Disentangled Image Colorization via Global Anchors

MENGHAN XIA, Tecent AI Lab, China
WENBO HU, The Chinese University of Hong Kong, China
TIEN-TSIN WONG, The Chinese University of Hong Kong, China
JUE WANG, Tecent AI Lab, China

(a) Classification (b) Regression (c) Autoregression (d) Ours (e) Ground truth

Fig. 1. Comparison with existing colorization frameworks: classification model [Zhang et al. 2016](a), regression model [Zhang et al. 2017](b), and autoregression model [Kumar et al. 2021](c), and ours (d). The colors of the bus and human clothes are inherently multimodal while involve long-range correlated structures. Notably, our results show superiority in color vividness and structural consistence. Flickr ©Snakebite90; Flickr ©stacey shintani.

Colorization is multimodal by nature and challenges existing frameworks to achieve colorful and structurally consistent results. Even the sophisticated autoregressive model struggles to maintain long-distance color consistency due to the fragility of sequential dependence. To overcome this challenge, we propose a novel colorization framework that disentangles color multimodality and structure consistency through global color anchors, so that both aspects could be learned effectively. Our key insight is that several carefully located anchors could approximately represent the color distribution of an image, and conditioned on the anchor colors, we can predict the image color in a deterministic manner by utilizing internal correlation. To this end, we construct a colorization model with dual branches, where the color modeler predicts the color distribution for anchor color representation, and the color generator predicts the pixel colors by referring the sampled anchor colors. Importantly, the anchors are located under two principles: color independence and global coverage, which is realized with clustering analysis on the deep color features. To simplify the computation, we creatively adopt soft superpixel segmentation to reduce the image primitives, which still nicely reserves the reversibility to pixel-wise representation. Extensive experiments show that our method achieves notable superiority over various mainstream frameworks in perceptual quality. Thanks to anchor-based color representation, our model has the flexibility to support diverse and controllable colorization as well.

CCS Concepts: • **Computing methodologies** → **Image processing**.

Additional Key Words and Phrases: Anchor-based colorization, adaptive anchor locating, structural consistency

ACM Reference Format:
Menghan Xia, Wenbo Hu, Tien-Tsin Wong, and Jue Wang. 2022. Disentangled Image Colorization via Global Anchors. *ACM Trans. Graph.* 41, 6, Article 204 (December 2022), 13 pages. https://doi.org/10.1145/3550454.3555432

Authors' addresses: Menghan Xia, Tecent AI Lab, Shenzhen, China, menghanxyz@gmail.com; Wenbo Hu, The Chinese University of Hong Kong, Hong Kong, China, wbhu@cse.cuhk.edu.hk; Tien-Tsin Wong, The Chinese University of Hong Kong, Hong Kong, China, ttwong@cse.cuhk.edu.hk; Jue Wang, Tecent AI Lab, Shenzhen, China, arphid@gmail.com.

0730-0301/2022/12-ART204 $15.00
https://doi.org/10.1145/3550454.3555432

1 INTRODUCTION

As a classic graphics problem, image colorization has been attracting research interests of the community for decades. Its popularity is mainly attributed to the fascinating applications, e.g., rekindling black and white photos or remastering legacy films, and even assisting artistic expression with color collocation, etc. Among various colorization paradigms, automatic colorization [Zhang

Table 1. Feature comparison among various colorization frameworks.

Framework	Colorfulness	Consistence	Diversity	Efficiency
Regression model	*low*	*fair*	✗	*high*
Classification model	*high*	*low*	✓	*high*
Autoregression model	*high*	*fair*	✓	*low*
Disentangled model (ours)	*high*	*high*	✓	*high*

et al. 2016] is the most general one and serves as the technical basis of other derivatives, such as interactive colorization [Levin et al. 2004], example-based colorization [Li et al. 2017], and text guided colorization [Cho et al. 2018]. Recently, as the prevail of deep learning, data-driven colorization methods make significant progress by exploiting large-scale data priors. Nevertheless, despite sufficient paired data is available for supervision, it still remains challenging to achieve visually satisfactory results because of the hinder of multimodality that one object may take on multiple possible colors. As shown in Fig. 1, the bus or the man's T-shirt is potential to be red, yellow and blue, etc, but the ground truth is just one of them.

Early attempts formulate the colorization as a regression problem under the supervision of the ground truth [Cheng et al. 2015]. They tend to generate desaturated colors because regression losses encourage conservative average modes. Even with semantic guidance [Iizuka et al. 2016; Su et al. 2020] or global attention [Antic 2019] introduced, this problem can only be partially alleviated. To model the color distribution, some works propose to formulate the colorization as per-pixel multinomial classification over the quantized color gamut [Larsson et al. 2016; Zhang et al. 2016]. Nevertheless, it is non-trivial to guarantee structural consistency by pixel-independently color sampling. As a more sophisticated formulation, the autoregressive model is employed to model the pixel color distribution and inter-pixel dependence together [Guadarrama et al. 2017; Kumar et al. 2021], but the long-distance consistence is vulnerable to the error accumulation from sequential dependence.

We argue that image colorization is an internally entangled task. On the one hand, it needs to predict a reasonable color for each pixel based on semantic contexts, which is color-dependent and involves *color multimodality*. On the other, it also needs to preserve the inter-pixel color affinity, which is color-agnostic and involves *structural consistency*. For example, A T-shirt might be red or blue, which is ambiguous, but it is certain that the internal pixels should share the same color. However, existing colorization frameworks generally solve the two sub-tasks through a single prediction, which hence struggle to coordinate them well.

Based on this concept, we propose to solve the colorization problem with a disentangled framework. The key idea is to specially locate several color anchors that can approximately represent the color distribution of the whole image (*multimodal color* only), and then by specifying the anchor colors, the image color can be deterministically predicted by utilizing the global color affinity (*consistent structure* only). To this end, we propose a colorization model with dual branches, i.e. the probabilistic color modeler predicts the color distribution for anchor representation and the color generator predicts the image colors conditioned on the assigned anchor colors. In particular, the anchors are located through clustering analysis on the deep color features, which fulfill the requirements of inter-anchor color independence and global coverage. To avoid the complexity of pixel-wise attention computation, we creatively adopt the soft superpixel segmentation in our model, which not only reduces image primitives in a reasonable way but also suppresses color bleeding by leveraging luminance cues. All these designs enable our method to achieve both vibrant colors and consistent structures. Moreover, by manipulating the color anchors, our model even supports diverse and controllable colorization. Table 1 compares the features of existing mainstream colorization frameworks and ours.

Since colorization aims at generating visually plausible colors rather than recovering the actual ground truth, those ground-truth based fidelity metrics (like PSRN, SSIM, LIPIS) are inapplicable for evaluation, unless the color ambiguity gets removed. Instead, the visual perception related metrics (like FID, IS and Colorfulness) are more reflective to the demanded colorization quality. Quantitative evaluation under perceptual metrics shows that our method outperforms the state-of-the-arts at a clear margin. User study further confirms our superiority in perceptual realism. In addition, ablation studies justify the effectiveness of our technical designs consistently. Our contribution are as follows:

- We propose the disentangled colorization concept for the first time, which contributes novel insight to the long-standing colorization problem.
- We present a novel colorization model that disentangles the color multimodality and structure consistency via global anchors, which pushes forward the state-of-the-art significantly.
- We introduce soft superpixel segmentation to colorization task and demonstrate its advantages.

2 RELATED WORKS

Automatic Colorization. Automatic colorization is generally learning based method since data prior becomes indispensable cues. Early attempts define colorization as a regression problem and solve it with simple prediction models [Cheng et al. 2015; Deshpande et al. 2015]. Recently, some end-to-end CNN models are proposed to promote the performance by integrating some advanced techniques. To encourage global semantic guidance, Iizuka et al. [2016] propose a dual-task structure to jointly learn pixel color prediction and image classification. To learn color priors of separate categories, semantic segmentation maps [Guadarrama et al. 2017; Zhao et al. 2020] or instance-level bounding box [Su et al. 2020] is integrated into the colorization framework. Vitoria et al. [2020] further adopt Generative Adversarial Network (GAN) to promote the realism of colorized results. Antic et al. [2019] develop a popular open-source colorization project that is based on Self-attention GAN [Zhang et al. 2019] and uses lots of engineering tricks to optimize the results. Although remarkable results can be obtained on color-deterministic cases (like natural scenery), these regression based methods still struggle to generate vivid and structurally coherent colors for most man-made scenes that are generally with color uncertainty. Considering this limitation, some probabilistic modeling based methods are proposed to take the color multimodality into account. However, per-pixel probability

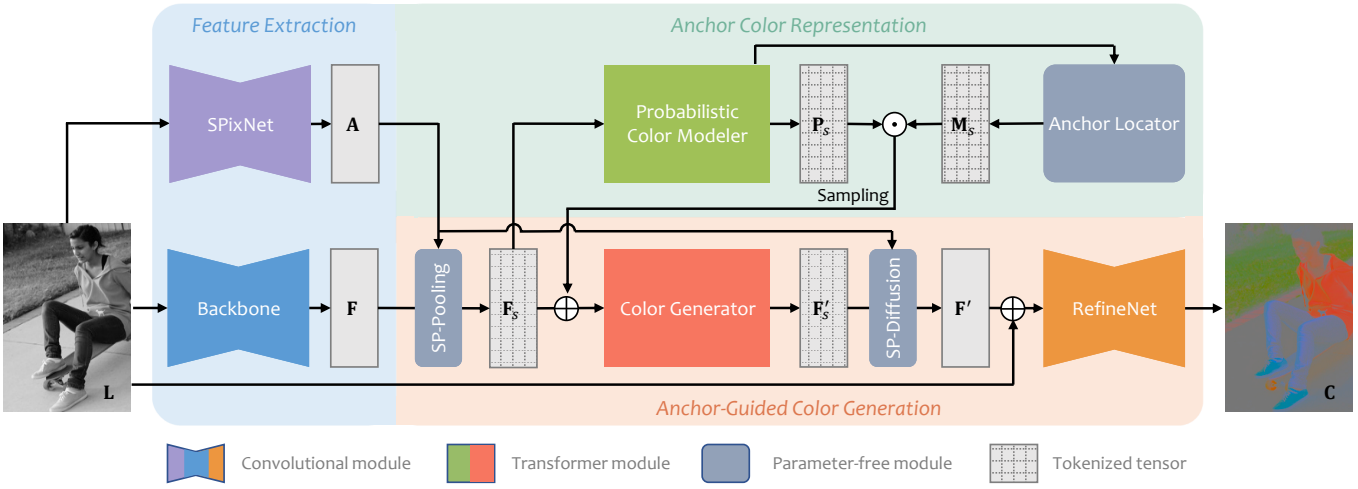


Fig. 2. System overview. Given a grayscale input **G**, we first extract the feature map **F** and the local affinity association map **A**. Then, the tokenized backbone feature **F**′ is used for anchor-based color modeling and anchor-guided color generation respectively. Particularly, the anchors are adaptively located by the anchor locator. Finally, by pixel-level refinement, we obtain the color output **C**. ⊙ denotes Hadamard product and ⊕ denotes channel concatenation.

prediction [Larsson et al. 2016; Zhang et al. 2016] poses challenge to sampling structurally consistent colors, while image-level based modeling [Deshpande et al. 2017; Messaoud et al. 2018] (e.g., mixture Gaussian model) is only feasible to specific category (human face, church, etc) because of limited expressiveness. Stepping forward further, seminal works [Guadarrama et al. 2017; Zhao et al. 2020] propose to model the color multimodality and inter-pixel correlation through a uniform autoregressive model. Most recently, Kumar et al. [2021] utilize Transformer based autoregressive model to further promote the performance. Unfortunately, such model is vulnerable to error accumulation from sequential dependence and thus can not guarantee long-distance consistence well.

Example-Based Colorization. Early example-based methods try to transfer colors from an example image to the grayscale image by utilizing global color statistics [Welsh et al. 2002] or some local similarity measurements [Bugeau et al. 2014; Ironi et al. 2005]. Apparently, such hand-crafted similarity metrics can not guarantee reasonable correspondences between varied image contents and thus tends to cause noticeable artifacts. Recently, some works propose to make use of deep features extracted from pretrained semantic understanding network (i.e. VGG-19) for reliable matching, so that those visual difference but semantically-related content can be transferred between images [He et al. 2019]. However, these methods require a suitable content-related reference to work properly, which poses challenge for users, even with automatic retrieval system assisted [Chia et al. 2011]. To alleviate this obstacle, He et al. [He et al. 2018] propose to learn the example-based colorization mapping from the large-scale dataset, which achieves robustness to the given reference thanks to the learned natural color distributions. Contributing in another line, Wu et al. [Wu et al. 2021] make the first attempt to utilize GAN inversion [Abdal et al. 2019; Gu et al. 2020] techniques to obtain a content-related reference image for example-guided colorization, though its application is restricted by the samples used for GAN training.

Interactive Colorization. Interactive methods require users to provide color hints to guide the colorization process. The pioneering work [Levin et al. 2004] solves a Markov Random Field to propagate sparse color strokes to the whole grayscale image under the assumption that adjacent pixels probably have similar color. The follow-ups leverage more advanced similarity measurements, like texture [Qu et al. 2006], intrinsic distance [Yatziv and Sapiro 2006], etc, to achieve better propagation. While remarkable results can be obtained with careful interactions, it demands intensive manual work and professional skills. By exploiting the capability of deep neural networks, a real-time user interactive colorization model [Zhang et al. 2017] is proposed and only requires sparse color points to generate a visual plausible result. More recently, an edge-enhancing framework is proposed to post-process the color bleeding artifacts by providing interactive scribbles [Kim et al. 2021]. Additionally, global hints like color palettes [Chang et al. 2015] or even cross-modal text description [Cho et al. 2018] are also useful interaction measures to guide the colorization.

3 OVERVIEW

Given a grayscale input, our disentangled colorization model learns to represent the color distribution through several carefully located anchors, and then predict the pixel-wise colors conditioned on the freely sampled anchor colors. Fig. 2 illustrates the model structure that consists of three functional modules, i.e. feature extraction, anchor color representation, and anchor-guided color generation. Specifically, we first extract the shared features **F** and the superpixel association map **A** from the input grayscale $\mathbf{L} \in \mathbb{R}^{H\times W\times 1}$ through the backbone network and *SPixNet* respectively. By pooling **F** with **A**, we get the superpixel based feature tokens $\mathbf{F}_s$ that are then fed to the probabilistic color modeler and color generator separately. Note that, the color modeler predicts the color probabilities $\mathbf{P}_s$ for all primitive tokens, from which the color anchors are selected with the anchor location mask $\mathbf{M}_s$ that is computed by the anchor locator.

Sampling from the color distributions of anchors (i.e. $\mathbf{P}_s^a = \mathbf{M}_s \odot \mathbf{P}_s$), a set of anchor colors $\mathbf{C}_s^a$ and $\mathbf{F}_s$ are fed to the color generator and then pixel-wisely refined by the *RefineNet* to generate the final color output $\mathbf{C} \in \mathbb{R}^{H\times W\times 2}$ in Lab color space.

The advantage of this disentangled model lies in: (i) the color multimodality of the image is represented by the color distribution of several independent anchors, which are free of structural consistency constraint; (ii) once the anchor colors are determined, the image color prediction almost equals to learning global affinity between the anchors and other primitives, which are free of color ambiguity. The anchors are allocated by clustering analysis on colors affinity, which guarantees the anchors with color independence and global coverage. In training phase, we perform clustering on the chromatic channels of the ground truth and select one primitive from each color cluster as anchors. In inference phase, the clustering analysis is conducted on the learned features of the color modeler, which makes similar effects.

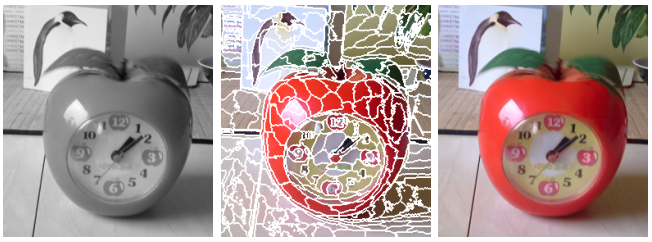
Our model can be trained in an end-to-end manner. The loss function and training details will be introduced in Section 4.4 and Section 5.1 respectively. In inference phase, our model still reserves decent manipulative flexibility for users. Apart from the default automatic colorization, we can further obtain diverse or controllable colorization results by sampling from the anchor color distribution or manually modifying anchor colors and locations.

4 METHODOLOGY

We aim at an automatic colorization method that guarantees vivid colors and consistent structures through disentanglement learning. As diagrammed in Fig. 2, our model adopts two kinds of mainstream network architecture, Convolutional Neural Networks (CNN) and Transformers [Vaswani et al. 2017], because of their respective merits. Particularly, the CNN backbone network holds the major model capacity and extracts features based on structured receptive field. For the superiority in capturing global attentions, two light-weight Transformers (only 3% parameters of the whole model), i.e. the color modeler and the color generator, are additionally employed to capture the global semantics and color correlations. Besides, *SPixNet* predicts a soft association map for superpixel based feature representation, and *RefineNet* further refines the output of the color generator in pixel level. Note that, although multiple modules are used, our model is still in line with most mainstream colorization models in parameter amount and computational efficiency (see Table 5). The network architectures are detailed in the supplementary.

4.1 Local Affinity Aggregation

Digital images generally consist of massive pixels that present strong color correlation within neighborhoods. It is very necessary to reduce the colorization primitives through affinity aggregation. On the one hand, concise primitives facilitate more effective global attention computation (avoiding distraction). On the other, representing a set of color-homogeneous pixels with one color primitive benefits spatial consistency. Specifically, we introduce the soft association map based superpixel segmentation [Yang et al. 2020] to our colorization model, because of the unique merits.



(a) Input grayscale (b) Superpixel segmentation (c) Reconstructed colors

Fig. 3. Association map based superpixel segmentation and reconstruction. For visualization purpose, the segmentation is overlaid on the ground-truth color image (b), and the chromatic channels reconstructed via Eq. 2 are combined with the input grayscale as a color image (c). Flickr ©Emma.

Following [Yang et al. 2020], we represent the superpixel segmentation as a soft association map $\mathbf{A} \in \mathbb{R}^{H\times W\times|\mathcal{N}_p|}$, where each entry $\mathbf{A}(\mathbf{p},\mathbf{s})$ denotes the probability of the pixel $\mathbf{p}$ being assigned to the candidate superpixel $\mathbf{s}$, and we only consider $|\mathcal{N}_p| = 9$ surrounding superpixels as candidates, such that $\sum_{\mathbf{s}\in\mathcal{N}_p}\mathbf{A}(\mathbf{p},\mathbf{s}) = 1, \forall\mathbf{p}$. Note that, the superpixels are initialized by partitioning the image into a regular grid of size $\tilde{H}\times\tilde{W}$ (where $\tilde{H} = \frac{H}{S}, \tilde{W} = \frac{W}{S}$ and S is the grid cell size.), namely each grid cell is an initial superpixel seed. Given the predicted association map $\mathbf{A}$ and the pixel property $\mathbf{F}$ (e.g. pixel values or other features), we can compute the center of any superpixel $\mathbf{s} = \{\mathbf{v}_s, \mathbf{l}_s\}$, where $\mathbf{v}_s$ is the property vector and $\mathbf{l}_s$ is the location vector, as follow:

$$\mathbf{v}_s = \frac{\sum_{\mathbf{p}\in\mathcal{P}_s}\mathbf{F}(\mathbf{p})\cdot\mathbf{A}(\mathbf{p},\mathbf{s})}{\sum_{\mathbf{p}\in\mathcal{P}_s}\mathbf{A}(\mathbf{p},\mathbf{s})}, \quad \mathbf{l}_s = \frac{\sum_{\mathbf{p}\in\mathcal{P}_s}\mathbf{p}\cdot\mathbf{A}(\mathbf{p},\mathbf{s})}{\sum_{\mathbf{p}\in\mathcal{P}_s}\mathbf{A}(\mathbf{p},\mathbf{s})}. \tag{1}$$

Here $\mathcal{P}_s = \{\mathbf{p}_i|\mathbf{s}\in\mathcal{N}_{p_i}\}$ denotes the pixel set with a possibility to be assigned to the target superpixel $\mathbf{s}$. This is actually an average pooling operation within each superpixel, called *SP-Pooling*. Inversely, given the association map $\mathbf{A}$ and superpixel representation, we can also reconstruct the pixel property $\mathbf{F}'$, as follow:

$$\tilde{\mathbf{F}}(\mathbf{p}) = \sum_{\mathbf{s}\in\mathcal{N}_p}\mathbf{v}_s\cdot\mathbf{A}(\mathbf{p},\mathbf{s}), \quad \tilde{\mathbf{p}}(\mathbf{p}) = \sum_{\mathbf{s}\in\mathcal{N}_p}\mathbf{l}_s\cdot\mathbf{A}(\mathbf{p},\mathbf{s}). \tag{2}$$

Accordingly, we call this operation *SP-Diffusion*. In our method, the association map $\mathbf{A}$ is predicted by the *SPixNet* under the supervision of the self-reconstruction of the ground-truth colors $\mathbf{C}^{gt}$ (see Section 4.4 for details). Fig. 3 illustrates an example that the soft association map enables faithful color gradient reconstruction from the superpixels, which would be impossible by those traditional hard superpixel segmentation. Here, we visualize the superpixels by assigning each pixel $\mathbf{p}$ to the grid cell with the highest probability: $\hat{\mathbf{s}} = \arg\max_{\mathbf{s}}\mathbf{A}(\mathbf{p},\mathbf{s})$.

4.2 Color Anchor Construction

Considering the semantic correlations among image primitives, we are motivated to represent the multimodal colors of the whole image with some sparse color anchors. The anchors should be allocated under some requirements. First, these anchors are uncorrelated in terms of their potential colors, so that we can assign colors to them independently yet risks no structure inconsistency. Second, their colors are representative to the whole image so that no color

(a) Ground-truth color (b) Clusters on ground-truth color (c) Clusters on Learned color features

Fig. 4. Anchor location by feature clustering. For visualization, we assign each cluster with a unique color, from which the selected anchor is marked with a star. $K = 5$ anchors are used in this example.

ambiguity exists once their colors are determined. To achieve this, we propose a simple yet effective solution, i.e. clustering the primitives based on color affinity and taking one primitive from each color cluster as the anchor. To learn unbiased color features, we employ the color modeler to model the color distribution for all primitives (i.e. superpixels).

Given the backbone feature $\mathbf{F}$, we first tokenize it to superpixel primitives $\mathbf{F}_s$ through *SP-Pooling* and feed them to the Transformer based probabilistic color modeler with position encoding applied. Then, we obtain the color probabilities of each primitive $\mathbf{P}_s \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times 313}$, where the color gamut is quantized into 313 color bins as did in [Zhang et al. 2016]. Although spatial inconsistency may occur if we sample the primitive color independently, we observe that the learned color features (i.e. the feature before the linear projection layer) hold similar color affinity as the ground-truth color. Fig. 4 illustrates an example to compare the clustering results under the two paradigms. The quantitative evaluation is studied in Section 5.4. One thing worthy noting is that the learned color features are only ready for clustering when the training completes. So, we perform clustering on the ground-truth colors in training phase, while we do that on the learned color features in inference phase. Particularly, among the primitives of each cluster, we simply take the one with the largest size (i.e. the pixel amount under hard segmentation) as the anchor. The motivation is to exclude those superpixels that locate nearby object boundaries or regions with busy structures, where the superpixels are usually segmented with fine/small size. The selected anchor locations are denoted by a binary mask $\mathbf{M}_s$, where one means anchors and zero means other primitives. So, the color distribution of the input image can be approximately represented by the probabilistic color of the anchors $\mathbf{P}_a = \mathbf{P}_s \odot \mathbf{M}_s$. In theory, any clustering algorithm is applicable to our problem and we adopt K-Means because of its stable performance. Mean-Shift is well-known as determining the cluster numbers automatically but its drastic sensitivity to hyper-parameters makes it infeasible to our problem.

4.3 Anchor-Guided Color Generation

The colorization ambiguity could be removed by specifying the anchor colors $\mathbf{C}_s^a$, since they represent the color distribution of the whole image. In this way, it encourages the model to exploit the internal color correlation and hence promote structural consistency. Accordingly, our color generator takes as input the concatenation of $\{\mathbf{F}_s, \mathbf{C}_s^a, \mathbf{M}_s\}$ and outputs the conditional color feature $\mathbf{F}_s'$ that can

(a) Independently sampled probabilistic color (b) Anchor-guided color w/o refinement (c) Anchor-guided color with refinement

Fig. 5. Robustness to anomalous anchors. Top row: multiple color anchors are assigned to the color-homogeneous pillow, which causes conflicting color guidance (b). Bottom row: no color anchor is assigned to remove color ambiguity of the right-side hood, which causes structural inconsistency (b). For both cases, the luminance-guided refinement improves the structural consistency effectively (c). Anchors are marked with white boundary in (b). The top image is from Flickr ©Erin Nealey.

be mapped into the conditional color probabilities $\mathbf{P}_s'$ through a side linear projection layer (not shown in Fig. 2 to avoid distraction). To further refine the colors, we employ the *RefineNet* to generate the final colors $\mathbf{C}$ from the combination of the input grayscale $\mathbf{L}$ and the pixel-wise conditional feature $\mathbf{F}'$ that is diffused from $\mathbf{F}_s'$ by *SP-Diffusion*. Here, the grayscale input makes effects in providing luminance cues to identify and correct color artifacts. During training, the anchor colors are assigned with the ground-truth colors, so that we can take the ground truth for supervision. For inference, we can freely sample the anchor colors from the predicted distribution to guide the color generation.

Our anchors sometimes are not perfectly located, because the anchor number is an empirically hyper-parameter (i.e. the cluster number of K-Means). For example, when the anchors are not enough or more than needed, the anchor-guided colorization may suffer from color ambiguity or conflicting color guidance. Fig. 5 illustrates two typical examples. Fortunately, our model inherently has some features to cope with such accidents. First, the probabilistic color modeler shares the same backbone feature with the anchor-guided color generator, whose structural consistent property, as encouraged by the anchor-guided branch, also benefits the predicted probabilistic colors. As shown in Fig. 5(a), the independently sampled primitive colors present certain structural consistency already. It means that even if more than one anchors are located in a color-homogeneous region, they still have a good chance to be assigned with consistent colors by independently sampling. Besides, as illustrated in Fig. 5(c), our *RefineNet* plays an important role to correct inconsistent colors by referring to the luminance cues. This feature benefits from joint training, which is studied in Section 5.4.

4.4 Loss Function

Our model is trained with three loss terms that supervise the learning of superpixel segmentation, color distribution modeling, and conditional color generation respectively.

Affinity aggregation loss. We use a similar training objective as [Yang et al. 2020] to supervise *SPixNet*. The major difference is that we use the ground-truth chromatic channels $\mathbf{C}^{gt} \in \mathbb{R}^{H\times W\times 2}$ for self-reconstruction while take the grayscale $\mathbf{L}$ as input. This is possible because adjacent pixels with similar luminance generally have similar color [Levin et al. 2004]. Specifically, the affinity aggregation loss $\mathcal{L}_{aggr}$ is formulated to group pixels with similar chromatic property and enforce the superpixels to be spatially compact, written as:

$$\mathcal{L}_{aggr} = \frac{1}{N}\sum_{\mathbf{p}} ||\mathbf{C}^{gt}(\mathbf{p}) - \tilde{\mathbf{C}}^{gt}(\mathbf{p})||_2 + \frac{\alpha}{S}||\mathbf{p} - \tilde{\mathbf{p}}||_2. \tag{3}$$

Here $\tilde{\mathbf{C}}^{gt}$ and $\tilde{\mathbf{p}}$ are computed through Eq. 2 and Eq. 1. $N = H \cdot W$. S is the superpixel sampling interval, and α is a weight balancing the two terms. We empirically set $S = 16$ and $\alpha = 3e^{-4}$ in our experiment.

Color distribution loss. As stated above, the output of the probabilistic color modeler is $\mathbf{P}_s \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times 313}$, where the color distribution is represented by a 313-way probability vector. To construct the supervision signal, we first apply *SP-Pooling* to $\mathbf{C}^{gt}$ and then convert the color values into one-hot representation $\mathbf{P}_s^{gt}$. Alike to classification tasks, we adopt cross entropy to compute the color distribution loss $\mathcal{L}_{dist}$, where $\tilde{N} = \tilde{H} \cdot \tilde{W}$ for normalization.

$$\mathcal{L}_{dist} = \frac{1}{\tilde{N}}\sum_{\mathbf{s}} -\mathbf{P}_s^{gt}(\mathbf{s}) \log \mathbf{P}_s(\mathbf{s}). \tag{4}$$

Conditional generation loss. The anchor-guided color generation is supervised in two aspects. First, the color generator is required to generate per-primitive colors by referring the color correlation between primitives, especially between anchors and other primitives. Considering the possible color ambiguity, we adopt cross entropy to measure the discrepancy of the superpixel based conditional color distribution $\mathbf{P}'_s$. Second, the *RefineNet* is required to correct and refine the conditional color features toward better perceptual quality. So, our conditional generation loss $\mathcal{L}_{color}$ is defined as:

$$\mathcal{L}_{color} = \frac{1}{\tilde{N}}\sum_{\mathbf{s}} -\mathbf{P}_s^{gt}(\mathbf{s}) \log \mathbf{P}'_s(\mathbf{s}) + \beta \sum_{l} \omega_l ||\Phi_l(\mathbf{C}^{gt}) - \Phi_l(\mathbf{C})||_1. \tag{5}$$

Here Φ_l is the feature maps for l-th layer of a pretrained VGG-19 network [Simonyan and Zisserman 2015], where the five layers $\{conv1\_1, conv2\_1, conv3\_1, conv4\_1, conv5\_1\}$ are adopted. ω_l denotes the weight for each layer. $\beta = 5.0$ is empirically set to balance the magnitude difference of the two terms. During training, the anchors are assigned with the ground-truth color to make the supervision signal applicable.

Overall, we first train the *SPixeNet* individually with $\mathcal{L}_{aggr}$ until converges. Then, with the pretrained *SPixNet* frozen, the model is trained from scratch under the loss function $\mathcal{L} = \mathcal{L}_{dist} + \lambda\mathcal{L}_{color}$. We set $\lambda = 1.0$ to emphasize the color distribution modeling and anchor-guided color generation equally.

5 EXPERIMENTAL RESULTS

We evaluate our method through quantitative and qualitative comparison with existing state-of-the-art methods over multiple representative datasets. The perceptual realism is assessed via user study. Besides, ablation studies are conducted to identify how our method works.

5.1 Implementation Details

We train *SPixNet* using batch size 256 and linearly decaying the learning rate from $lr = 2e^{-4}$ to $2e^{-6}$ within 20 epochs. With the pretrained *SPixNet* frozen, we train the full model using batch size 96 and linearly decaying the learning rate from $lr = 2e^{-4}$ to $2e^{-6}$ within 60 epochs. Adam [Kingma and Ba 2014] optimizer with $\beta_1 = 0$ and $\beta_2 = 0.9$ is used. Our model is trained on the training set of ImageNet [Deng et al. 2009] only but evaluated on multiple validation set of other datasets. Images in training are resized to a fixed size (256×256), though our model can process images of arbitrary resolution in inference. Unless specified, we use the default trained model ($K = 8$ color anchors) to generate results for all these quantitative and qualitative evaluations, without any case by case hyper-parameter tuning involved. Our source code and pretrained model are released at: https://github.com/MenghanXia/DisentangledColorization.

Prior arts. We compare our method with recent learning-based automatic colorization methods, including three categories. (i) **Classification model**: CIColor [Zhang et al. 2016] that predicts per-pixel color probabilities. (ii) **Regression model**: UGColor [Zhang et al. 2017] that utilizes random color hint to simulate user interaction during training while supports automatic colorization (i.e. feeding no hint) in inference; InstColor [Su et al. 2020] that employs instance detection for individual color prediction; Deoldify [Antic 2019] that adopts self-attention to utilize long-distance correlations; ChromaGAN [Vitoria et al. 2020] that employs a dual-branch structure for joint image classification and color prediction. (iii) **Autoregression model**: ColTran [Kumar et al. 2021] that use an encoder-decoder Tranformer for low-resolution color autoregression, with additional color and spatial enhancement followed. For evaluation, we use their official codes and released model weights, which are either trained on the same training set as ours or additionally fine-tuned on COCO-Stuff.

Dataset. Following the prior practice [Su et al. 2020; Zhang et al. 2016], we perform evaluations on the ImageNet ctest [Deng et al. 2009] (10k images) that is a subset of the ImageNet validation split used as a standard evaluation benchmark, and the validation set of COCO-Stuff [Caesar et al. 2018] (5k images). Additionally, we take the legacy photo dataset [Luo et al. 2020] that contains about 37k historical black-and-white photos to check the performance on real-world legacy photos.

5.2 Qualitative Evaluation

Most colorization methods work quite well on natural scenes, since vegetation, sky, lake/sea and animals have little color ambiguity. However, they are still challenged to achieve satisfactory results on man-made scenes, such as human clothes, vehicles,

(a) Input (b) UGColor (c) Deoldify (d) ColTran (e) Ours (f) Ground truth

Fig. 6. Comparative showcase of challenging examples. Flickr ©L Sanford; Flickr ©SonnyandSandy; Flickr ©Morning Calm Weekly Newspaper Photo Archive; Flickr ©Ray Forster; Flickr ©Paul; Flickr ©Shinya Suzuki; Flickr ©mirsasha; Flickr ©hmmlargeart.

sport facilities, and articles of daily use, etc, which can present diverse colors potentially. In Fig. 6, we make visual comparison on several typical examples with the most competitive state-of-the-art methods, UGColor [Zhang et al. 2017], Deoldify [Antic 2019], and ColTran [Kumar et al. 2021], while the result of other methods are available in the supplementary. Readers can check more qualitative comparison in the supplementary. In general, we can inspect the colorization quality in three aspects, i.e. color

(a) Input (b) UGColor (c) Deoldify (d) ColTran (e) Ours

Fig. 7. Comparative showcase of colorizing historical photos (no ground truth available). Images are from the KeystoneDepth dataset (Public Domain).

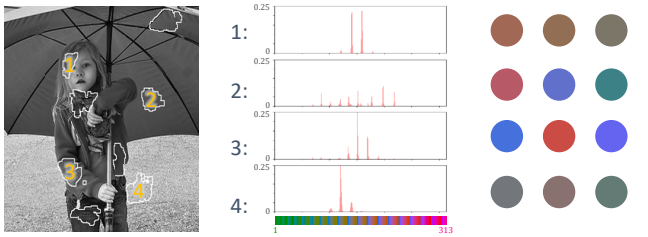
realism (for example, human face can not be green), colorfulness, and structure consistency (the color-homogeneous parts should be assigned with the same color). As we can observe, UGColor and Deoldify tend to generate desaturated colors and introduce spatial inconsistency, which is the common weakness of regression based methods in front of color multimodality. Benefiting of the sequential dependence modeling through autoregression, ColTran achieves a relatively better performance, including the colorfulness and structure consistence. However, it is still common to see artifacts in local parts, such as the trousers of the man (third row (d)) and the athletes' left leg (bottom row (d)). In contrast, our method (e) achieves noticeable superiority in all aspects, which generates visual plausible appearances with outstanding structural consistency and colorfulness. This is mainly attributed to the advantages of our disentangled colorization paradigm, where the multimodal colors and structural consistency are well guaranteed through separate branches and in a non-interfering manner. Moreover, the adoption of superpixel primitives avoid the color bleeding issue by utilizing the local luminance cues. In Fig. 7, we make comparison on real-world legacy black-and-white photos and our method still holds the advantages mentioned above. It indicates the good generalization of our model since it is only trained on the training split of ImageNet.

User study. There is no generally accepted criterion for colorization evaluation and human visual system is still the most reliable measure. We conduct a user study to evaluate the colorization in perceptual realism. To prepare the evaluation set, we randomly select 30 samples from the validation set of COCO-Stuff and 27 valid color images are used, which covers both natural scenes and man-made scenes (as provided in the supplementary). For each sample, the participants are shown a series of color images, including the colorized images by different methods and the ground-truth image, placed in random order. Then, the participants are asked to rank the top-3 images in terms of perceptual realism, i.e. the best, the second-best, and the third-best. The motivation here is to simplify the evaluation but still introduce certain discriminability between different results. In this study, 54 participants with good vision and color recognition complete the evaluation successfully. We analyze the collected evaluation result in two aspects: (i) *Rank score*: the score of each method according to the rank-score table: {the-best:3; second-best:2; third-best:1, others:0}; (ii) *Best-rated*: the times of being selected as the best.

The statistics are tabulated in Table 2. Overall, our results are regarded as more perceptually realistic than other competitors but still lag behind the ground-truth images. Besides, the superiority of our method to the second-best method Deoldify is not as large as we expected from the visual comparison in Fig. 6. It can be explained by the fact that existing methods only struggle in man-made scenes but work well in natural scenes while the randomly

Table 2. Statistic from user study. The rank-score table is: {the-best:3; second-best:2; third-best:1, others:0}. Best-rated counts the times of being selected as the best. The sum over all the 27 examples are used.

Property	InstCol	UGColor	Deoldify	ColTran	Ours	Ground truth
Rank score ↑	23.69	17.30	26.67	8.28	28.31	57.76
Best-rated ↑	1.96	1.56	2.94	0.78	4.13	15.63



(a) Anchor locations (b) Color distribution (c) Top-3 peak colors

Fig. 8. Visualization of the color distribution of representative color anchors. Flickr ©Will-travel.

(a) Automatic colorization (b) Diverse sampling (c) User manipulation

Fig. 9. Diverse and controllable colorization. For (a) and (b), the automatically located 8 anchors are used. For (c), the anchor locations and their assigned colors are modified manually. The anchors are marked with white boundary on the final result. Flickr ©Will-travel; Flickr ©Timo Kuusela.

selected evaluation set contains many natural scenes, which brings down the distinctiveness of our method in some degree.

Diverse and controllable colorization. In our model, we represent the potential color distribution of the input image with several color anchors, each of which is associated with individual probabilistic colors. Fig. 8 illustrates the color distribution of some anchors, which generally present multiple peaks. Interestingly, the anchors with higher color ambiguity (e.g., the umbrella) tend to have more peaks and more diverse colors, and vice versa. Thanks to such anchor-based color representation, our trained model has the flexibility to support diverse and controllable colorization. Specifically, by sampling different anchor color values, we can obtain diverse colorization results, as exampled in Fig. 9(a),(b). In addition, users can even control the colorization by manually modifying the anchor locations, e.g., removing anchors or adding new anchors, and assigning any preferred colors to them. Fig. 9(c) demonstrates two examples that are colorized with user manipulated anchors. It brings advantages in two folds. First, it allows users to achieve their intended colorization results through color selection and modification. Second, it also serves as a measure to address artifacts caused by abnormally allocated anchors. Fig. 9 evidences the consistent spatial affinity between the colorized images despite conditioned on varied anchor colors, which indicates the effective disentanglement of the color multimodality and structural consistency in our model.

5.3 Quantitative Evaluation

Metrics. Considering the color multimodality, the goal of colorization is to generate visually plausible colors rather than recover the actual ground truth. As a result, we quantitatively evaluate the colorization performance in two aspects, i.e. perceptual realism and color vividness. Frechet' Inception Score (FID) [Heusel et al. 2017] measures the distribution similarity between the colorization results and the ground truth, which reflects the perceptual realism in some sense. When no ground-truth dataset available (such as the Legacy-Photo dataset), we adopt Inception Score (IS) [Salimans et al. 2016] to measure the perceptual realism. Colorfulness [Hasler and Süsstrunk 2003] reflects the color vividness in the way of human vision perception. According to the paper, the moderately colorful, averaged colorful and quite colorful values of general images are 33, 45 and 59 respectively. In addition, following the practice of previous methods [Su et al. 2020; Vitoria et al. 2020; Zhang et al. 2017], we also provide the evaluation measured by PSNR, SSIM, and LIPIS [Zhang et al. 2018] just for reference. But we argue that these ground-truth dependent metrics can not well reflect the actual colorization performance because plausible colorization probably diverges largely with the ground truth.

As the quantitative comparison shown in Table 3, the regression based methods, such as UGColor [2017], Deoldify [2019], InstColor [2020], ChromaGAN [2020], generally perform better on PSNR, SSIM, and LPIPS, since they consider the colorization as a deterministic color regression problem. However, due to this less reasonable assumption, their results usually degrade into desaturated colors and even introduce spatially inconsistent colors, which are reflected by the inferior colorfulness and FID. In contrast, CIColor [2016], ColTran [2021] and ours model the color distribution explicitly and thus achieves notably higher colorfulness. As diverse colors is learned, their performance on the fidelity metrics (i.e. PSNR, SSIM, and LPIPS) are much lower than those regression based methods. Interestingly, once the color ambiguity is removed by providing several ground-truth color hints, our method (as ours* in Table 3) achieves competitive performance in those metrics. The adoption of ground-truth color hints also improves the FID

Table 3. Quantitative results on the validation sets from different methods. The best items are highlighted in bold.

Method	ImageNet (10k)					COCO-Stuff (5k)					Legacy-Photo (37k)	
	FID ↓	Colorfulness ↑	PSNR ↑	SSIM ↑	LPIPS ↓	FID ↓	Colorfulness ↑	PSNR ↑	SSIM ↑	LPIPS ↓	IS ↑	Colorfulness ↑
CIColor [2016]	11.58	42.89	21.96	0.897	0.224	21.44	42.46	22.08	0.902	0.217	11.04	32.36
UGColor [2017]	6.85	27.91	**24.26**	**0.919**	**0.174**	14.74	28.64	**24.34**	**0.924**	**0.165**	11.81	20.11
Deoldify [2019]	5.78	23.41	23.34	0.907	0.188	12.75	23.62	23.49	0.914	0.181	12.88	24.45
InstColor [2020]	7.35	25.54	22.03	0.909	0.919	12.24	29.38	22.35	0.838	0.238	12.00	19.64
ChromaGAN [2020]	9.60	29.34	22.85	0.876	0.230	20.57	29.34	22.74	0.871	0.233	11.75	23.28
ColTran [2021]	6.37	38.64	21.81	0.892	0.218	11.65	38.95	22.11	0.898	0.210	13.08	11.69
Ours	**5.57**	**51.43**	20.72	0.862	0.229	**10.59**	**52.85**	20.46	0.851	0.236	**13.72**	**32.67**
Ours*	2.76	33.70	27.57	0.918	0.125	6.48	32.16	26.73	0.908	0.138	N/A	N/A

(a) Ground truth (b) No anchor branch (c) Mini-patch primitive (d) No refinement (e) Post-refinement (f) Ours

Fig. 10. Visual comparison from ablation studies. (b)~(e) denotes the result of our method with one specific design ablated. White arrows indicate the artifacts. Flickr ©Chris; Flickr ©mirsasha.

because the anchor colors assigned from the ground truth are free of causing conflictive color guidance. Similarly, we can reduce such cases by using less anchor numbers, which also improves FID scores (see Fig. 11). FID (and IS) is a relatively reliable metric that mainly criticizes structural inconsistency of colors while not so picky at color vividness. Anyway, our method holds the best performance in colorfulness and FID (or IS) over all datasets, thanks to the superiority of our disentangled colorization framework. For different datasets, COCO-Stuff is more challenging than ImageNet because the former mainly contains multiple-object images while the latter is dominated by single-object images. In the Legacy-Photo dataset, ColTran [2021] has an unstable performance because of the fragility of sequential inference in complex scenes. It sometimes even generates grayscale-alike results, as evidenced in Fig. 7 (top) and the colorfulness metric.

5.4 Ablation Study

We study several major designs of our proposed method in this section. For comparative analysis, we perform quantitative evaluation on the COCO-Stuff dataset with FID and colorfulness measured.

Disentangled structure. The main advantage of our method lies in the proposed disentangled paradigm, which makes the color multimodality and structural consistency learned separately. To verify this claim, we construct a baseline by removing the anchor representation branch, which becomes a regression based model. Alike to other regression based methods, it suffers from color ambiguity and tends to cause desaturated colors and structural inconsistency. The visual results shown in Fig. 10 evidence this phenomena. However, the quantitative evaluation is presented in Table 4, showing that the baseline achieves much lower colorfulness but even better FID. It is possibly benefit from our other designs like affinity aggregation and joint refinement, which have considerable advantages on suppressing artifacts. In some sense, the conservative colorization that assigns less vivid colors to avoid dramatic color inconsistency tends to help on FID. This situation can also be observed in Fig. 11. That is to say, FID and colorfulness should be considered comprehensively in order to reflect the visually perceived colorization quality.

Affinity aggregation. We employ soft association based superpixels to reduce colorization primitives, which is a kind of affinity based aggregation. A common practice used by existing methods [Kumar et al. 2021; Zhang et al. 2016] is to shrink the output resolution by assigning each mini-patch with a single color. To show our advantage, we construct a baseline by replacing the superpixel segmentation with mini-patch partition. The quantitative results are shown in Table 4. This superiority comes from the higher accuracy of aggregating pixels based on luminance/color affinity rather than naively assuming the pixels within a local patch sharing the same

Table 4. Quantitative results of ablation studies.

Studied component	Model variant	FID ↓	Colorfulness ↑
Disentangled structure	No anchor branch	9.77	24.98
	Dual branch	10.59	52.85
Affinity aggregation	Mini-patch primitive	11.52	50.78
	Soft superpixel	10.59	52.85
Pixel-level refinement	No refinement	14.70	56.71
	Post refinement	11.49	46.88
	Joint refinement	10.59	52.85
Clustering feature	Ground-truth color	10.11	52.54
	Learned Feature	10.59	52.85

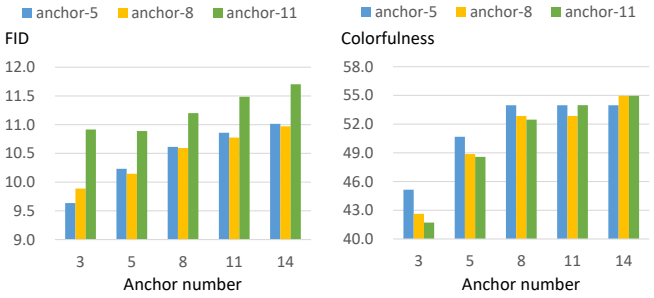


Fig. 11. Effects of the number of anchors. Anchor-5, anchor-8, and anchor-11 denote three models that are trained with 5, 8, 11 anchors respectively. They are evaluated by adopting various number of anchors in inference.

color. In Fig. 10(c), we can observe some color-bleeding effect on the girl's neck because of the less reasonable aggregation assumption.

Pixel-wise Refinement. As we discussed in Section 4.3, the *RefineNet* plays an important role for artifacts suppression by utilizing the luminance cues for color correction. We speculate that the jointing training is a key requirement to achieve that advantage. To verify this, we build two baselines: (i) *No refinement* that the model is trained without the *RefineNet*; (ii) *Post-refinement* that the *RefineNet* is trained individually as a post-processing for (i). The quantitative comparison is presented in Table 4, which shows that both *No refinement* and *Post-refinement* can not compete with our proposed joint training. In Fig. 10, we can find that some local artifacts introduced in (d) but still remain in (e). In contrast, our results (f) take on perceptually realistic appearances.

Clustering Feature. In Section 4.2, we propose to utilize the learned color feature for primitive clustering, so as to allocate anchors with color independence and global coverage. To study the effectiveness, we evaluate the result of our method when the ground-truth color is used for clustering for comparison. The quantitative results are presented in Table 4, which shows that our learned feature works similarly as the ground-truth color in aspect of anchor location. Besides, we also provide some visual comparison with another naive baseline that randomly scatters the anchors, in the supplementary.

The number of anchors. We use color anchors to represent the color distribution of the input image, so it is necessary to study the

Fig. 12. Colorization with different number of anchors used. The model trained with 8 anchors is evaluated here.

influence when different number of anchors are used. Specifically, we construct three model variants: anchor-5, anchor-8, anchor-11, which are trained with 5, 8, 11, anchors respectively. For each variant, we can choose arbitrary number of anchors to use in inference phase. Particularly, we evaluate these models with 3, 5, 8, 11, 14 color anchors used respectively, and show the quantitative results in Fig. 11. For all these models, colorfulness improves but FID increases as more anchors are used in inference. The explanation is that more color anchors means higher probability to cause contradictory color guidance, which damages perceptual realism and hence increases FID. Anyhow, more anchor anchors could introduce more colorful image components and thus increases colorfulness. So, there is a trade-off in some sense. We provide an example in Fig. 12 to demonstrate this situation. Besides, we observe that the model anchor-8 achieves the best comprehensive performance in most cases. Although the optimal anchor number depends case by case, we empirically choose the model *anchor-8* with 8 color anchors used in inference as our default setting. Note that, our model has decent robustness to anchor location, as already demonstrated in Fig. 5.

5.5 Efficiency Analysis

Our model has a CNN-and-Transformer hybrid structure that includes three functional blocks: feature extraction, anchor construction, and color generation. To analyze the computational efficiency, we count the workload of each block in the aspects of model parameters, floating-point operations (FLOP), and inference timing, as illustrated in Fig. 13. We test the model on an NVIDIA V100 GPU by sequentially feeding 100 images that are randomly sampled from COCO-Stuff dataset and resized to 256×256, and then take the average time as the inference timing. We observe that the major computation happens at feature extraction and color generation since the U-shaped CNNs, including backbone network, *SpixNet*, and *RefineNet*, take up most of the model parameters. In contrast, the two lightweight Transformers, i.e. the probabilistic color modeler and the color generator, only have about 3% of the total parameters (1.33M out of 43.06M). Notably, despite the parameter amount and FLOPs are smaller, the anchor construction consumes the most time with respect to the other two blocks. This is because the anchor location has K-Means clustering involved, which introduces iterative computation.

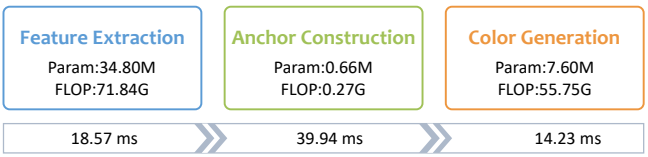


Fig. 13. Computation division of our model. The inference timing is about forward-passing a single image sized 256 × 256 on an NVIDIA V100 GPU.

Table 5. Efficiency comparison on different colorization models.

Method	Param. (M)	FLOP (G)	Timing (ms)
CIColor [2016]	32.24	41.78	7.29
UGColor [2017]	34.19	75.22	15.88
Deoldify [2019]	218.22	141.32	29.85
InstColor [2020]	176.89	526.09	184.75
ChromaGAN [2020]	47.52	18.85	3.65
ColTrans [2021]	70.71	5782	110708.47
Ours	43.06	127.86	72.74

In Table 5, we provide the efficiency comparison with existing colorization methods. Alike to application scenario, we only count the modules that are used during inference. Except for ColTran that conducts sequential pixel inference, all the other methods generate the result through a single forward. ColTran has very huge computation because it is a purely Transformer architecture. InstColor has a slow inference because it requires object detection for individual colorization. Overall, our model holds a fair efficiency performance while achieves a significant superiority in colorization quality.

5.6 Limitation and Discussion

As a common weakness of automatic colorization, our method can not generate plausible colors for semantically impenetrable image content, i.e. the objects or scenes unseen in training set. Fig. 14(top) shows a bowl of candies that are colorized with less vivid colors. Besides, when the input image contains many independent instances as exampled in Fig. 14(bottom), our method tends to give less colorful results because the color anchors likely fail to cover the scene color distribution well. Fortunately, controllable colorization, as supported by our method, might be an remedial measure for such situations. Technically, we expect more advanced clustering algorithms to address this problem by choosing the number of clusters adaptively to the data. Anyway, it will be a more interesting direction to equip the model with an self-learning module for anchor location prediction. At last, the colors generated by our color generator are mainly determined by the anchor colors, which hinders the color diversity within the resultant image in some degree. This may be explained as a limitation of anchor based colorization, namely it is encouraged to build correlation across the objects that have similar color distribution. Such cases could be the second example shown in Fig. 6, where the T-shits of the two persons are colorized with the same red color. Of course, it would be a way to alleviate this problem by using more color anchors and with diverse color sampled, but at the risk of causing structure inconsistency.

(a) Input grayscale (b) Our colorization (c) Ground truth

Fig. 14. Limitation. Our method can not generate vivid colors in the case of unseen objects or a scene filled with many color-independent instances. The bottom image is from Flickr ©Oleg Klementiev.

As another future work, our method has a promising applicability to video colorization, especially in maintaining long-term temporal coherence. Since adjacent inter-frame coherence can be achieved through optical flow constraint, the long-term temporal consistence might be guaranteed by tracking [Zhou et al. 2020] the color anchors and temporally preserving their colors to avoid error accumulation.

6 CONCLUSION

We proposed a novel image colorization method that disentangles the color modality and structural consistency to achieve both aspects effectively. As far as we know, it is the first automatic colorization method that can achieve vivid colors and structural consistency at the same time. As evidenced by extensive evaluation, our method outperforms existing state-of-the-arts by a large margin, especially in visual quality and perceptual metrics. Moreover, our method demonstrates good generalization across multiple dataset, including modern color images and legacy grayscale photos. On a single trained model, our method supports diverse colorization and controllable colorization, which broadens the application scenarios. Besides, various ablation studies justified the effectiveness of our proposed techniques, which may further inspire other researches. Anyway, as the first disentangled colorization framework, we adopt a relatively straightforward solution for color anchor location, which deserves further investigation to promote the performance.

REFERENCES

Rameen Abdal, Yipeng Qin, and Peter Wonka. 2019. Image2StyleGAN: How to Embed Images Into the StyleGAN Latent Space?. In *IEEE International Conference on Computer Vision (ICCV)*.

Jason Antic. 2019. DeOldify: A open-source project for colorizing old images (and video).

Aurélie Bugeau, Vinh-Thong Ta, and Nicolas Papadakis. 2014. Variational Exemplar-Based Image Colorization. *IEEE Trans. Image Process. (TIP)* 23, 1 (2014), 298–307.

Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. 2018. COCO-Stuff: Thing and Stuff Classes in Context. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huiwen Chang, Ohad Fried, Yiming Liu, Stephen DiVerdi, and Adam Finkelstein. 2015. Palette-based photo recoloring. *ACM Trans. Graph. (TOG)* 34, 4 (2015), 139:1–139:11.

Zezhou Cheng, Qingxiong Yang, and Bin Sheng. 2015. Deep Colorization. In *IEEE International Conference on Computer Vision (ICCV)*.

Alex Yong-Sang Chia, Shaojie Zhuo, Raj Kumar Gupta, Yu-Wing Tai, Siu-Yeung Cho, Ping Tan, and Stephen Lin. 2011. Semantic colorization with internet images. *ACM Trans. Graph. (TOG)* 30, 6 (2011), 1–8.

Wonwoong Cho, Hyojin Bahng, David Keetae Park, Seungjoo Yoo, Ziming Wu, Xiaojuan Ma, and Jaegul Choo. 2018. Text2Colors: Guiding Image Colorization through Text-Driven Palette Generation. In *European Conference on Computer Vision (ECCV)*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Aditya Deshpande, Jiajun Lu, Mao-Chuang Yeh, Min Jin Chong, and David A. Forsyth. 2017. Learning Diverse Image Colorization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Aditya Deshpande, Jason Rock, and David A. Forsyth. 2015. Learning Large-Scale Automatic Image Colorization. In *IEEE International Conference on Computer Vision (ICCV)*.

Jinjin Gu, Yujun Shen, and Bolei Zhou. 2020. Image Processing Using Multi-Code GAN Prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sergio Guadarrama, Ryan Dahl, David Bieber, Jonathon Shlens, Mohammad Norouzi, and Kevin Murphy. 2017. PixColor: Pixel Recursive Colorization. In *British Machine Vision Conference 2017 (BMVC)*.

David Hasler and Sabine Süsstrunk. 2003. Measuring colorfulness in natural images. In *Human Vision and Electronic Imaging VIII*.

Mingming He, Dongdong Chen, Jing Liao, Pedro V. Sander, and Lu Yuan. 2018. Deep exemplar-based colorization. *ACM Trans. Graph. (TOG)* 37, 4 (2018), 47:1–47:16.

Mingming He, Jing Liao, Dongdong Chen, Lu Yuan, and Pedro V. Sander. 2019. Progressive Color Transfer With Dense Semantic Correspondences. *ACM Trans. Graph. (TOG)* 38, 2 (2019), 13:1–13:18.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2016. Let there be color! Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Trans. Graph. (TOG)* 35, 4 (2016), 1–11.

Revital Ironi, Daniel Cohen-Or, and Dani Lischinski. 2005. Colorization by Example. In *Eurographics Symposium on Rendering Techniques*.

Eungyeup Kim, Sanghyeon Lee, Jeonghoon Park, Somi Choi, Choonghyun Seo, and Jaegul Choo. 2021. Deep Edge-Aware Interactive Colorization against Color-Bleeding Effects. In *IEEE International Conference on Computer Vision (ICCV)*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint:1511.06349* (2014).

Manoj Kumar, Dirk Weissenborn, and Nal Kalchbrenner. 2021. Colorization Transformer. In *International Conference on Learning Representations (ICLR)*.

Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. 2016. Learning Representations for Automatic Colorization. In *European Conference on Computer Vision (ECCV)*.

Anat Levin, Dani Lischinski, and Yair Weiss. 2004. Colorization using optimization. *ACM Trans. Graph. (TOG)* 23, 3 (2004), 689–694.

Bo Li, Fuchen Zhao, Zhuo Su, Xiangguo Liang, Yu-Kun Lai, and Paul L. Rosin. 2017. Example-Based Image Colorization Using Locality Consistent Sparse Representation. *IEEE Trans. Image Process. (TIP)* 26, 11 (2017), 5188–5202.

Xuan Luo, Yanmeng Kong, Jason Lawrence, Ricardo Martin-Brualla, and Steven M. Seitz. 2020. KeystoneDepth: History in 3D. In *International Conference on 3D Vision (3DV)*. https://keystonedepth.cs.washington.edu

Safa Messaoud, David A. Forsyth, and Alexander G. Schwing. 2018. Structural Consistency and Controllability for Diverse Colorization. In *European Conference on Computer Vision (ECCV)*.

Yingge Qu, Tien-Tsin Wong, and Pheng-Ann Heng. 2006. Manga colorization. *ACM Trans. Graph. (TOG)* 25, 3 (2006), 1214–1220.

Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved Techniques for Training GANs. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*.

Jheng-Wei Su, Hung-Kuo Chu, and Jia-Bin Huang. 2020. Instance-Aware Image Colorization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Patricia Vitoria, Lara Raad, and Coloma Ballester. 2020. ChromaGAN: Adversarial Picture Colorization with Semantic Class Distribution. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*.

Tomihisa Welsh, Michael Ashikhmin, and Klaus Mueller. 2002. Transferring color to greyscale images. *ACM Trans. Graph. (TOG)* 21, 3 (2002), 277–280.

Yanze Wu, Xintao Wang, Yu Li, Honglun Zhang, Xun Zhao, and Ying Shan. 2021. Towards Vivid and Diverse Image Colorization with Generative Color Prior. In *IEEE International Conference on Computer Vision (ICCV)*.

Fengting Yang, Qian Sun, Hailin Jin, and Zihan Zhou. 2020. Superpixel Segmentation With Fully Convolutional Networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liron Yatziv and Guillermo Sapiro. 2006. Fast image and video colorization using chrominance blending. *IEEE Trans. Image Process. (TIP)* 15, 5 (2006), 1120–1129.

Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. 2019. Self-Attention Generative Adversarial Networks. In *International Conference on Machine Learning (ICML)*.

Richard Zhang, Phillip Isola, and Alexei A. Efros. 2016. Colorful Image Colorization. In *European Conference on Computer Vision (ECCV)*.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Richard Zhang, Jun-Yan Zhu, Phillip Isola, Xinyang Geng, Angela S. Lin, Tianhe Yu, and Alexei A. Efros. 2017. Real-time user-guided image colorization with learned deep priors. *ACM Trans. Graph. (TOG)* 36, 4 (2017), 119:1–119:11.

Jiaojiao Zhao, Jungong Han, Ling Shao, and Cees G. M. Snoek. 2020. Pixelated Semantic Colorization. *Int. J. Comput. Vis. (IJCV)* 128, 4 (2020), 818–834.

Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. 2020. Tracking Objects as Points. In *European Conference on Computer Vision (ECCV)*.