PalGAN: Image Colorization with Palette Generative Adversarial Networks

Yi Wang[1], Menghan Xia[2], Lu Qi[3], Jing Shao[4], and Yu Qiao[1]✉

[1]Shanghai AI Laboratory [2]Tencent AI Lab [3]UC Merced [4]SenseTime Research
{wangyi,qiaoyu}@pjlab.org.cn menghanxyz@gmail.com
shaojing@senseauto.com

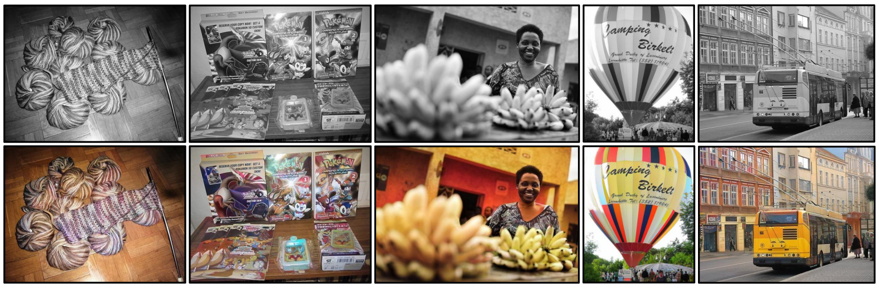


Fig. 1: Our colorization results. 1_{st} row: inputs, and 2_{nd} row: our predictions.

Abstract. Multimodal ambiguity and color bleeding remain challenging in colorization. To tackle these problems, we propose a new GAN-based colorization approach PalGAN, integrated with palette estimation and chromatic attention. To circumvent the multimodality issue, we present a new colorization formulation that estimates a probabilistic palette from the input gray image first, then conducts color assignment conditioned on the palette through a generative model. Further, we handle color bleeding with chromatic attention. It studies color affinities by considering both semantic and intensity correlation. In extensive experiments, PalGAN outperforms state-of-the-arts in quantitative evaluation and visual comparison, delivering notable diverse, contrastive, and edge-preserving appearances. With the palette design, our method enables color transfer between images even with irrelevant contexts.

Keywords: Image Colorization, Generative Adversarial Networks, Attention, Color Transfer

1 Introduction

Colorization means to predict the missing chrome information from the given gray image. It is an interesting and practical task in computer vision, widely used in legacy footage processing [27], color transfer [1,39], and other visual editing applications [3,52]. It is also exploited as a proxy task for self-supervised learning [25], since predicting perceptually natural colors from the given grayscale image

✉Corresponding author

heavily relies on scene understanding. However, even the ground-truth color is available for supervision, it is still very challenging to predict pixel colors from gray images, due to the ill-posed nature that one input grayscale could correspond to multiple possible color variants.

Most current methods [54,56,26,12,23,38,49,17,3] formulate colorization as a pixel-level regression task, suffering from multimodal representation more or less. With the large-scale training data and end-to-end learning models, they can learn the color distribution prior conveniently, *e.g.* vegetation greenish tones, human skin colors, *etc.*. Anyhow, when it comes to objects with inherently color ambiguity (*e.g.* human clothes, cars, and other man-made stuff), these approaches tend to predict the brownish average colors. To tackle such multi-modality, researches [54,56,24] proposed to formulate the color prediction as pixel-level color classification, which allows multiple colors to be assigned to each pixel based on posterior probability. Unfortunately, these suffer from regional color inconsistency due to the independent pixel-wise sampling mechanism. In this regard, means of utilizing the sequential modeling [12,23] can only partially help the sampling issue, because the unidirectional sequential dependence of 2D flattened pixel primitives causes error accumulation and hinders the learning efficiency.

Apart from the multimodal issue, color bleeding is another common issue in colorization due to inaccurate identification of semantic boundaries. To suppress such visual artifacts, most works [54,56,26,38,49,17,3] resort to Generative adversarial networks (GAN) to encourage the generated chrome distribution to be indistinguishable from that of the real-life color images. Currently, no special algorithms or modules for deep models have been proposed to enhance the performance of this aspect, which matters the visual pleasantness considerably.

To avoid modeling the color multimodality pixel-wisely, we propose a new colorization framework PalGAN that predicts the pixel colors in a coarse-to-fine paradigm. The key idea is to first predict the global palette probability (*e.g.* palette histogram) from the grayscale. It does not collapse into a single specific colorization solution but represents a certain color distribution of the potential color variants. Then, the uncertainty about the per-pixel color assignment is modeled with a generative model in the GAN framework, conditioned on the grayscale and palette histogram. Therefore, multiple colorization results could be achieved by changing the palette histogram input.

To guarantee the color assignment with semantic correctness and regional consistency, we study color affinities by a proposed chromatic attention module. It explicitly aligns color affinity with both semantics and low-level characteristics. In structure, chromatic attention includes global interaction and local delineation. The former enables global context utilization for color inference by using semantic features in the attention mechanism. The latter preserves regional details by mapping the gray input to color through local affine transformation. The transformation is explicitly parameterized by the correlation between gray input and color feature. Experiments illustrate the effectiveness of our method. It achieves impressive visual results (Fig. 1) and quantitative superiority over state-of-the-art approaches over ImageNet [9] and COCO-

Stuff [5]. Our method also works well with the user-specified palette histogram from a reference image, which could even have no content correlation with the input grayscale. So, by nature, our method supports diverse coloring results with certain controllability. Our code and pretrained models are available at https://github.com/shepnerd/PalGAN.

Generally, our contributions are three-fold: i) We propose a new colorization framework PalGAN that decomposes colorization to palette estimation and pixel-wise assignment. It circumvents the challenges of color ambiguity and regional homogeneity effectively, and supports diverse and controllable colorization by nature. ii) We explore the less-touched color affinities and propose an effective module named chromatic attention. It considers both semantic and local detail correspondence, applying such correlations to color generation. It alleviates notable color bleeding effects. iii) Our method surpasses state-of-the-arts in perceptual quality (FID [16] and LPIPS [55]) notably. It is known that there exists a trade-off between perceptual and fidelity results in multiple low-level tasks. We argue perceptual effects matter more than fidelity as colorization aims to produce realistic colorized results rather than restore identical pixel-wise colors as the ground truth. Regardless, our method can achieve best both fidelity (PSNR and SSIM) and perceptual performance with proper tuning.

2 Related Work

2.1 Colorization

User Guided Colorization Some of early works [7,8,18,29,39,47,21,6,34] in colorization turn to a reference image for transferring its color statistics to the given gray one. With the prevalence of deep learning, such color transfer is characterized in neural feature space for introducing semantic consistency [15]. These works perform decently when the reference and input share similar semantics. Its applications are limited by the reference retrieval quality, especially when handling complicated scenes.

Besides of reference images, several systems require users to give sufficient local color hints (usually in scribble form) before colorizing inputs [27,37,52,21]. Then approaches propagate the given colors based on their local affinities. Besides, some attempts are made [3] to explore other modalities like languages to instruct what colors are used and how they are distributed.

Learning-based Colorization This line of work [54,56,10,17,24,19] gives colorful images only from the gray inputs, learning a pixel-to-pixel mapping. Large-scale datasets are exploited in a self-supervised fashion, converting colorful pictures to gray ones for pair-wise training. Iizuka *et. al.* [17] utilize image-level labels for associating predicted color with global semantics, using a global-and-local convolutional neural network. Larsson *et. al.* [24] and Zhang *et. al.* [54] introduce pixel-level color distribution matching by classification, alleviating color unbalance and multi-modal outputs. Besides, extra input hints are integrated into learning systems by simulation in [56], providing automatic and semi-automatic ways to colorize images. Recently, transformer architectures are explored for this task considering their expressiveness on non-local modeling [23].

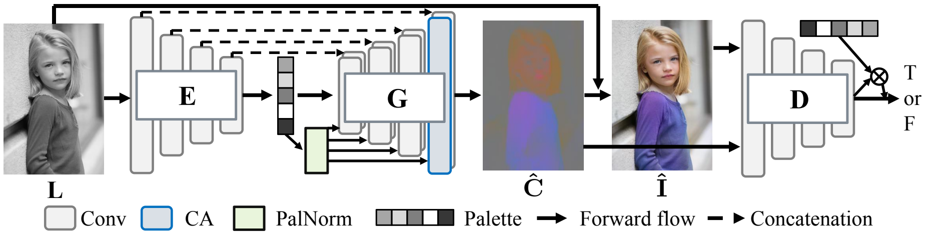


Fig. 2: Our colorization system framework.

Some work explicitly exploits additional priors from pretrained models for colorization. Su *et. al.* [38] study leveraging instance-level annotations (e.g., instance bounding boxes and classes) by using an off-the-shelf detector. It will make the colorization model focuses on color rendering without the need of recognizing high-level semantics. In addition to the mentioned pretrained discriminative models, pretrained generative ones are also exploitable in improving colorization performance in diversity. Wu *et. al.* [49] explore to incorporate generative color prior from a pretrained BigGAN [4] to help a deep model produce colored results with diversities. They design an extra encoder to project the given gray image into latent code, then estimate colorful images from BigGAN. With such primary predictions, they further refine the color results by the intermediate features in BigGAN. Afifi *et. al.* [1] propose employing a pretrained StyleGAN [20] for image recoloring, and color is controlled by histogram features.

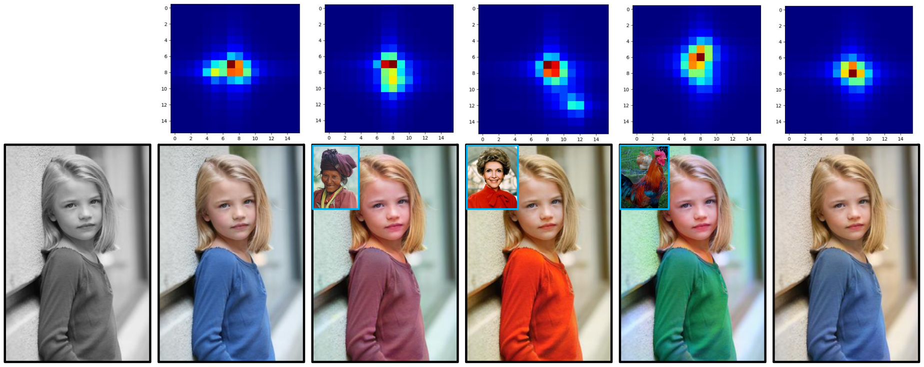
2.2 GAN-based Image-to-image Translation

Image-to-image translation aims to learn the transformation between the input and output image. Colorization can be formulated to this task and handled by Generative Adversarial Networks [11] (GAN) based approaches [19,41,35,30,44]. They employ an adversarial loss that learns to discriminate between real and generated images, and then minimize this loss by updating the generator to make the produced results look realistic [57,28,31,50,36,51,45,46,42].

3 Method

PalGAN aims to colorize grayscale images. It formulates colorization as a palette prediction and assignment problem. Compared with directly learning the pixel-to-pixel mapping from gray to color as adopted by most learning-based methods, this disentanglement fashion not only brings empirical colorization improvements (Section 4), but also enables us to manipulate global color distributions by adjusting or regularizing palettes.

For PalGAN, its input is a grayscale image (*i.e.* the luminance channel of color images) $\mathbf{L} \in \mathcal{R}^{h\times w\times 1}$, and the output is the estimated chromatic map $\hat{\mathbf{C}} \in \mathcal{R}^{h\times w\times 2}$ that will be used as the complementary *ab* channels together with $\mathbf{L}$ in CIE *Lab* color space. PalGAN consists of palette generator $\mathcal{T}_{\mathbf{E}}$, palette assignment generator $\mathcal{T}_{\mathbf{G}}$, and a color discriminator $\mathbf{D}$. In inference, only $\mathcal{T}_{\mathbf{E}}$ and $\mathcal{T}_{\mathbf{G}}$ are employed. The whole framework is given in Fig. 2.



(a) (b) (c) (d) (e) (f)

Fig. 3: Visualizations of palettes ($1_{\rm st}$ row, shown in jet colormap) and how they work on colorization ($2_{\rm nd}$ row). (a) Input, (b) the ground truth, (c)-(e) reference-based colorization, (f) automatic colorization.

3.1 Palette Generator

$\mathcal{T}_{\mathbf{E}}$ estimates the global palette probabilities from the given gray image as $\hat{\mathbf{h}} = \mathcal{T}_{\mathbf{E}}(\mathbf{L})$. We employ a 2D chromatic histogram $\hat{\mathbf{h}} \in \mathcal{R}^{N_a \times N_b \times 1}$ to represent palette probabilities (N_a and N_b denotes bin numbers of a and b axes respectively), modeling the chromatic information statistics instead of learning a deterministic one. $\mathcal{T}_{\hat{P}}$ is an encoder network with several convolutional layers and a few multiple-layer perceptions (MLP), ended with a sigmoid function. The former is to extract features and the latter is to transform spatial features to a histogram (in vector form). With the explicit representation of the color palette in histogram form, we find it not only makes global color distribution more predictable, but also manipulative by introducing proper regularizations.

The user-guided colorization [56,6,34] has demonstrated the effectiveness of utilizing the color histogram of a reference image for colorizing images. Compared to the existing practice [56,6,34], we make one step further, *i.e.* synthesizing a palette histogram conditioned on the input grayscale instead of taking that from a user-specified reference image. This design brings two non-trivial advantages. First, it makes our method to be a self-contained fully automatic colorization system, instead of depending on any outside guidance (i.e. a reference image) to work. Second, in general cases, the palette histogram estimated each specific grayscale may offer more accurate and instructive information for the colorization process, than that from a reference image selected in the wild. We empirically demonstrate this in Section 4.4. In Fig. 3, we visualize the predicted palette histogram (f), in comparison with the ground-truth (b) and those of reference images (c e).

3.2 Palette Assignment Generator

$\mathcal{T}_{\mathbf{G}}$ conducts color assignment task via conditional image generation. It produces the corresponding ab from the gray image conditioned on palette histogram $\hat{\mathbf{h}}$

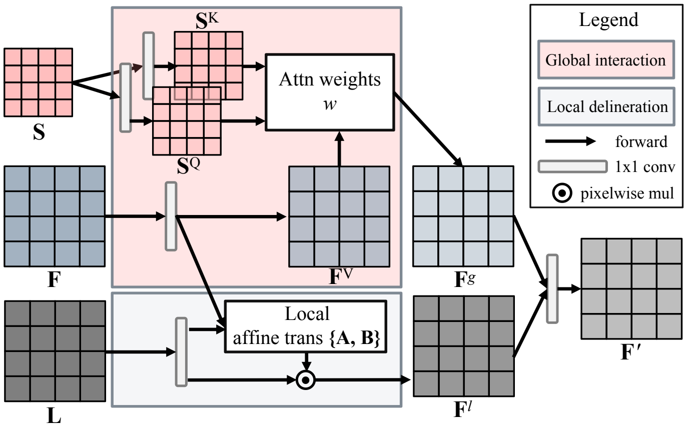


Fig. 4: The illustration of chromatic attention.

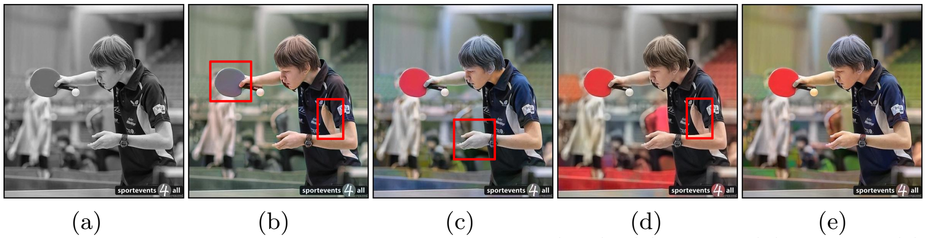


Fig. 5: Ablation studies of chromatic attention (CA). (a) input, (b) wo CA, (c) w Global, (d) w Local, (e) full CA. Please zoom in.

and extra latent code z (sampled from a normal distribution), as $\hat{\mathbf{C}} = \mathcal{T}_{\mathbf{G}}(\mathbf{L}|\hat{\mathbf{h}}, z)$. It is a convolutional generator is composed of common residual blocks used in image translation [14,19], together with our customized Palette Normalization (PN) layer and Chromatic Attention (CA) module. The palette normalization is designed to promote the conformity of the generated chromatic channels to the palette guidance $\hat{\mathbf{h}}$, which is used along with each Batch Normalization layers. Specifically, the PN layer normalizes its input feature first and then performs an affine transformation parameterized by $g(\hat{\mathbf{h}})$ (where $g(\cdot)$ is a fully-connected layer). Besides, we propose a chromatic attention module (Fig. 4) to explicitly align color affinity to their corresponding semantic and low-level characteristics, which mitigates potential color bleeding or semantic misunderstanding effectively. We discuss the designs below in detail, along with a visualization of the effects of its components shown in Fig. 5.

Chromatic Attention The proposed Chromatic Attention (CA) module incorporates both semantic and low-level affinities into constructing color relations. These two are realized by *global interaction* and *local delineation* submodules (Fig. 4). Specifically, inputs to CA are a high-resolution feature map $\mathbf{F}$ (of the size $\mathcal{R}^{128\times128\times64}$ from $\mathcal{T}_{\mathbf{G}}$), high-level feature map $\mathbf{S}$, and resized gray input $\mathbf{L}$. It outputs two feature maps $\mathbf{F}^g$ and $\mathbf{F}^l$ from global interaction and local delineation respectively, and fuses them into a feature map residual, adding back to the input feature map, as:

$$\mathrm{CA}(\mathbf{F}, \mathbf{S}, \mathbf{L}) = \mathbf{F} + \mathbf{F}' = \mathbf{F} + f(\mathbf{F}^g \oplus \mathbf{F}^l) = \mathbf{F} + f(\mathrm{CA}_g(\mathbf{F}|\mathbf{S}) \oplus \mathrm{CA}_l(\mathbf{L}|\mathbf{F})), \quad (1)$$

where $f(\cdot)$ is a nonlinear fusion operation formed by two consecutive convolutional layers, and $\oplus$ is channel-wise concatenation operation. $\mathrm{CA}_g(\cdot)$ and $\mathrm{CA}_l(\cdot)$ denote *global interaction* and *local delineation*, respectively. In this paper, we use $\mathbf{F}, \mathbf{F}' \in \mathcal{R}^{128\times128\times64}$.

Global Interaction We reconstruct every regional feature point from the input feature map using a weighted sum of other ones, and such local weight is computed according to their semantic similarity. Formally, it is written as $\mathbf{F}^g_p = \sum_{q\in\mathbf{F}} w_{pq}\mathbf{F}^{\mathrm{V}}_q$, where p and q denote a patch centering at pixel location p and q within $\mathbf{F}$, respectively. And w_{pq} is calculated from the region-wise interaction in the learned high-level feature maps from the input gray images. The region-wise feature interaction is measured by the cosine similarity between the normalized regional features, as:

$$w_{pq} = \frac{\exp(w'_{pq})}{\sum_{k\in\mathbf{S}} \exp(w'_{pk})} \quad \text{where} \quad w'_{pq} = \frac{\mathbf{S}^{\mathrm{K}}_p \cdot \mathbf{S}^{\mathrm{Q}}_q}{|\mathbf{S}^{\mathrm{K}}_p||\mathbf{S}^{\mathrm{Q}}_q|}, \quad (2)$$

where $\mathbf{S}$ denote high-level feature map, extracted from intermediate representation of the encoder $\mathcal{T}_{\hat{P}}$. $\mathbf{S}^{\mathrm{K}}$ and $\mathbf{S}^{\mathrm{Q}}$ denote two translated feature maps from $\mathbf{S}$ using convolution.

Local Delineation Though color changes in texture and edges are delicate, overlooking these subtle variances leads to notable visual degradation. To preserve these details, we design a local delineation module to complement global interaction. We adopt the assumption that local color affinity is linearly correlated with its corresponding intensity [58,40]. We propose to learn such local relationship in the guided filter manner [13,48], which preserves edges from the guidance well. Our given local preserving module computes a learnable local affine transformation $\{\mathbf{A} \in \mathcal{R}^{128\times128\times64}, \mathbf{B} \in \mathcal{R}^{128\times128\times64}\}$ to map the gray image $\mathbf{L} \in \mathcal{R}^{256\times256\times1}$ to its corresponding *ab* feature map, as:

$$\mathbf{F}^l = \mathbf{A} \odot \mathbf{L} \downarrow + \mathbf{B}, \quad (3)$$

where $\odot$ is the element-wise multiplication operator and $\downarrow$ is downsampling one to ensure the spatial size of $\mathbf{L}$ is the same as $\mathbf{F}^l$. $\{\mathbf{A}, \mathbf{B}\}$ are parameterized by the a learnable local correlation between $\mathbf{L}$ and $\mathbf{F}$, as:

$$\mathbf{A} = \Psi(\frac{\mathrm{cov}(\mathbf{F}, \mathbf{L})}{\mathrm{var}(\mathbf{L}) + \epsilon}), \ \mathbf{B} = \overline{\mathbf{F}} - \mathbf{A} \odot \overline{\mathbf{L}} \quad (4)$$

where Ψ is a learnable transformation parameterized by a small convolutional net, $\mathrm{cov}(\cdot,\cdot)$ computes the local covariance between two feature maps (within a fixed window size) while var() computes the local variance of the given feature map. $\overline{\mathbf{F}}$ and $\overline{\mathbf{L}}$ denote the smoothed versions of $\mathbf{F}$ and $\mathbf{L}$ by a mean filter, respectively. ϵ is a small positive number for computational stability.

Palette Optimization To further ensure the proposed palette assignment generator is responsive to the given palette, we minimize the discrepancy between the palette extracted from the predicted chromatic channels and that from the corresponding ground truth. However, common histograms from images are non-differentiable due to the hard thresholds. Follow the practice of [1], we regard the palette histogram as a joint distribution over a and b, represented by a weighted sum of kernels. Formally, the color histogram is written as:

$$\mathbf{h}(a,b) = \frac{1}{Z}\sum_{x} k(\mathbf{C}_a(x), \mathbf{C}_b(x), a, b), \tag{5}$$

where $\mathbf{C}_a(x)$ and $\mathbf{C}_b(x)$ denote the values of pixel x in a and b channels, respectively. k is the used kernel function to measure the difference between $(\mathbf{C}_a(x), \mathbf{C}_b(x))$ and a given (a, b), Z is a normalization factor. In this paper, we adopt inverse-quadratic kernel [1], which is:

$$k(\mathbf{C}_a(x), \mathbf{C}_b(x), a, b) = \prod_{i \in \{a,b\}} (1 + (\frac{|\mathbf{C}_i(x) - i|}{\sigma})^2)^{-1}, \tag{6}$$

where σ controls the smoothness of adjacent bins. We find $\sigma = 0.1$ works best. *Regularization* To diversify the predicted colors, we introduce palette regularization, combating against the dull colors brought by imbalanced color distribution. On one hand, we employ ab histogram in probabilistic palette form to measure the color distribution in the predicted color map and ground truth. Minimizing their discrepancies explicitly considers different color ratios, avoiding converging to a few dominant ones. On the other hand, we diversify the produced colors by increasing the possibility of rare colors (statistically in training samples). We exploit the entropy of the probabilistic palette to control such diversity. Formally, the entropy of $\hat{\mathbf{h}}$ is $E(\hat{\mathbf{h}}) = -\sum_{i=1}^{|\hat{\mathbf{h}}|} \hat{\mathbf{h}}_i \log \hat{\mathbf{h}}_i$. To improve the color diversity in $\hat{\mathbf{h}}$, we can maximize $E(\hat{\mathbf{h}})$.

3.3 Color Discriminator

We give a color discriminator utilizing the palette, improving the result from the adversarial training. We incorporate the palette into the discriminator in a condition projection manner [33]. We employ convolutional discriminator $\mathbf{D}$, converting the input (the concatenation between the ab image and its converted RGB one) into a 1D feature embedding $\mathbf{g} \in \mathbb{R}^{\mathbf{256}\times\mathbf{1}}$. Then such feature is fused with the palette by the inner product. The likelihood of the realness of the input is given as:

$$p(\mathbf{C} \oplus \mathbf{I}) = (\mathbf{W}\mathbf{g})^{\mathrm{T}}\mathbf{h}, \tag{7}$$

where $\mathbf{W} \in \mathbb{R}^{n^2\times 256}$ is a learnable linear projection, and $\mathbf{I} \in \mathbb{R}^{h\times w\times 3}$ is the converted rgb version of $\mathbf{C}$ and $\mathbf{L}$.

3.4 Learning Objective

Palette estimation and assignment are trained with different optimization targets. For palette estimation, it is learned concerning palette reconstruction and

Table 1: Quantitative results on the validation sets from different methods.

Method	ImageNet (ctest10k)				ImageNet (val50k)				COCO-Stuff			
	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
CIColor [54]	22.30	0.902	0.221	12.20	22.26	0.902	0.221	9.39	21.84	0.895	0.234	22.32
UGColor [56]	24.26	0.918	0.174	7.49	24.26	0.919	0.173	4.60	24.34	0.924	0.165	14.74
Lei *et. al.* [26]	24.52	0.917	0.202	12.60	24.03	0.918	0.189	6.35	24.59	0.922	0.191	23.10
Deoldify [2]	23.54	0.914	0.187	5.78	22.97	0.911	0.185	3.87	23.98	0.939	0.161	12.75
ColTrans [23]	21.81	0.892	0.218	6.37	22.12	0.894	0.216	3.81	22.11	0.898	0.210	11.65
Ours^1	24.19	0.917	**0.161**	**4.60**	24.25	0.917	**0.161**	**2.78**	24.56	0.924	**0.148**	**7.70**
Ours^2	**24.66**	**0.920**	0.170	5.24	**24.54**	**0.920**	0.168	3.62	**24.72**	**0.944**	0.156	8.93
InstColor* [38]	23.03	0.909	0.191	7.35	23.06	0.910	0.190	4.94	22.35	0.838	0.238	12.24
GPColor* [49]	21.66	0.871	0.230	5.46	21.81	0.880	0.230	3.62	N/A	N/A	N/A	N/A
Ours*	27.75	0.932	0.110	4.20	27.53	0.913	0.118	2.42	28.28	0.936	0.105	7.21

regularization as:

$$\mathcal{L}_{\mathbf{E}} = \underbrace{\lambda_{\text{rec1}}|\mathbf{h}-\hat{\mathbf{h}}|_1}_{\text{reconstruction}} - \underbrace{\lambda_{\text{rg}}E(\hat{\mathbf{h}})}_{\text{regularization}}\ , \tag{8}$$

where λ_{rec1} and λ_{rg} balance the influences of different terms, set to 5.0 and 1.0, respectively.

The optimization target for palette assignment is formed by pixel-level regression, palette reconstruction, and adversarial training, as:

$$\mathcal{L}_{\mathbf{G}} = \underbrace{\lambda_{\text{reg}}|\mathbf{C}-\hat{\mathbf{C}}|_1}_{\text{regression}} + \underbrace{\lambda_{\text{rec2}}|\mathbf{h}-\tilde{\mathbf{h}}|_1}_{\text{reconstruction}} + \underbrace{\lambda_{\text{adv}}\mathcal{L}_{\text{adv}}}_{\text{adversarial}}, \tag{9}$$

where $\tilde{\mathbf{h}}$ are extracted from $\hat{\mathbf{C}}$ using Eqn. 5. λ_{reg}, λ_{rec2}, and λ_{adv} are set to 5.0, 1.0, 1.0, respectively.

For the used adversarial loss, we employ hinge loss version. Its training target of generator is

$$\mathcal{L}_{adv} = -\mathrm{E}_{\mathbf{L}\sim\mathbb{P}_{\mathbf{L}}}\mathbf{D}(\hat{\mathbf{C}}\oplus\hat{\mathbf{I}}), \tag{10}$$

where $\hat{\mathbf{C}} = \mathcal{T}_{\mathbf{G}}(\mathbf{L}|\mathcal{T}_{\mathbf{E}}(\mathbf{L}))$, $\hat{\mathbf{I}}$ is a converted *rgb* version from $\hat{\mathbf{C}}$ and $\mathbf{L}$, and $\mathbb{P}_{\mathbf{L}}$ denotes the gray-scale image distribution. The optimization goal for the discriminator is

$$\mathcal{L}^{\mathbf{D}}_{adv} = \mathrm{E}_{\mathbf{I}\sim\mathbb{P}_{\mathbf{I}}}[\max(0, 1-\mathbf{D}(\mathbf{C}\oplus\mathbf{I}))] + \mathrm{E}_{\mathbf{L}\sim\mathbb{P}_{\mathbf{L}}}[\max(0, 1+\mathbf{D}(\hat{\mathbf{C}}\oplus\hat{\mathbf{I}})]. \tag{11}$$

where $\mathbb{P}_{\mathbf{I}}$ denotes the *rgb* image distribution, and $\mathbf{C}$ is converted from $\mathbf{I}$.

Training We jointly train palette generator $\mathcal{T}_{\mathbf{E}}$ and palette assignment generator $\mathcal{T}_{\mathbf{G}}$ in an progressive fashion. Specifically, for the inputs $\{\mathbf{L}_i\}$ to $\mathcal{T}_{\mathbf{E}}$, the corresponding inputs to $\mathcal{T}_{\mathbf{G}}$ are $\{\mathbb{1}(p_{\mathbf{h}} > 0.8)\mathbf{h}_i + (1-\mathbb{1}(p_{\mathbf{h}} > 0.8))\hat{\mathbf{h}}_i\}$, where $\mathbb{1}$ is an indicator function of value 1 if its condition holds true, and 0, otherwise. $p_{\mathbf{h}}$ is sampled from a uniform distribution $\mathcal{U}[\tau, 1]$. We start training with $\tau = 1$, then linearly decrease it to 0 when approaching the end of learning.

4 Experiments

We evaluate our method along with existing representative works on ImageNet [9] and COCO-Stuff [5]. On ImageNet we take two evaluation protocols. One is to evaluate all methods on a selective subset *ctest10k* (with 10K pictures) of its validate data (with 50K pictures) following the protocols in [24]. Another is to run on the full validation set, same as in [49]. For COCO-Stuff, we test all methods on its 5K validating images.

4.1 Implementation

We employ spectral normalization [32] on the whole model and a two time-scale update rule in training (lr for the generator and discriminator are $1e-4$ and $4e-4$, respectively) to stabilize learning. Adam [22] optimizer with $\beta_1 = 0$ and $\beta_2 = 0.9$ is used. For the applied batch normalization, we take the sync version. We train our method on the training set of ImageNet with 40 epochs with 8 TiTAN 2080ti using batch size 64. Images in training are randomly cropped in a fixed size (256 × 256) from the resized ones with aspect ratio unchanged. In testing, we resize images into 256 × 256 ones and do evaluations.

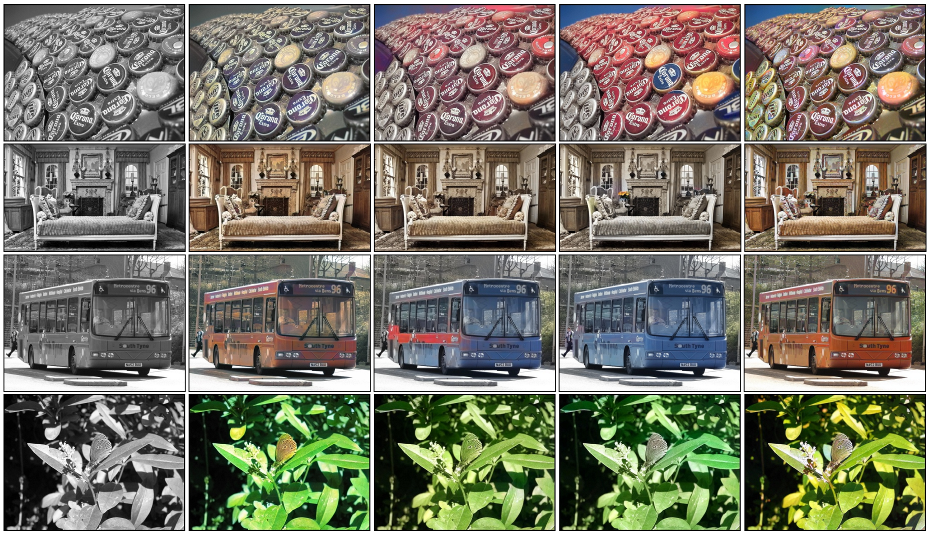
Baselines We focus on the recent learning-based colorization methods for comparisons. Deoldify[2], CIColor [54], UGColor [56], Video Colorization [26], InstColor [38], ColTrans [23], and GPColor [49] are employed for comparisons. Note InstColor is learned with a pretrained object detection model (requiring both labels and bounding boxes), and GPColor exploits a pretrained (on ImageNet with labels) BigGAN. Other approaches including ours are only trained with paired gray-colorful images. For UGColor, we use its fully automatic version where no color hints are used. We use their released model for testing.

Metrics We employ pixel-wise similarity measures PSNR, SSIM, image-level perceptual metric LPIPS [55], and Fréchet Inception Distance (FID) [16] to quantitatively evaluate colorization results. LPIPS and FID are more consistent with human evaluations compared with PSNR and SSIM.

4.2 Quantitative Evaluations

Compared with other methods, our proposed PalGAN (ours^1 in Tab. 1) gives the best perceptual scores (FID & LPIPS) both on ImageNet (FID: 4.60 and 2.78, LPIPS: 0.161 and 0.161 from ctest10K and val50K, respectively) and COCO-Stuff (FID: 7.70, LPIPS: 0.148) without exploiting any annotations or hints, which outperforms other methods. It validates the superiority in realness and diversity of our results. We also achieve competitive fidelity scores (PSNR & SSIM) among all. It shows the well-behaved color restoration ability of PalGAN. If given the ground truth palette, our method (ours* in Tab. 1) can deliver impressive fidelity performance as well as a generative one. It shows the upper bound performance of our method for reference. For methods in Tab. 1 denoted with *, they employ external priors *e.g.* annotations.

Considering the trade-off between fidelity and perceptual results, we can get the best of both worlds on all benchmarks compared with others (ours^2 in Tab. 1) with proper training setting ($\lambda_{adv} = 0.1$ and other regularization coefficients remain still).



(1) Input (2) Deoldify (3) UGColor (4) InstColor (5) Ours

Fig. 6: Visual comparison on ImageNet and COCO-Stuff.

Table 2: User study. Each entry gives the percentage of cases where colorization results are favored compared with GT.

Method	Ours	Coltrans	GPCol	InstCol	Deoldify	UGCol
Rate	**47.20%**	41.50%	39.25%	37.50%	41.13%	42.50%

4.3 Qualitative Evaluations

As shown in Fig. 6, our colorization results give natural, diverse, and fine chrome predictions considering both semantic correspondence and local gradient change. It suffers less from the common color bleeding compared with other methods, owing to chromatic attention. More results are given in Supp.

User Studies Tab. 2 gives human evaluations on our methods with the compared ones. Following the protocol in [54,23], we conduct a colorization Turning test. Specifically, the ground truth color image and its corresponding colorization result (from ours or other methods) are given to 20 participants in random order. These participants need to determine which one is more realistic than the other for no more than 2 seconds. There are 40 colorization predictions from each method, randomly chosen from ImageNet ctest10k. Tab. 2 presents that our method beats the competitors with a large margin.

Colorization of Legacy Photos Though our model is trained in a self-supervised manner using synthetic data, it generalizes well on real-world black-and-white legacy pictures (from [15]), as given in Fig. 7. Color boundaries and consistency are well handled in these cases, working well on the object and portrait.

Reference-based Colorization With the intermediate palette, our approach can conduct reference-based (or example-based) colorization by feeding it with the palette from the reference color image, as given in Fig. 7 and 8. Note even using

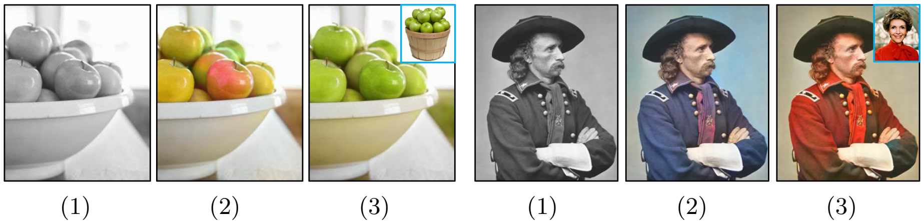


Fig. 7: Our method on legacy images. (1) Inputs, (2) our automatic results, (3) our reference-based results.

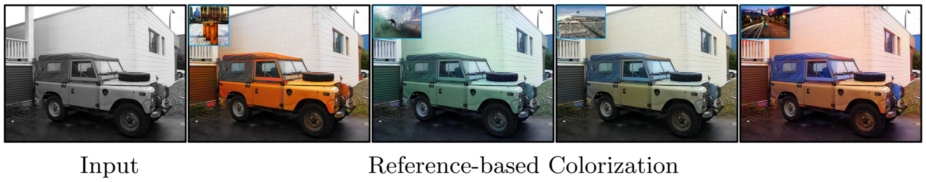


Fig. 8: Our reference-based colorization.

palettes from an image without semantic correlations with the input (Fig. 8), PalGAN still well tunes the given color distribution according to the semantics of the given image, keeping color regionally consistent. Note the car appearances in Fig. 8 present impressive diversity and realness.

4.4 Ablation Studies

Our key designs are ablated on COCO-Stuff as follows.

Palette Prediction and Assignment We validate the effectiveness of our colorization formulation with the proposed model structure compared with a naive autoencoder (AE) and variational one (VAE). Specifically, AE shares the same computational units with PalGAN, except it generates feature maps instead of the palette from its encoder, and utilizes common BN instead of PalNorm in its decoder. VAE is almost the same as PalGAN but it changes the intermediate product palette into a latent vector constrained by Normal distribution. The optimization of AE and VAE is nearly the same as ours except they do not have the palette reconstruction and regularization term, and VAE employs one more term for regularizing the intermediate latent code.

In Tab. 3, we find PalGAN gets significant improvements on FID compared with AE and VAE, while its fidelity performance (PSNR and SSIM) is inferior to AE. It suggests intermediate latent code (in PalGAN and VAE) performs better at color generation than feature maps (in AE), and feature maps excel at fidelity restoration. It validates the effectiveness of our formulation and method on the usage of palette considering its generative performance. Moreover, Fig. 9 illustrates visual differences between varied structures in one example. A high fidelity score of AE does not guarantee the realism of its result.

The effectiveness of the predicted palette is studied. We use palettes from random reference images to simulate failed palette estimations in our method, and give the corresponding evaluation in Tab. 3 (PalGAN w rand ref). It shows the

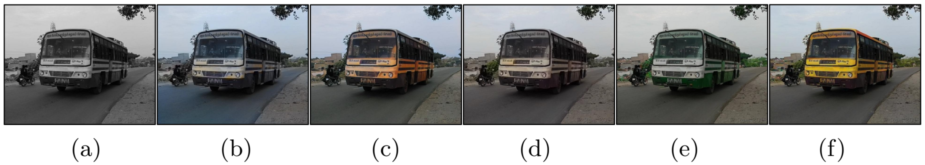


Fig. 9: Ablation studies of model structures. (a) input, (b) AE, (c) VAE, (d) PalGAN w PatchD, (e) PalGAN wo $E(\hat{\mathbf{h}})$. (f) full PalGAN.

Table 3: Quantitative results on COCO-Stuff using different structures.

Structure	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
AE	**25.89**	**0.928**	**0.146**	14.15
VAE	23.21	0.905	0.179	11.76
UGC w CA	24.52	0.923	0.162	11.38
PalGAN w rand ref	20.88	0.883	0.240	9.64
PalGAN w SA	22.68	0.892	0.175	9.02
PalGAN w PatchD [19,41]	23.07	0.895	0.183	8.44
PalGAN w BN	22.36	0.895	0.209	9.97
PalGAN w SPADE [35]	24.06	0.916	0.167	7.90
PalGAN wo $E(\hat{\mathbf{h}})$	24.58	0.924	0.149	8.17
PalGAN	24.56	0.924	0.148	**7.70**

dramatic fidelity and generative performance drop, meaning our palette generator can learn effective chrome distribution for colorization. This is also supported by the visualizations of palettes and their corresponding images in Fig. 3.

Chromatic Attention We explore how the proposed chromatic attention affects colorization, given in Tab. 3 and 4. Compared with naive self-attention (PalGAN w SA in Tab. 3, and SA is applied on the high-level feature maps **S**), our chromatic attention enhances both generative and fidelity performance notably. In Tab. 4, with global interaction in chromatic attention, the generative performance will be improved non-trivially on FID (9.90 → 8.34). It is consistent with the observations in prior image generation works [53,4,43,44] that employing attention will boost generation results. For the local delineation, it focuses on pixel-level restoration, giving notable fidelity increase on PSNR (21.93 → 24.52) and SSIM (0.902 → 0.924). Generally, CA achieves the best of both worlds as it enhances both pixel- and perceptual-level performance. Moreover, we give visual comparison of the ablation study on the chromatic attention in Fig. 5.

Note CA is a generic parametric module. It can be applied to previous methods *e.g.* UGC [56], and it can further improve the corresponding quantitative results (24.34 → 24.51, 0.924 → 0.925, 0.165 → 0.162, and 14.74 → 11.38 on PSNR, SSIM, LPIPS, and FID, respectively).

PalNorm and Color Discriminator In Tab. 3, we find PalNorm yields better quantitative results than BN and SPADE [35] (we use gray-input as semantic layout to generate pixel-wise affine transformation). Besides, PalGAN (default with Color Discriminator) beats PalGAN with Patch Discriminator [41]. These show the effectiveness of our designed PalNorm and Color discriminator.

Table 4: Quantitative results on COCO-Stuff by ablating chromatic attention.

G	L	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓
✗	✗	21.93	0.902	0.203	9.90
✗	✓	24.52	**0.924**	**0.146**	9.97
✓	✗	23.32	0.907	0.174	8.34
✓	✓	**24.56**	**0.924**	0.148	**7.70**

Table 5: Quantitative results on COCO-Stuff about palette with different bins.

#Bins	16	64	256	576	1024
PSNR ↑	23.52	24.48	**24.56**	23.34	23.31
SSIM ↑	0.917	0.919	**0.924**	0.913	0.915
LPIPS ↓	0.172	0.153	**0.148**	0.152	0.159
FID ↓	8.24	7.92	**7.70**	8.04	8.16

Palette Configuration We systematically explore different factors of the employed palette. Tab. 5 shows how the number of bins of palette affects the colorization results. Generally, when #Bins is relatively small, increasing it (16 →256) will lead to a performance increase on almost all used metrics; while #Bins is relatively large, increasing it (256 →1024) will lead to a performance drop. We conjecture this is caused by the tradeoff between the fineness and sparsity of the used palette. The rise of #Bins enhances both its fineness and sparsity. The former reduces ambiguities of palette and the latter makes the optimization of palette reconstruction harder. Empirically, #Bins is 256 is an acceptable setting, used in all experiments.

Also, as given in Tab. 3 (PalGAN wo $E(\hat{\mathbf{h}})$), applying diversity regularization on the estimated palette can improve our generative performance.

Limitation In the user-guided colorization, current PalGAN lacks fine-grained control as we utilize a global palette. In addition, PalGAN cannot well address small-size regions with independent semantics (*e.g.* small objects), since the global interaction in CA cannot well represent these areas using semantic embeddings from small-scale feature maps. Failure cases are given in the supp.

5 Concluding Remarks

In this paper, we study multimodal challenges and color bleeding in colorization from a new perspective. We give a new formulation of colorization for multimodal representation. It introduces the palette as an intermediate variable. This leads to a new and workable colorization method by palette estimation and assignment, yielding diverse and controllable colorful outputs. Additionally, we address the color bleeding issue by explicitly studying color affinities using chromatic attention. It not only leverages semantic affinities to coordinate color, but also exploits the correlation between intensity and their corresponding chrome to delineate color details. Our method is experimentally proven effective and surpasses existing state-of-the-arts non-trivially.

Acknowledgment This work is partially supported by the Shanghai Committee of Science and Technology (Grant No. 21DZ1100100).

References

1. Afifi, M., Brubaker, M.A., Brown, M.S.: Histogan: Controlling colors of gan-generated and real images via color histograms. In: CVPR. pp. 7941–7950 (2021) 1, 4, 8
2. Antic, J.: A deep learning based project for colorizing and restoring old images (and video!), https://github.com/jantic/DeOldify 9, 10
3. Bahng, H., Yoo, S., Cho, W., Park, D.K., Wu, Z., Ma, X., Choo, J.: Coloring with words: Guiding image colorization through text-based palette generation. In: ECCV. pp. 431–447 (2018) 1, 2, 3
4. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018) 4, 13
5. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: CVPR. pp. 1209–1218 (2018) 3, 10
6. Chang, H., Fried, O., Liu, Y., DiVerdi, S., Finkelstein, A.: Palette-based photo recoloring. TOG **34**(4), 139–1 (2015) 3, 5
7. Charpiat, G., Hofmann, M., Schölkopf, B.: Automatic image colorization via multimodal predictions. In: ECCV. pp. 126–139. Springer (2008) 3
8. Chia, A.Y.S., Zhuo, S., Gupta, R.K., Tai, Y.W., Cho, S.Y., Tan, P., Lin, S.: Semantic colorization with internet images. TOG **30**(6), 1–8 (2011) 3
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009) 2, 10
10. Deshpande, A., Lu, J., Yeh, M.C., Jin Chong, M., Forsyth, D.: Learning diverse image colorization. In: CVPR. pp. 6837–6845 (2017) 3
11. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NeurIPS. pp. 2672–2680 (2014) 4
12. Guadarrama, S., Dahl, R., Bieber, D., Norouzi, M., Shlens, J., Murphy, K.: Pixcolor: Pixel recursive colorization. arXiv preprint arXiv:1705.07208 (2017) 2
13. He, K., Sun, J., Tang, X.: Guided image filtering. TPAMI **35**(6), 1397–1409 (2012) 7
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016) 6
15. He, M., Chen, D., Liao, J., Sander, P.V., Yuan, L.: Deep exemplar-based colorization. TOG **37**(4), 1–16 (2018) 3, 11
16. 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: NeurIPS. pp. 6626–6637 (2017) 3, 10
17. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. TOG **35**(4), 1–11 (2016) 2, 3
18. Ironi, R., Cohen-Or, D., Lischinski, D.: Colorization by example. Rendering techniques **29**, 201–210 (2005) 3
19. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR. pp. 1125–1134 (2017) 3, 4, 6, 13
20. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. arXiv preprint arXiv:1812.04948 (2018) 4
21. Kim, E., Lee, S., Park, J., Choi, S., Seo, C., Choo, J.: Deep edge-aware interactive colorization against color-bleeding effects. In: ICCV. pp. 14667–14676 (2021) 3

22. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 10
23. Kumar, M., Weissenborn, D., Kalchbrenner, N.: Colorization transformer. arXiv preprint arXiv:2102.04432 (2021) 2, 3, 9, 10, 11
24. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. In: ECCV. pp. 577–593. Springer (2016) 2, 3, 10
25. Larsson, G., Maire, M., Shakhnarovich, G.: Colorization as a proxy task for visual understanding. In: CVPR (2017) 1
26. Lei, C., Chen, Q.: Fully automatic video colorization with self-regularization and diversity. In: CVPR. pp. 3753–3761 (2019) 2, 9, 10
27. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. In: SIGGRAPH, pp. 689–694 (2004) 1, 3
28. Li, W., Lin, Z., Zhou, K., Qi, L., Wang, Y., Jia, J.: Mat: Mask-aware transformer for large hole image inpainting. In: CVPR. pp. 10758–10768 (2022) 4
29. Liu, X., Wan, L., Qu, Y., Wong, T.T., Lin, S., Leung, C.S., Heng, P.A.: Intrinsic colorization. In: SIGGRAPH Asia, pp. 1–9 (2008) 3
30. Liu, X., Yin, G., Shao, J., Wang, X., et al.: Learning to predict layout-to-image conditional convolutions for semantic image synthesis. In: NeurIPS. pp. 570–580 (2019) 4
31. Liu, Z., Wang, Y., Qi, X., Fu, C.W.: Towards implicit text-guided 3d shape generation. In: CVPR. pp. 17896–17906 (June 2022) 4
32. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. arXiv preprint arXiv:1802.05957 (2018) 10
33. Miyato, T., Koyama, M.: cgans with projection discriminator. arXiv preprint arXiv:1802.05637 (2018) 8
34. Nguyen, R.M., Price, B., Cohen, S., Brown, M.S.: Group-theme recoloring for multi-image color consistency. In: Computer Graphics Forum. vol. 36, pp. 83–92. Wiley Online Library (2017) 3, 5
35. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: CVPR. pp. 2337–2346 (2019) 4, 13
36. Qi, L., Kuen, J., Wang, Y., Gu, J., Zhao, H., Lin, Z., Torr, P., Jia, J.: Open-world entity segmentation. arXiv preprint arXiv:2107.14228 (2021) 4
37. Qu, Y., Wong, T.T., Heng, P.A.: Manga colorization. TOG **25**(3), 1214–1220 (2006) 3
38. Su, J.W., Chu, H.K., Huang, J.B.: Instance-aware image colorization. In: CVPR. pp. 7968–7977 (2020) 2, 4, 9, 10
39. Tai, Y.W., Jia, J., Tang, C.K.: Local color transfer via probabilistic segmentation by expectation-maximization. In: CVPR. vol. 1, pp. 747–754. IEEE (2005) 1, 3
40. Torralba, A., Freeman, W.T.: Properties and applications of shape recipes (2002) 7
41. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: CVPR. pp. 8798–8807 (2018) 4, 13
42. Wang, Y., Chen, Y.C., Tao, X., Jia, J.: Vcnet: A robust approach to blind image inpainting. In: ECCV. pp. 752–768 (2020) 4
43. Wang, Y., Chen, Y.C., Zhang, X., Sun, J., Jia, J.: Attentive normalization for conditional image generation. In: CVPR. pp. 5094–5103 (2020) 13
44. Wang, Y., Qi, L., Chen, Y.C., Zhang, X., Jia, J.: Image synthesis via semantic composition. In: ICCV. pp. 13749–13758 (2021) 4, 13
45. Wang, Y., Tao, X., Qi, X., Shen, X., Jia, J.: Image inpainting via generative multi-column convolutional neural networks. In: NeurIPS (2018) 4

46. Wang, Y., Tao, X., Shen, X., Jia, J.: Wide-context semantic image extrapolation. In: CVPR (2019) 4
47. Welsh, T., Ashikhmin, M., Mueller, K.: Transferring color to greyscale images. In: Proceedings of the 29th annual conference on Computer graphics and interactive techniques. pp. 277–280 (2002) 3
48. Wu, H., Zheng, S., Zhang, J., Huang, K.: Fast end-to-end trainable guided filter. In: CVPR. pp. 1838–1847 (2018) 7
49. Wu, Y., Wang, X., Li, Y., Zhang, H., Zhao, X., Shan, Y.: Towards vivid and diverse image colorization with generative color prior. In: ICCV. pp. 14377–14386 (2021) 2, 4, 9, 10
50. Xia, M., Wang, Y., Han, C., Wong, T.T.: Enhance convolutional neural networks with noise incentive block. arXiv preprint arXiv:2012.12109 (2020) 4
51. Xu, X., Wang, Y., Wang, L., Yu, B., Jia, J.: Conditional temporal variational autoencoder for action video prediction. arXiv preprint arXiv:2108.05658 (2021) 4
52. Yatziv, L., Sapiro, G.: Fast image and video colorization using chrominance blending. TIP **15**(5), 1120–1129 (2006) 1, 3
53. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318 (2018) 13
54. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV. pp. 649–666. Springer (2016) 2, 3, 9, 10, 11
55. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018) 3, 10
56. 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. arXiv preprint arXiv:1705.02999 (2017) 2, 3, 5, 9, 10, 13
57. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV. pp. 2223–2232 (2017) 4
58. Zomet, A., Peleg, S.: Multi-sensor super-resolution. In: Sixth IEEE Workshop on Applications of Computer Vision (WACV). pp. 27–31. IEEE (2002) 7