cause mismatches between the lines and the color regions, leading to inaccurate cross-domain correspondence estimation. The GAN-based methods also suffer from the unstable training process. Optical flow estimation methods [37] struggle to accurately extract motion features due to the sparsity of features in line art images. Methods based on large pre-trained stable video diffusion models [1] operate in a compressed latent space, which often causes the loss of fine-grained de-

* Corresponding author.

tails and unclear edges.

To address the above challenges, we propose a unified guided diffusion model named the **O**ffset **P**rior-based **D**iffusion **M**odel (OPDM) for the cartoon video colorization task. Specifically, two types of reference features (past and future keyframes) and offset features are used to generate the target colorization images based on the target line arts. In particular, the offset features are generated between successive frames using a novel Offset-Adapter module based on deformable convolutions, which learns the spatial alignment between the target line arts and reference images to guide the diffusion model in reconstructing the colorized images. During training, we incorporate both the forward diffusion process and the reverse denoising process. This model forms a reconstruction architecture that ensures the consistency and quality of the generated content. Finally, experiments on ATD-12K [35] and our newly collected long cartoon video dataset (AniColorSet) demonstrate that the proposed method outperforms existing state-of-the-art (SOTA) methods.

Our main contributions are summarized as follows:

- We propose an offset prior-based diffusion model that leverages the powerful generative capabilities of diffusion models, integrating multiple sources of information to achieve high-quality line art colorization.
- We introduce a new training scheme for diffusion-based colorization, which combines the forward diffusion and reverse denoising processes in the training stage to ensure the consistency of the generated content.
- We design an effective Offset-Adapter that generates the diverse offsets through the idea of offset in the deformable convolution to provide cross-domain matching priors.
- We conduct extensive experiments on public animation datasets and a long-video dataset collected by ourselves to validate the effectiveness of our method.

2. Related Work

Cartoon Image Colorization. Reference-based cartoon image colorization aims to transfer the colors from reference images to cartoon line arts. Since line art is composed of sparse lines and lacks texture information, directly applying colorization methods designed for natural images often leads to poor results. With the development of learning-based approaches, GAN-based methods have achieved better coloring performance compared to traditional colorization methods. These GAN-based models typically integrate cross-modal modules to transfer colors from the reference image to the corresponding semantic regions in the line art. Lee *et al*. [21] proposed utilizing the self-augmentation method [2] to generate geometrically distorted but identical images, which helps address the problem of insufficient training data. They also designed an attention-based pixel-wise feature transfer module to learn which parts of the reference image should be transferred to which parts of the line art. Yoo *et al*. [46] introduced a memory network using key-value memory to extract color information relevant to the target region and proposed a novel triplet loss with unsupervised learning, eliminating the need for class labels. Li *et al*. [23] proposed a novel attention mechanism with gradient modulation to eliminate gradient conflicts during training. Liu *et al*. [24] presented a reference-guided structure-aware colorization method that preserves structural lines and improves the visual realism of both global color composition and local colorization. Cao *et al*. [3] introduced a channel-wise and spatial-wise convolutional attention module to enhance feature extraction, and employed a stop-gradient attention module combining cross-attention and self-attention to capture cross-domain semantic correlations.

Cartoon Video Colorization. For reference-based cartoon video colorization, directly applying cartoon image colorization methods or natural video colorization methods [22, 25, 49] fails to achieve satisfactory results, because cartoon video sequences often exhibit exaggerated and discontinuous changes across frames. Thasarathan *et al*. [38] concatenated the previous frame and the target line art as input to the discriminator to guide the target line art colorization. Zhang *et al*. [47] designed a matching module that computes the similarity between line arts and uses the reference image together with the similarity scores to infer target color features. Casey *et al*. [6] proposed a transformer-based method to learn the spatial and visual relationships between segments across sequences of cartoon line art. Shi *et al*. [33] introduced a distance attention layer to transfer colors from the reference image to the line art based on similarity, and designed a refinement network based on a 3D convolutional architecture to improve temporal consistency. Wang *et al*. [40] proposed a transformation region enhancement network, which incorporates a region localization module and a feature enhancement module to better colorize regions with geometric transformations.

Diffusion-based Cartoon Colorization. Diffusion models employing denoising diffusion probabilistic models (DDPMs) and latent diffusion models (LDMs) [32] have achieved impressive results in different computer vision tasks such as video editing [7], super-resolution [42], inpainting [28], and deblurring [31]. With the widespread application of diffusion models in the natural image colorization task, Cao *et al*. [4] first introduced the diffusion model into the anime face line art colorization. Aiming to address the huge training consumption problem of the diffusion model, they developed a hybrid training strategy including the pre-training and fine-tuning stages, which showed good performance with a few iterations of fine-tuning. Carrillo *et al*. [5] proposed a user-guided colorization method based on a diffusion model, where the initial color strokes are fed into the diffusion model through cross-attention.

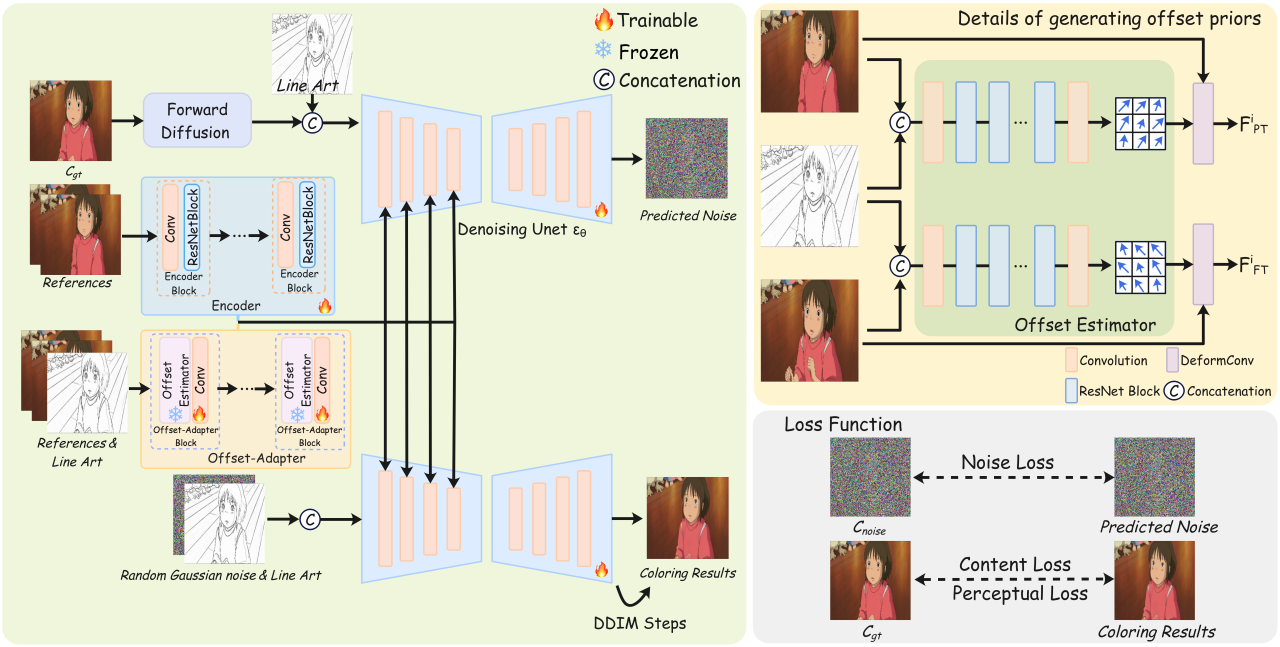

Figure 2. The overall framework of our proposed OPDM. The target color image first undergoes the forward diffusion process, and the target line art is concatenated as the image-level condition input. For the feature-level input, reference features from the previous and future keyframes are extracted through an encoder. The Offset-Adapter generates offset features from the two reference frames and the target line art. Guided by the line art, the reference features, and the offset features, OPDM reconstructs the target color image through a combined forward diffusion and reverse denoising process during training. Content and perceptual losses are introduced alongside the denoising loss for training the model. In the inference stage, only the reverse denoising process is performed to generate the final colorized image.

Huang *et al*. [15] proposed the first pretrained stable video diffusion-based colorization method to generate long cartoon videos with high quality and temporally consistent. Similarly, AniDoc [29] also used pretrained stable video diffusion, and proposed correspondence guidance to handle misalignment, binarized line arts to imitate real-word line art, and the sparse line art training method to reduce the reliance on dense line arts. Liu *et al*. [26] proposed a stable diffusion-based line art colorization method with a patch shuffling module and point-driven control scheme to achieve precise matching.

3. Method

3.1. Motivation and Proposed Method

With the development of diffusion models, they have become a mainstream framework for cartoon colorization tasks. However, the sampling process of diffusion models always starts with Gaussian noise, which contributes to the diversity of generated content. Additionally, there is a misalignment problem across different image domains, leading to unsatisfactory colorization results. Using only the first frame as a reference often results in coloring errors when new objects appear. To address these issues, we propose OPDM, as illustrated in Figure 2. We leverage three types of conditions to guide the diffusion model in reconstructing high-quality colorization content, aimed at transferring the color from reference images to target line arts. Firstly, we employ an encoder to produce multiscale reference features as feature-level input, while target line arts are directly fed into OPDM as image-level input. Next, we introduce the Offset-Adapter, based on deformable convolution [10], to provide explicit spatial correspondence as additional feature-level input. Finally, we combine the forward diffusion and reverse denoising processes to reconstruct the target colorization content and define the loss function.

3.2. Offset-Adapter

Given two reference images and target line arts, the standard diffusion model can learn not only color transfer but also the alignment between references and line arts. However, its implicit spatial matching ability is limited, as it relies on feature interactions, which can lead to incorrect color propagation. To address this issue, we propose a new module, named the Offset-Adapter, which provides explicit spatial correspondence priors. This module is based on the properties of deformable convolution network (DCN), as shown in Figure 2. DCN was originally proposed to handle geometric changes in objects. In recent years, several video super-resolution tasks [9, 39, 41] have demonstrated that DCN performs well in aligning multiple frames, due to the

diverse offsets introduced at each position in the DCN.

Therefore, we construct four Offset-Adapter blocks, each consisting of offset estimators and 2D convolutional layers, to generate offset features at different scales. First, we design the offset estimator to estimate the offsets between the target line art and the reference images, as shown in Figure 2. Specifically, we build a parallel double-branch structure to process the previous and future reference frames separately. For each branch, the target line art and the corresponding reference frame are concatenated as input to the offset estimator. The input first passes through a 2D convolution layer followed by five residual blocks, which adopt the standard bottleneck design with skip connections. Then, a 2D convolution is applied to produce the respective offset features, with $2 \times N \times N$ channels, as described in Eq. 1. In this context, the two dimensions represent the horizontal and vertical coordinates of each position within the corresponding convolution kernel. The offset estimator takes the target line arts and the two references as input to compute the sampling parameters as follows:

$$\Theta = f_\theta(F^i_{Ref}, F^i_l), i \in \{1, 2, 3, 4\}, \tag{1}$$

where Θ represents the learnable offsets, F^i_{Ref} denotes reference image features at different scales, and F^i_l represents target line art features at different scales. For example, when $N = 3$, we have $\Theta = \{\Delta p_n \mid n = 1, \ldots, |R|\}$, $R = \{(-1, -1), (-1, 0), \ldots, (0, 1), (1, 1)\}$ [39]. Secondly, we use 2D convolutions to align the channels of features in the U-Net encoder. Finally, we obtain four-level offset features $F_o = \{F^1_o, F^2_o, F^3_o, F^4_o\}$.

Additionally, to obtain more accurate offsets to guide reconstruction, we use a fixed offset estimator to encode the spatial correspondence across the two image domains. Due to the lack of pixel correspondence ground truth, we leverage the alignment property of deformable convolution for supervised learning to indirectly compute the offsets. After training, we fix the estimator and integrate it into the Offset-Adapter, where the generated offset features are injected into the diffusion model to provide additional spatial correspondence information.

3.3. Offset Prior-based Diffusion Model

Diffusion-based colorization models [4, 5] require reference images or scribbles, along with the target line art, as conditions, and typically employ a conditional diffusion model. Building on this, our OPDM introduces additional matching correspondence features to align target line arts with reference images, guiding the diffusion model to transfer colors effectively. As shown in Figure 2, we perform the diffusion process in the image space to preserve more details.

In the forward diffusion process, we gradually transform a ground truth image C_{gt} into a pure noise image $C_{noise} \sim \mathcal{N}(0, \mathbf{I})$ by adding Gaussian noise over T steps. The process is formulated as:

$$q(C^{1:T}_{noise}|C_{gt}) = \prod_{t=1}^{T} q(C^t_{noise}|C^{t-1}_{noise}), \tag{2}$$

$$q(C^t_{noise} \mid C^{t-1}_{noise}) = \mathcal{N}(C^t_{noise}; \sqrt{1-\beta_t}C^{t-1}_{noise}, \beta_t\mathbf{I}), \tag{3}$$

where $\mathcal{N}$ represents the Gaussian distribution, and β_t is predefined at timestep t.

Afterwards, C_{noise} and the conditions are input together into the U-Net structure. To better utilize the conditional information, we concatenate the line arts and C_{noise} as image-level inputs. Then, we employ an encoder to extract multiscale reference features. The encoder consists of four blocks, downsampling the reference condition inputs four times through the feature extractor to align the channels with the U-Net features. The reference condition inputs first pass through a 2D convolution layer and five residual blocks to generate the condition features, followed by a max-pooling operation to downsample the features at each level. This process forms the multi-scale condition features $F_r = \{F^1_r, F^2_r, F^3_r, F^4_r\}$. Similarly, we also obtain offset features F_o (see details in Section 3.2). To balance the effectiveness and efficiency of generation, the reference and offset features are injected into the ResNet block of the denoising U-Net encoder structure to fuse with the main features through an addition operation, thereby modulating the backbone features. The condition features and condition-guided operation can be defined as follows:

$$F_r = \mathcal{F}_\theta(Concat(C_{Ref1}, C_{Ref2})), \tag{4}$$

$$F^i_m = F^i_e + F^i_r + F^i_o, i \in \{1, 2, 3, 4\}, \tag{5}$$

where C_{Ref1} and C_{Ref2} represent two keyframes: the previous frame and the future frame; F^i_m denotes modulated features, F^i_e denotes features from the U-Net encoder, F^i_r represents reference features, F^i_o represents offset features, and $\mathcal{F}_\theta$ is an encoder.

Joint Diffusion-Reverse Process for Reconstruction. The reverse process of a conventional diffusion model reconstructs images from random Gaussian noise. However, in the colorization task, the diverse generated results that are often favored in standard image generation tasks may lead to unstable colorization outcomes. To address this issue, we combine the forward diffusion process with the denoising process in our model in the training stage. This improvement provides the diffusion model with a clear generation goal, effectively adding an additional control mechanism to optimize the model for accurate color reconstruction.

Specifically, we use three different conditions to guide the generation process of the diffusion model,

$$\begin{aligned} &p_\theta(C^{0:T} \mid (l, \mathcal{F}_\theta(r), g_\theta(l, r)))) = \\ &p_\theta(C^T) \prod_{t=1}^{T} p_\theta(C^{t-1} \mid C^t, (l, \mathcal{F}_\theta(r), g_\theta(l, r))), \end{aligned} \tag{6}$$

where l is target line art, r is reference images, and g_θ represents proposed Offset-Adapter. To combine the diffusion and denoising processes in the condition-guided content reconstruction model, rather than training the network solely to predict noise at each step, we also task the network with accurately generating colorized images. The final optimization goal of the OPDM training process is to minimize the image prediction error, and the training function can be formulated as follows,

$$\mathcal{L}_{recon} = \mathbb{E}_{C_{gt},l,r,\epsilon\sim\mathcal{N}(0,I)}[\ell(C_{gt}, G_\theta(\epsilon, r, l))], \quad (7)$$

where ϵ denotes noise, ℓ measures the difference between images, and G_θ represents our proposed diffusion model. In the sampling process, we adopt DDIM [36] to maintain the quality of the colorized images while significantly reducing the generation time.

3.4. Loss Function

We use Eq. 7 to optimize the diffusion model, which can reconstruct the target colorization images. The reconstruction loss consists of noise loss, content loss and perceptual loss [16]. Noise loss is MSE loss. The content loss combines L1 loss and SSIM loss [44], which is formulated as:

$$\mathcal{L}_{content} = \|\hat{y} - y\|_1 + (1 - \text{SSIM}(\hat{y}, y)), \quad (8)$$

where $\hat{y}$ is the generated target color image and y is the ground truth color image. Besides, we introduce perceptual loss to ensure that the generated results are perceptually consistent with the ground truth images. We extract multiple layer features through the pre-trained model VGG19 network [34] to compute the perceptual loss as:

$$\mathcal{L}_{perce} = \sum_{i=1}^{5} \frac{1}{N_i} \left\|\Phi_{\hat{y}}^i - \Phi_y^i\right\|_2^2. \quad (9)$$

Finally, the reconstruction loss of OPDM is:

$$\mathcal{L}_{recon} = \lambda_{noise}\mathcal{L}_{noise} + \lambda_{cont}\mathcal{L}_{cont} + \lambda_{perce}\mathcal{L}_{perce}, \quad (10)$$

where λ controls the weights of different terms. We set λ_{noise} and λ_{cont} are 1, λ_{perce} is 0.1.

4. Experiments

4.1. Experimental Settings

Datasets. We construct a diverse line art colorization dataset to train our model and validate our results using ATD-12K [35]. To ensure style consistency, we select the Japanese-style sequences from the ATD-12K, including 4,881 training sequences and 759 test sequences. We extract line arts from the color frames using Sketch-Keras [27]. To further evaluate the effectiveness of our method, we construct a new long cartoon video dataset (AniColorSet). Specifically, we collect cartoon clips from The Spy Family, From Up on Poppy Hill, Ponyo, When Marnie Was There, and Deep Inside My Heart. Although these clips originate from different animations than those in ATD-12K, their styles are visually similar. We segment the videos into short clips and extract line arts using the same method applied to ATD-12K. AniColorSet contains 14 sequences, each comprising 29 frames, which are used to evaluate the coloring performance of different models.

Implementation Details. We implement our method in PyTorch and conduct all experiments using a single NVIDIA A40 GPU. The Adam optimizer [18] is employed with an initial learning rate of 1×10^{-5}. We train OPDM for 150K iterations with a batch size of 1, taking approximately 3 days. During the training stage of OPDM, we set the number of timesteps to 500 and the sampling steps to 10. We train our offset estimator for 80 epochs with a batch size of 10 and a learning rate of 1×10^{-4}. During both training and inference, all images from the two datasets are uniformly resized to 224×224 size as input. For performance evaluation, we use Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [44], Quaternion Structural Similarity (QSSIM) [19], Multi-scale SSIM (MS-SSIM) [43], and FID [12] to assess the quality of the generated results. Notably, QSSIM is designed for evaluating the quality of color images based on quaternions, providing a more comprehensive assessment of color fidelity. For long-sequence training and testing, we further evaluate temporal consistency using Color Distribution Consistency (CDC) [17] and Warp Error (WE) [20].

4.2. Comparison

Comparison Methods. We compare our method against state-of-the-art (SOTA) reference-based colorization methods, including TCVC [38], Color2Embed [48], SketchStructure [24], DeepColorization [33], and AnimeDiffusion [4]. For a fair comparison, we train all methods on the ATD-12K dataset and evaluate them on the same test sets of ATD-12K and AniColorSet. Since SketchStructure† does not release their code, we use our implementation of their method. Specifically, we first pre-train it on its original dataset and then fine-tune it using ATD-12K. In addition, to fairly compare our method with LVCD [15] — the latest stable video diffusion-based animation colorization method — we retrain our model using the training set released by LVCD and evaluate it on our AniColorSet. As AniDoc† [29] and MangaNinja† [26] only provide inference models, we use their released model weights to colorize the line arts.

Quantitative Comparison. We compare our method with all competing approaches on the ATD-12K test set and AniColorSet. For AniColorSet, we evaluate all methods with a gap of 16 frames, where the first and last frames are pro-

Methods	ATD-12K					AniColorSet(16)				
	PSNR↑	SSIM↑	QSSIM↑	MS-SSIM↑	FID↓	PSNR↑	SSIM↑	QSSIM↑	MS-SSIM↑	FID↓
TCVC [38]	23.40	0.847	0.873	0.909	48.68	23.34	0.804	0.827	0.859	66.36
Color2Embed [48]	20.34	0.726	0.784	0.718	78.46	18.54	0.674	0.613	0.673	132.41
SketchStructure† [24]	19.30	0.681	0.717	0.660	89.41	17.64	0.624	0.676	0.608	148.44
DeepColorization [33]	30.34	0.921	0.936	0.949	35.62	27.85	0.868	0.884	0.906	48.12
AnimeDiffusion [4]	28.26	0.898	0.918	0.933	38.88	25.68	0.836	0.860	0.875	57.84
Ours	**34.63**	**0.962**	**0.967**	**0.975**	**19.63**	**31.33**	**0.914**	**0.926**	**0.935**	**28.50**

Table 1. Comparison of cartoon colorization performance across ATD-12K and AniColorSet.

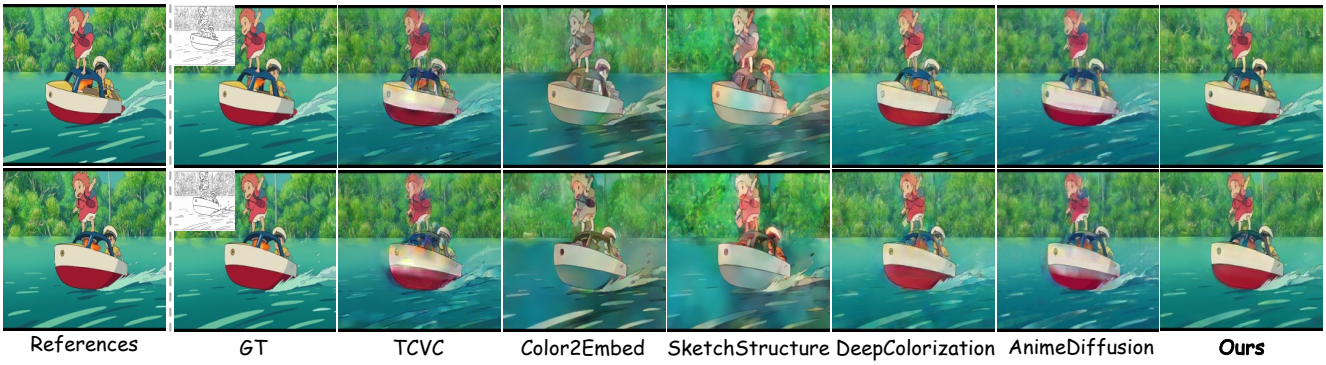

Figure 3. Colorization results of our method and comparative methods on AniColorSet. See further results in the supplementary.

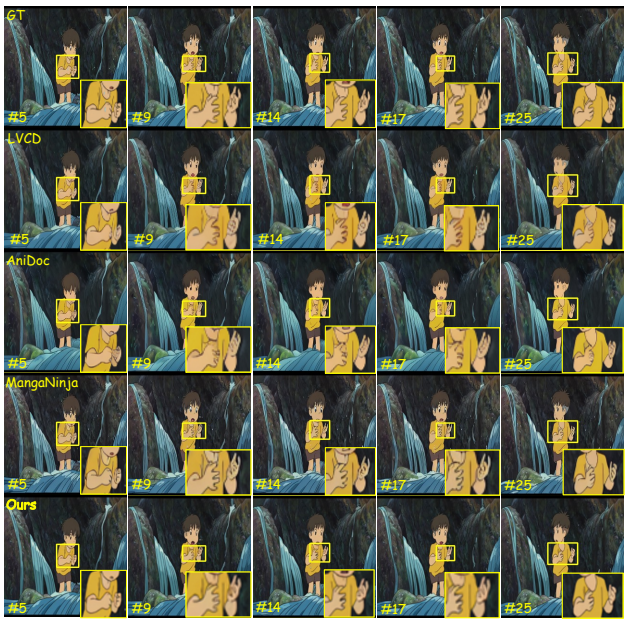

Figure 4. Comparison results with LVCD, AniDoc, MangaNinja and our method (default). Zoom in to see the details of the colorized images.

vided as reference images to guide the colorization of intermediate line art images. As shown in Table 1, our proposed method achieves state-of-the-art performance across all metrics. For a fair comparison with LVCD, AniDoc, MangaNinja, our method (1-Input) and our method (default), we train our model on the LVCD dataset with a frame gap of 16. During inference, we evaluate all four methods on 29-frame sequences. Since the input image resolutions of these methods are inconsistent, we follow LVCD's setting by resizing all frames to 256×256 and normalizing pixel values to [0.0, 1.0] for metric calculation. The quantitative results are summarized in Table 2. Additional evaluation metrics, including MS-SSIM, QSSIM, and WE, are reported in the supplementary materials to complement the main analysis and further demonstrate the effectiveness of our method. Our method achieves the best performance in terms of PSNR, SSIM, FID and CDC.

Methods	PSNR↑	SSIM↑	FID↓	CDC↓	Params(M)
LVCD [15]	29.26	0.908	44.89	0.009104	3071.84
AniDoc† [29]	24.12	0.811	91.51	0.005622	2313.05
MangaNinja† [26]	24.97	0.819	61.22	0.025350	3043.83
Ours (1-Input)	29.38	0.914	49.99	0.003949	205.00
Ours (Default)	**31.19**	**0.926**	**38.45**	**0.003184**	205.58

Table 2. Quantitative comparison of our proposed method with diffusion-based video colorization methods on AniColorSet.

Qualitative Comparison. We present qualitative comparisons between our method and all competing methods on the two datasets. Here, we demonstrate colorization results

Methods	PSNR↑	SSIM↑	QSSIM↑	MS-SSIM↑	FID↓
RDM	13.86	0.625	0.672	0.747	104.27
Ours_{v1}	33.85	0.957	0.963	0.970	25.47
Ours_{v2}	14.80	0.691	0.732	0.841	76.30
Ours_{v3}	34.08	0.959	0.964	0.971	22.41
Ours_{v4}	34.41	0.961	**0.967**	0.974	22.45
Full Model	**34.63**	**0.962**	**0.967**	**0.975**	**19.63**

Table 3. Ablation studies of each component on ATD-12K.

on AniColorSet in Figures 3 and 4.

The qualitative results on AniColorSet are shown in Figure 3, where we select 2 representative frames from the generated 14 frames (more comparisons are provided in the supplementary materials). The images generated by TCVC exhibit noticeable color inconsistencies and severe artifacts. Color2Embed and SketchStructure fail to colorize key regions (e.g. girl and the boat). DeepColorization produces results with artifacts and incorrect color assignments, for example, on the girl's bag. The outputs of AnimeDiffusion suffer from significant blurring. In contrast, our method effectively handles variations in object orientations and locations while maintaining color consistency and realism.

In Figure 4, we present a qualitative comparison of colorization results among LVCD, AniDoc, MangaNinja, and our method on our AniColorSet. More comparison results are seen in the supplementary. LVCD and AniDoc suffer from loss of fine details and slight distortion of hand line structures. MangaNinja fails to maintain accurate line preservation and realistic background generation, such as for the stream. In contrast, our method generates visually realistic results with better structural preservation and background consistency, while minimizing artifacts and chromatic aberrations.

4.3. Ablation Studies

In this section, we present the ablation studies to evaluate the effect of each component of our method. We construct the reference-based diffusion method (RDM) as a baseline which is composed of the conventional conditional diffusion model. Among them, the introduced new training strategy is denoted as TS; the Offset-Adapter is recorded as OA, and the perceptual loss is represented as PL. Therefore, different models which are named Ours_{v1}, Ours_{v2}, Ours_{v3}, Ours_{v4} are built to conduct ablation experiments. More ablation results are in the supplementary.

Training Strategy. To assess the impact of different training strategies on colorization, we conduct experiments by removing the reverse denoising process during the training stage. As in Table 3, the proposed training method significantly enhances the colorization quality.

Offset-Adapter. To evaluate the effectiveness of the Offset-Adapter, we conduct ablation experiments by adding this module to RDM and Ours_{v1}. In Table 3, the use of the Offset-Adapter to provide matching correspondence features for RDM and Ours_{v1} improves performance. To further demonstrate its effectiveness, we replace the Offset-Adapter with PWCNet (FlowDM). Specifically, we use PWCNet to generate the optical flows and input these flows into an encoder to form multi-scale optical flow features that modulate the backbone features. Due to the limitations of our experimental setup, the image resolution used for training and testing is 160×160. It is worth noting that since downsampling is required when calculating MS-SSIM, the image resolution is resized to 224 for this calculation. As indicated in Table 4, our method provides a notable improvement over PWCNet.

Methods	PSNR↑	SSIM↑	QSSIM↑	MS-SSIM↑	FID↓
Offset Adapter vs FlowNet					
FlowDM	32.30	0.939	0.947	0.959	42.32
Ours*	**34.08**	**0.956**	**0.961**	**0.972**	**32.37**
KernelSize					
K=1	34.32	0.959	0.965	0.973	21.57
K=3	34.50	0.961	**0.967**	0.974	20.45
K=5	**34.63**	**0.962**	**0.967**	**0.975**	**19.63**
Deformable-based colorization model					
DCM	33.65	0.956	0.962	0.968	37.29
Ours	**34.63**	**0.962**	**0.967**	**0.975**	**19.63**

Table 4. Ablation results of Offset-Adapter on ATD-12K. * indicates the variant trained with a different input resolution.

TimeStep	PSNR↑	SSIM↑	QSSIM↑	MS-SSIM↑	Time↓	FID↓
T=5	34.44	0.961	0.966	0.974	**0.31s**	**20.31**
T=10	**34.63**	**0.962**	**0.967**	**0.975**	0.52s	**19.63**

Table 5. The influence of different timesteps on ATD-12K.

Since the offsets are indirectly generated using deformable convolution to align with the target line art, we also compare the results with those generated directly using deformable convolution. We use deformable convolution and a U-Net structure to construct the deformable-based colorization model (DCM), concatenating the two aligned images to generate the final colorization results. As shown in Table 4, the colorization results produced using the offset features as conditions in the diffusion model outperform those of DCM.

We also explore the effect of different convolution kernel sizes for the Offset-Adapter on colorization results. As discussed in Section 3.2, different convolution kernel sizes can generate varying numbers of offsets, and these diverse offsets can provide more information for the diffusion model. Therefore, we study how the offsets generated by convolu-

tion kernels of different sizes in the Offset-Adapter affect the colorization results. As the number of convolution kernels increases, the model's computational complexity also grows. To explore the trade-off between performance and computational efficiency, we evaluate three kernel sizes: 1, 3, and 5, as shown in Table 4. The results indicate that using a kernel size of 5 yields slightly better performance than size 3, whereas size 1 leads to a noticeable degradation. Thus, increasing the number of offsets improves the model's performance to some extent.

Sampling Steps. The generation quality of the OPDM is influenced by the number of sampling steps; however, using too many sampling steps can reduce generation efficiency. Therefore, with the support of our hardware setup, we test two different sampling step configurations, as shown in Table 5. With 10 sampling steps, we obtain satisfactory colorization results, with each image taking approximately 0.5 seconds to process, which is considered acceptable.

4.4. Coloring of Hand-drawn Line Arts

In cartoon production, the cartoon sequence frames drawn by animators differ from the cartoon data obtained in the published animation. To verify the effectiveness of our method, we apply it to color hand-drawn line art from the Anita Dataset [30]. As shown in Figure 5, our method, trained with automatically extracted line art, can adapt to hand-drawn line art.

4.5. Various Types of Line Arts

Line Art Generalization. We also examine the impact of using different types of line art on our model's colorization performance. We employ Anime2Sketch [45] and the Convey Geometry and Semantics Method (CGSM) [8] to extract line art from the ATD-12K. In the inference stage, we input these two types of line art into our model to evaluate its generalization capabilities. The colorized images produced from line art extracted using Anime2Sketch, CGSM and Sketch-Keras are shown in Figure 6. Our model, trained on line art extracted with Sketch-Keras, demonstrates a certain degree of generalization to other types of line art.

Line Art Augmentation. We also investigate whether using different types of line art together to train a single model can improve performance. As shown in Figure 7, incorporating a variety of line art types during training enhances colorization performance for hand-drawn line art. For instance, our model trained with a single type of line art produces incorrect colors in areas lacking shadow lines, whereas the model trained with multiple line art types effectively addresses this issue.

5. Limitations and Conclusion

In this paper, we propose a diffusion-based method for cartoon video colorization using an offset prior, dubbed

Figure 5. Colorization results of hand-drawn line arts.

Figure 6. The coloring results of different lineart extraction methods on ATD-12K.

Figure 7. Comparative colorization results on hand-drawn data using line art augmentation and single line art.

OPDM. Specifically, we construct a multi-conditional diffusion model with a feature encoder as the baseline. An Offset-Adapter, based on the idea of deformable convolution, is designed to provide additional spatial matching information for the diffusion model. Finally, the forward diffusion and reverse denoising processes are combined during training to ensure content consistency. To better evaluate colorization performance in long videos, we introduce AniColorSet. Extensive experimental results on publicly available benchmarks and our AniColorSet demonstrate that our method outperforms competitors both quantitatively and qualitatively, while being computationally efficient.

Although our method generates better colorization results than other SOTA methods, it struggles with large-sized images due to performing the diffusion model at the image level. For exaggerated changes, our method cannot always produce perfect colorization results. In future work, we will explore novel autoencoder designs for better detail preservation, and investigate more robust motion modeling methods to handle spatial variation across frames and domains.

Acknowledgement

This work was supported by the Marsden Fund Council managed by the Royal Society of New Zealand (No. MFP-20-VUW-180) and the Faculty Strategic Research Grant (No. 412684) from Victoria University of Wellington.

References

[1] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 1

[2] F.L. Bookstein. Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 11(6):567–585, 1989. 2

[3] Yu Cao, Hao Tian, and P. Y. Mok. Attention-aware anime line drawing colorization. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1637–1642, 2023. 2

[4] Yu Cao, Xiangqiao Meng, P. Y. Mok, Tong-Yee Lee, Xueting Liu, and Ping Li. AnimeDiffusion: Anime diffusion colorization. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 30(10):6956–6969, 2024. 2, 4, 5, 6

[5] Hernan Carrillo, Michaël Clément, Aurélie Bugeau, and Edgar Simo-Serra. Diffusart: Enhancing line art colorization with conditional diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3486–3490, 2023. 2, 4

[6] Evan Casey, Víctor Pérez, and Zhuoru Li. The animation transformer: Visual correspondence via segment matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11323–11332, 2021. 2

[7] Duygu Ceylan, Chun-Hao P. Huang, and Niloy J. Mitra. Pix2Video: Video editing using image diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 23206–23217, 2023. 2

[8] Caroline Chan, Frédo Durand, and Phillip Isola. Learning to generate line drawings that convey geometry and semantics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7915–7925, 2022. 8

[9] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Understanding deformable alignment in video super-resolution. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*, pages 973–981, 2021. 3

[10] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 764–773, 2017. 3

[11] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2014. 1

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2017. 5

[13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 1

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6840–6851. Curran Associates, Inc., 2020. 1

[15] Zhitong Huang, Mohan Zhang, and Jing Liao. LVCD: reference-based lineart video colorization with diffusion models. *ACM Transactions on Graphics (TOG)*, 43(6):1–11, 2024. 3, 5, 6

[16] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision (ECCV)*, pages 694–711. Springer, 2016. 5

[17] Xiaoyang Kang, Xianhui Lin, Kai Zhang, Zheng Hui, Wangmeng Xiang, Jun-Yan He, Xiaoming Li, Peiran Ren, Xuansong Xie, Radu Timofte, Yixin Yang, Jinshan Pan, Zhongzheng Peng, Qiyan Zhang, Jiangxin Dong, Jinhui Tang, Jinjing Li, Chichen Lin, Qipei Li, Qirong Liang, Ruipeng Gang, Xiaofeng Liu, Shuang Feng, Shuai Liu, Hao Wang, Chaoyu Feng, Furui Bai, Yuqian Zhang, Guangqi Shao, Xiaotao Wang, Lei Lei, Siqi Chen, Yu Zhang, Hanning Xu, Zheyuan Liu, Zhao Zhang, Yan Luo, and Zhichao Zuo. NTIRE 2023 video colorization challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1570–1581, 2023. 5

[18] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[19] Amir Kolaman and Orly Yadid-Pecht. Quaternion structural similarity: A new quality index for color images. *IEEE Transactions on Image Processing (TIP)*, 21(4):1526–1536, 2012. 5

[20] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 170–185, 2018. 5

[21] Junsoo Lee, Eungyeup Kim, Yunsung Lee, Dongjun Kim, Jaehyuk Chang, and Jaegul Choo. Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5801–5810, 2020. 2

[22] Jinjing Li, Qirong Liang, Qipei Li, Ruipeng Gang, Ji Fang, Chichen Lin, Shuang Feng, and Xiaofeng Liu. RTTLC: Video colorization with restored transformer and test-time local converter. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1722–1730, 2023. 2

[23] Zekun Li, Zhengyang Geng, Zhao Kang, Wenyu Chen, and Yibo Yang. Eliminating gradient conflict in reference-based line-art colorization. In *European Conference on Computer Vision (ECCV)*, pages 579–596. Springer, 2022. 2

[24] Xueting Liu, Wenliang Wu, Chengze Li, Yifan Li, and Huisi Wu. Reference-guided structure-aware deep sketch colorization for cartoons. *Computational Visual Media (CVM)*, 8: 135–148, 2022. 2, 5, 6

[25] Yihao Liu, Hengyuan Zhao, Kelvin CK Chan, Xintao Wang, Chen Change Loy, Yu Qiao, and Chao Dong. Temporally consistent video colorization with deep feature propagation and self-regularization learning. *Computational Visual Media (CVM)*, 10(2):375–395, 2024. 2

[26] Zhiheng Liu, Ka Leong Cheng, Xi Chen, Jie Xiao, Hao Ouyang, Kai Zhu, Yu Liu, Yujun Shen, Qifeng Chen, and Ping Luo. MangaNinja: Line art colorization with precise reference following. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5666–5677, 2025. 3, 5, 6

[27] lllyasviel. SketchKeras. https://github.com/lllyasviel/sketchKeras, 2018. 5

[28] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11461–11471, 2022. 2

[29] Yihao Meng, Hao Ouyang, Hanlin Wang, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Zhiheng Liu, Yujun Shen, and Huamin Qu. AniDoc: Animation creation made easier. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18187–18197, 2025. 3, 5, 6

[30] Zhenglin Pan and Yu Zhu. Anita Dataset. https://zhenglinpan.github.io/AnitaDataset_homepage, 2024. 8

[31] Mengwei Ren, Mauricio Delbracio, Hossein Talebi, Guido Gerig, and Peyman Milanfar. Multiscale structure guided diffusion for image deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10721–10733, 2023. 2

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 2

[33] Min Shi, Jia-Qi Zhang, Shu-Yu Chen, Lin Gao, Yu-Kun Lai, and Fang-Lue Zhang. Reference-based deep line art video colorization. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 29(6):2965–2979, 2023. 1, 2, 5, 6

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv: 1409.1556*, 2014. 5

[35] Li Siyao, Shiyu Zhao, Weijiang Yu, Wenxiu Sun, Dimitris Metaxas, Chen Change Loy, and Ziwei Liu. Deep animation video interpolation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6587–6595, 2021. 2, 5

[36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2022. 5

[37] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8934–8943, 2018. 1

[38] Harrish Thasarathan, Kamyar Nazeri, and Mehran Ebrahimi. Automatic temporally coherent video colorization. In *2019 16th Conference on Computer and Robot Vision (CRV)*, pages 189–194, 2019. 1, 2, 5, 6

[39] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. TDAN: Temporally-deformable alignment network for video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3360–3369, 2020. 3, 4

[40] Ning Wang, Muyao Niu, Zhi Dou, Zhihui Wang, Zhiyong Wang, Zhaoyan Ming, Bin Liu, and Haojie Li. Coloring anime line art videos with transformation region enhancement network. *Pattern Recognition (PR)*, 141:109562, 2023. 1, 2

[41] Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. EDVR: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 0–0, 2019. 3

[42] Yufei Wang, Wenhan Yang, Xinyuan Chen, Yaohui Wang, Lanqing Guo, Lap-Pui Chau, Ziwei Liu, Yu Qiao, Alex C. Kot, and Bihan Wen. SinSR: Diffusion-based image super-resolution in a single step. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25796–25805, 2024. 2

[43] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, pages 1398–1402. Ieee, 2003. 5

[44] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*, 13(4):600–612, 2004. 5

[45] Xiao Yang Yiheng Zhu Xiaohui Shen Xiaoyu Xiang, Ding Liu. Anime2Sketch: A sketch extractor for anime arts with deep networks. https://github.com/Mukosame/Anime2Sketch, 2021. 8

[46] Seungjoo Yoo, Hyojin Bahng, Sunghyo Chung, Junsoo Lee, Jaehyuk Chang, and Jaegul Choo. Coloring with limited data: Few-shot colorization via memory augmented networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11283–11292, 2019. 2

[47] Qian Zhang, Bo Wang, Wei Wen, Hai Li, and Junhui Liu. Line art correlation matching feature transfer network for automatic animation colorization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3872–3881, 2021. 1, 2

[48] Hengyuan Zhao, Wenhao Wu, Yihao Liu, and Dongliang He. Color2Embed: Fast exemplar-based image colorization

using color embeddings. *arXiv preprint arXiv:2106.08017*, 2021. 5, 6

[49] Yuzhi Zhao, Lai-Man Po, Kangcheng Liu, Xuehui Wang, Wing-Yin Yu, Pengfei Xian, Yujia Zhang, and Mengyang Liu. SVCNet: Scribble-based video colorization network with temporal aggregation. *IEEE Transactions on Image Processing (TIP)*, 32:4443–4458, 2023. 2