Coloring with Words: Guiding Image Colorization Through Text-based Palette Generation

Hyojin Bahng*[1], Seungjoo Yoo*[1], Wonwoong Cho*[1], David Keetae Park[1,3], Ziming Wu[2], Xiaojuan Ma[2], and Jaegul Choo[1,3]

[1] Korea University
{hjj552,seungjooyoo,tyflehd21,heykeetae,jchoo}@korea.ac.kr
[2] Hong Kong University of Science and Technology
zwual@connect.ust.hk, mxj@cse.ust.hk
[3] Clova AI Research, NAVER Corp.

Abstract. This paper proposes a novel approach to generate multiple color palettes that reflect the semantics of input text and then colorize a given grayscale image according to the generated color palette. In contrast to existing approaches, our model can understand rich text, whether it is a single word, a phrase, or a sentence, and generate multiple possible palettes from it. For this task, we introduce our manually curated dataset called Palette-and-Text (PAT). Our proposed model called Text2Colors consists of two conditional generative adversarial networks: the text-to-palette generation networks and the palette-based colorization networks. The former captures the semantics of the text input and produce relevant color palettes. The latter colorizes a grayscale image using the generated color palette. Our evaluation results show that people preferred our generated palettes over ground truth palettes and that our model can effectively reflect the given palette when colorizing an image.

Keywords: Color Palette Generation · Image Colorization · Conditional Generative Adversarial Networks.

1 Introduction

Humans can associate certain words with certain colors. The real question is, can machines effectively learn the relationship between color and text? Using text to express colors can allow ample room for creativity, and it would be useful to visualize the colors of a certain semantic concept. For instance, since colors can leave a strong impression on people [19], corporations often decide upon the season‘s color theme from marketing concepts such as ‘passion.’ Through text input, even people without artistic backgrounds can easily create color palettes

* These authors contributed equally.

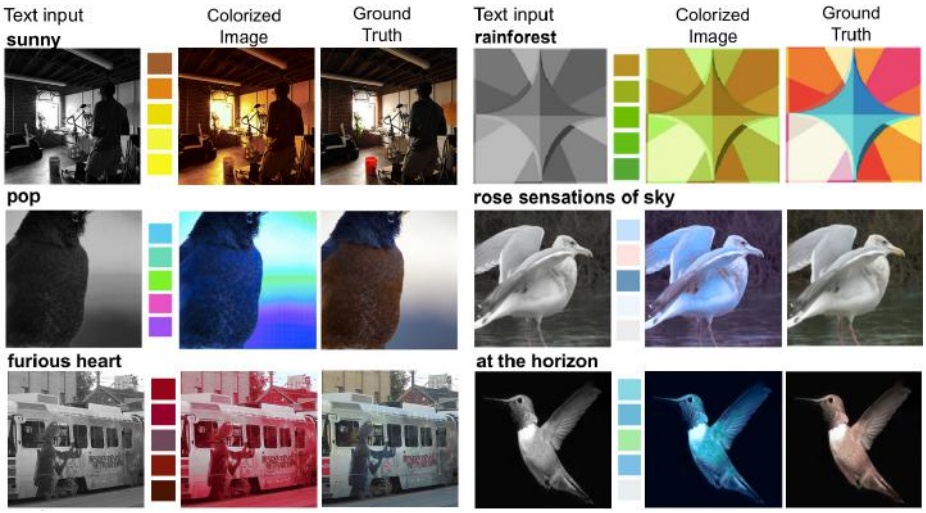

Fig. 1. Colorization results of Text2Colors given text inputs. The text input is shown above the input grayscale image, and the generated palettes are on the right of the grayscale image. The color palette is well-reflected in the colorized image when compared to the ground truth image. Our model is applicable to a wide variety of images ranging from photos to patterns (top right).

that convey high-level concepts. Since our model uses text to visualize aesthetic concepts, its range of future applications can encompass text to even speech.

Previous methods have a limited range of applications as they only take a single word as input and can recommend only a single color or a color palette in pre-existing datasets [12,8,15,25]. Other studies have further attempted to link a single word with a multi-color palette [21,36] since multi-color palettes are highly expressive in conveying semantics [18]. Compared to these previous studies, our model can generate multiple plausible color palettes when given rich text input, including both single- and multi-word descriptions, greatly increasing the boundary of creative expression through words.

In this paper, we propose a novel method to generate multiple color palettes that convey the semantics of rich text and then colorize a given grayscale image according to the generated color palette. Perception of color is inherently multimodal [4], meaning that a particular text input can be mapped to multiple possible color palettes. To incorporate such multimodality into our model, our palette generation networks are designed to generate multiple palettes from a single text input. We further apply our generated color palette to the colorization task. Motivated from previous user-guided colorizations that utilize color hints given by users [42,45], we design our colorization networks to utilize color palettes during the colorization process. Our evaluation demonstrates that the colorized outputs do not only reflect the colors in the palette but also convey the semantics of the text input.

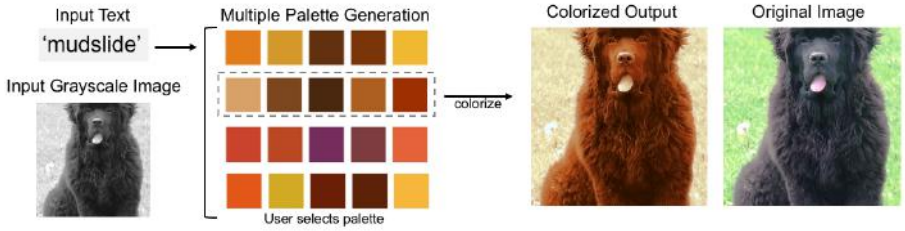

Fig. 2. How Text2Colors works. Our model can produce a diverse selection of palettes when given a text input. Users can optionally choose which palette to be applied to the final colorization output.

The contribution of this paper includes:
(1) We propose a novel deep neural network architecture that can generate multiple color palettes based on natural-language text input.
(2) Our model is able to use the generated palette to produce plausible colorizations of a grayscale image.
(3) We introduce our manually curated dataset called Palette-and-Text (PAT), which includes 10,183 pairs of a multi-word text and a multi-color palette. [4]

2 Related Work

Color Semantics Meanings associated with a color are both innate and learned [9]. For instance, red can make us instinctively feel alert [9]. Since color has a strong association with high-level semantic concepts [10], producing palettes from text input is useful in aiding artists and designers [18] and allows automatic colorization from palettes [42,5]. A downside to using text to choose a filter is that filter names do not usually convey the filter's colors [21], thus making it difficult for users to find the filter that matches their taste just by looking at filter names. To bridge this discrepancy between color palettes and their names, palette recommendation based on user text input has long been studied. Query-based methods [21,36] use text inputs to query an image from an image dictionary where colors are extracted from the queried image to make an associated palette. This method is problematic in that the text input is mapped to the image content of the queried image rather than the color that the text implies. Instead of looking for a target directly, learning-based approaches [14,27,23] match or generate color palettes to their linguistic descriptions by learning their semantic association from large-scale data. However, our model is the only generative model that supports phrase-level text input.

[4] Dataset and codes are publicly available at https://github.com/awesome-davian/Text2Colors/

Conditional GANs Conditional generative adversarial networks (cGAN) are GAN models that use conditional information for the discriminator and the generator [24]. cGANs have drawn promising results for image generation from text [32,31,43] and image-to-image translation [16,13,7]. StackGAN [43] is the first model to use conditional loss for text to image synthesis. Our model is the first to utilize the conditioning augmentation technique from StackGAN to output diverse palettes even when given the same input text.

Interactive Colorization Colorization is a multimodal task and desired colorization results for the same object may vary from person to person [4]. A number of studies introduce interactive methods that allow users to control the final colorization output [45,20]. In these models, users directly interact with the model by pinpointing where to color. Even though these methods achieve satisfactory results, a limitation is that users need to have a certain level of artistic skill. Thus instead of making the user directly color an image, other studies take a more indirect approach by utilizing color palettes to recolor an image [3,5]. Palette-based filters of our model are an effective way for non-experts to recolor an image [3].

Sequence-to-Sequence with Attention Recurrent Neural Networks (RNNs) are a popular tool due to their superior ability to learn from sequential data. RNNs are used in various tasks including sentence classification [39], text generation [37], and sequence-to-sequence prediction [38]. Incorporating attention into a sequence-to-sequence model is known to improve the model performance [22] as networks learn to selectively focus on parts of a source sentence. This allows a model to learn relations between different modalities as is done by our model (e.g., text - colors, text - action [1], and English - French [40]).

3 Palette-and-Text (PAT) Dataset

This section introduces our manually curated dataset named Palette-and-Text (PAT). PAT contains 10,183 text and five-color palette pairs, where the set of five colors in a palette is associated with its corresponding text description as shown in Figs. 3(b)-(d). Words vary with respect to their relationships with colors; some words are direct color words (e.g., pink, blue, etc.) while others evoke a particular set of colors (e.g., autumn or vibrant). To the best of our knowledge, there has been no dataset that matches a multi-word text and its corresponding 5-color palette. This dataset allows us to train our models for predicting semantically consistent color palettes with textual inputs.

Other Color Datasets Munroe's color survey [26] is a widely used large-scale color corpus. Based on crowd-sourced user judgment, it matches a text to a single color. Another dataset, Kobayashi's Color Image Scale [18], is a well-established multi-color dataset. Kobayashi only uses 180 adjectives to express

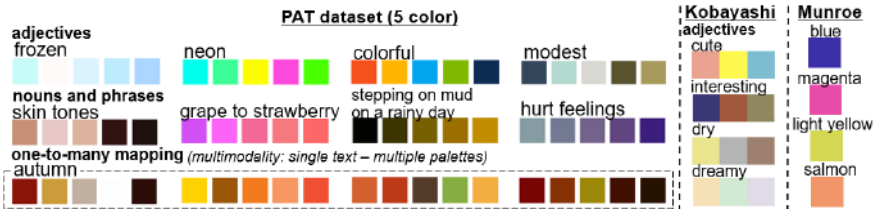

Fig. 3. Our Palette-and-Text (PAT) dataset. On the left are diverse text-palette pairs included in PAT. PAT has a very wide range of expression, especially when compared to existing datasets. Our dataset is designed to address rich text and multi-modality, where the same word can be mapped to a wide range of possible colors.

1170 three-color palettes, which greatly limits its range of expression. In contrast, our dataset is made up of 4,312 unique words. This includes much more text that was not traditionally used to express colors. Our task requires a more sophisticated dataset like PAT, that matches a text to multiple colors and is large enough for a deep learning model to learn from.

Data Collection We generated our PAT dataset by refining user-named palette data crawled from a community website called color-hex.com. Thousands of users upload custom-made color palettes on color-hex, and thus our dataset was able to incorporate a wide pool of opinions. We crawled 47,665 palette-text pairs and removed non-alphanumerical and non-English words. Among them, we found that users sometimes assign palette names in an arbitrary manner, missing their semantic consistency with their corresponding color palettes. Some names are a collection of random words (e.g., 'mehmeh' and 'i spilled tea all over my laptop rip'), or are riddled with typos (e.g., 'cause iiiiii see right through you boyyyyy' and 'greene gardn'). Thus, using unrefined raw palette names would hinder model performances significantly.

To refine the noisy raw data, four annotators voted whether the text paired with the color palette properly matches its semantic meanings. We then used only the text-palette pairs in which at least three annotators out of four agreed that semantic matching exists between the text and color palette. Including text-palette pairs in the dataset only when all four annotators agree was found to be unnecessarily strict, leaving not much room for personal subjectivity. Annotators perception is inherently subjective, meaning that a text-palette pair perfectly plausible to one person may not be agreeable to another. We wanted to incorporate such subjectivity by allowing a diverse selection of text-palette pairs. Mis-spelling and punctuation errors were manually corrected after the annotators finished sorting out the data.

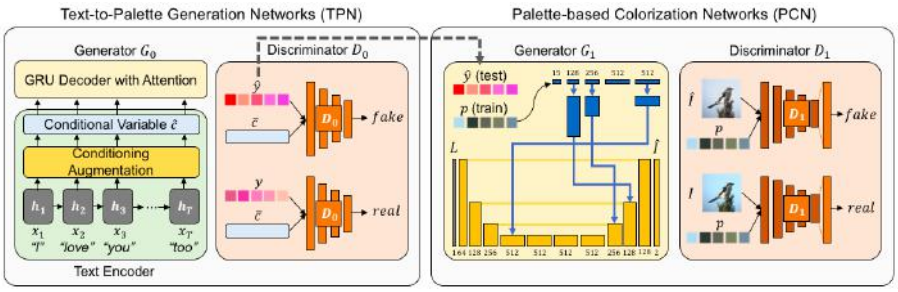

Fig. 4. Overview of our Text2Colors architecture. During training, generator G_0 learns to produce a color palette $\hat{y}$ given a set of conditional variables $\hat{c}$ processed from input text $x = \{x_1, \cdots, x_T\}$. Generator G_1 learns to predict a colorized output of a grayscale image L given a palette p extracted from the ground truth image. At test time, the trained generators G_0 and G_1 are used to produce a color palette from given text and then colorize a grayscale image reflecting the generated palette.

4 Text2Colors: Text-Driven Colorization

Text2Colors consists of two networks: Text-to-Palette Generation Networks (TPN) and Palette-based Colorization Networks (PCN). We train the first networks to generate color palettes given a multi-word text and then train the second networks to predict reasonable colorizations given a grayscale image and the generated palettes. We utilize conditional GANs (cGAN) for both networks.

4.1 Text-to-Palette Generation Networks (TPN)

Objective Function In this section, we illustrate the Text-to-Palette Generation Networks shown in Figs. 4 and 5. TPN produces reasonable color palettes associated with the text input. Let $x_i \in \mathbb{R}^{300}$ be word vectors initialized by 300-dimensional pre-trained vectors from GloVe [29]. Words not included in the pre-trained set are initialized randomly. Using the CIE *Lab* space for our task, $y \in \mathbb{R}^{15}$ represents a 15-dimensional color palette consisting of five colors with *Lab* values. After a GRU encoder encodes x into hidden states $h = \{h_1, \cdots, h_T\}$, we add random noise to the encoded representation of text by sampling latent variables $\hat{c}$ from a Gaussian distribution $\mathcal{N}(\mu(h), \Sigma(h))$. The sequence of conditioning vectors $\hat{c} = \{\hat{c}_1, \cdots, \hat{c}_T\}$ is given as *condition* for the generator to output a palette $\hat{y}$, while its mean vector $\bar{c} = \frac{1}{T}\sum_{i=1}^{T} \hat{c}$ is given as the condition for the discriminator. Our objective function of the first cGAN can be expressed as

$$L_{D_0} = \mathbb{E}_{y \sim P_{data}}[\log D_0(\bar{c}, y)] + \mathbb{E}_{x \sim P_{data}}[\log(1 - D_0(\bar{c}, \hat{y}))], \tag{1}$$

$$L_{G_0} = \mathbb{E}_{x \sim P_{data}}[\log(1 - D_0(\bar{c}, \hat{y}))], \tag{2}$$

where discriminator D_0 tries to maximize L_{D_0} against generator G_0 that tries to minimize L_{G_0}. The pre-trained word vectors x and the real color palette y is sampled from true data distribution P_{data}.

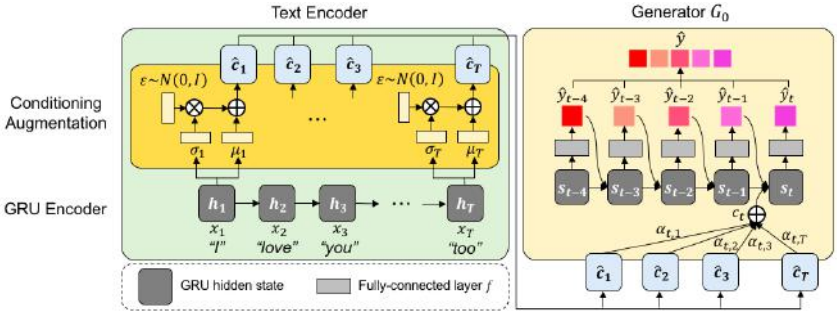

Fig. 5. Model architecture of a generator G_0 that produces the t-th color in the palette given a sequence of conditioning variables $\hat{c} = \{\hat{c}_1, \cdots, \hat{c}_T\}$ processed from an input text $x = \{x_1, \cdots, x_T\}$. Note that randomness is added to the encoded representation of text before it is passed to the generator.

Previous approaches have benefited from mixing the GAN objective with L_2 distance [28] or L_1 distance [13]. We have explored previous loss options and found the Huber (or smooth L_1) loss to be the most effective in increasing diversity among colors in generated palettes. The Huber loss is given by

$$L_H(\hat{y}, y) = \begin{cases} \frac{1}{2}(\hat{y} - y)^2 & \text{for } |\hat{y} - y| \leq \delta \\ \delta\,|\hat{y} - y| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases} \tag{3}$$

This loss term is added to the generator's objective function to force the generated palette to be close to the ground truth palette. We also adopted the Kullback-Leibler (KL) divergence regularization term [43], i.e.,

$$D_{KL}(\mathcal{N}(\mu(h), \Sigma(h)) \parallel \mathcal{N}(0, I)), \tag{4}$$

which is added to the generator's objective function to further enforce the smoothness over the conditioning manifold. Our final objective function is

$$L_{D_0} = \mathbb{E}_{y \sim P_{data}}[\log D_0(\bar{c}, y)] + \mathbb{E}_{x \sim P_{data}}[\log(1 - D_0(\bar{c}, \hat{y}))], \tag{5}$$

$$\begin{aligned} L_{G_0} = \mathbb{E}_{x \sim P_{data}}[\log(1 - D_0(\bar{c}, \hat{y}))] + \lambda_H L_H(\hat{y}, y) \\ + \lambda_{KL} D_{KL}(\mathcal{N}(\mu(h), \Sigma(h)) \parallel \mathcal{N}(0, I)), \end{aligned} \tag{6}$$

λ_H and λ_{KL} are the hyperparameters to balance the three terms in Eq. 6. We set $\delta = 1, \lambda_H = 100, \lambda_{KL} = 0.5$ in our model.

Networks Architecture

Encoding Text through Conditioning Augmentation. Learning a mapping from text to color is inherently multimodal. For instance, a text 'autumn' can be mapped to a variety of plausible color palettes. As text becomes longer, such

as 'midsummer to autumn' or 'autumn breeze and falling leaves', the scope of possible matching palettes becomes more broad and diverse. To appropriately model the multimodality of our problem, we utilize the conditioning augmentation (CA) [43] technique. Rather than using the fixed sequence of encoded text as input to our generator, we randomly sample latent vector $\hat{c}$ from a Gaussian distribution $\mathcal{N}(\mu(h), \Sigma(h))$ as shown in Fig. 5. This randomness allows our model to generate multiple plausible palettes given same text input.

To obtain the conditioning variable $\hat{c} = \{\hat{c}_1, \cdots, \hat{c}_T\}$, the pre-trained word vectors $x = \{x_1, \cdots, x_T\}$ are first fed into a GRU encoder to compute hidden states $h = \{h_1, \cdots, h_T\}$. This text representation is fed into a fully-connected layer to generate μ and σ (the values in the diagonal of Σ) for the Gaussian distribution $\mathcal{N}(\mu(h), \Sigma(h))$. Conditioning variable $\hat{c}$ is computed by $\hat{c} = \mu + \sigma \odot \epsilon$, where $\odot$ is the element-wise multiplication and $\epsilon \sim \mathcal{N}(0, I)$. The resulting set of vectors $\hat{c} = \{\hat{c_1}, \cdots, \hat{c_T}\}$ will be used as *condition* for our generator.

Generator. We design our generator G_0 as a variant of a GRU decoder with attention mechanism [22,2,6]. The i-th color of the palette $\hat{y}_i$ is computed as

$$\hat{y}_i = f(s_i) \text{ where } s_i = g(\hat{y}_{i-1}, c_i, s_{i-1}). \tag{7}$$

s_i is a GRU hidden state vector for time i, having the previously generated color $\hat{y}_{i-1}$, the context vector c_i, and the previous hidden state s_{i-1} as input. The GRU hidden state s_i is given as input to a fully-connected layer f to output the i-th color of the palette $\hat{y}_i \in \mathbb{R}^3$. The resulting five colors are combined to produce a single palette output $\hat{y}$.

The context vector c_i depends on a sequence of conditioning vectors $\hat{c} = \{\hat{c_1}, \cdots, \hat{c_T}\}$ and the previous hidden state s_{i-1}. The context vector c_i is computed as the weighted sum of these conditions $\hat{c_i}$'s, i.e.,

$$c_i = \sum_{j=1}^{T} \alpha_{ij} \hat{c}_j. \tag{8}$$

The weight α_{ij} of each conditional variable $\hat{c}_j$ is computed by

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T} \exp(e_{ik})} \text{ where } e_{ij} = a\left(s_{i-1}, \hat{c}_j\right). \tag{9}$$

$$a\left(s_{i-1}, \hat{c}_j\right) = w^T \sigma(W_s s_{i-1} + W_{\hat{c}} \hat{c}_j), \tag{10}$$

where $\sigma(\cdot)$ is a sigmoid activation function and w is a weight vector. The additive attention [2] $a\left(s_{i-1}, \hat{c}_j\right)$ computes how well the j-th word of the text input matches the i-th color of the palette output. The score α_{ij} is computed based on the GRU hidden state s_{i-1} and the j-th condition $\hat{c}_j$. The attention mechanism enables the model to effectively map complex text input to the palette output.

Discriminator. For the discriminator D_0, the conditioning variable $\bar{c}$ and the color palette are concatenated and fed into a series of fully-connected layers. By jointly learning features across the encoded text and palette, the discriminator classifies whether the palettes are real or fake.

4.2 Palette-based Colorization Networks (PCN)

Objective Function The goal of the second networks is to automatically produce colorizations of a grayscale image guided by the color palette as a conditioning variable. The inputs are a grayscale image $L \in \mathbb{R}^{H \times W \times 1}$ representing the lightness in CIE *Lab* space and a color palette $p \in \mathbb{R}^{15}$ consisting of five colors in *Lab* values. The output $\hat{I} \in \mathbb{R}^{H \times W \times 2}$ corresponds to the predicted *ab* color channels of the image. The objective function of the second model can be expressed as

$$L_{D_1} = \mathbb{E}_{I \sim P_{data}}[\log D_1(p, I)] + \mathbb{E}_{\hat{I} \sim P_{G_1}}[\log(1 - D_1(p, \hat{I}))], \tag{11}$$

$$L_{G_1} = \mathbb{E}_{\hat{I} \sim P_{G_1}}[\log(1 - D_1(p, \hat{I}))] + \lambda_H L_H(\hat{I}, I). \tag{12}$$

D_1 and G_1 included in the equation are shown in Fig.4. We have also added the Huber loss to the generator's objective function. In other words, the generator learns to be close to the ground truth image with *plausible* colorizations, while incorporating palette colors to the output image to fool the discriminator. We set $\lambda_H = 10$ in our model.

Networks Architecture

Generator. The generator consists of two sub-networks: the main colorization networks and the conditioning networks. Our main colorization networks adopts the U-Net architecture [33], which has shown promising results in colorization tasks [13,45]. The skip connections help recover spatial information [33], as the input and the output images share the location of prominent edges [13].

The role of the conditioning networks is to apply the palette colors to the generated image. During training, the networks are given a palette $p \in \mathbb{R}^{15}$ extracted from the ground truth image I. We utilize the Color Thief [5] function to extract a palette consisting of five dominant colors of the ground truth image. Similar to the previous work [45], the conditioning palette p is fed into a series of 1×1 *conv-relu* layers as shown in Fig. 4. The feature maps in layers 1, 2, and 4 are duplicated spatially to match the spatial dimension of the *conv*9, *conv*8, and *conv*4 features in the main colorization networks and added in an element-wise manner. The palette p is fed into upsampling layers with skip connections as well as the middle of the main networks. This allows the generator to detect prominent edges and apply palette colors to suitable locations of the image. During test time, we use the generated palette $\hat{y}$ from the first networks (TPN) as the conditioning variable, colorizing the grayscale image with the predicted palette colors.

Discriminator. As our discriminator D_1, we use a variant of the DCGAN architecture [30]. The image and conditioning variable p are concatenated and fed into a series of *conv-leaky relu* layers to jointly learn features across the image and the palette. Afterwards, it is fed into a fully-connected layer to classify whether the image is real or fake.

[5] `http://lokeshdhakar.com/projects/color-thief/`

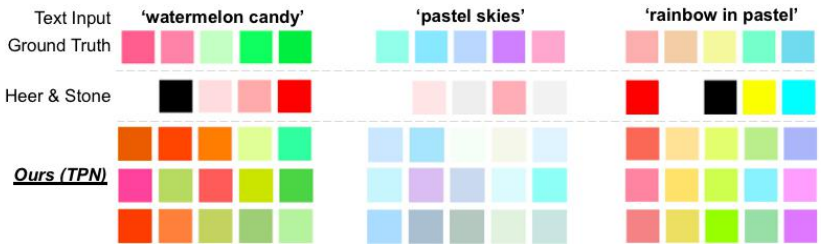

Fig. 6. Comparison to baselines and qualitative analysis on multimodality: Our TPN generates appealing color palettes that reflect all details of the text input. Also our model can generate multiple palettes with the same text input(three rows from bottom). In comparison, Heer and Stone [12]'s model frequently generates unrelated colors and has deterministic outputs.

4.3 Implementation Details

We first train D_0 and G_0 of TPN for 500 epochs using the PAT dataset. We then train D_1 and G_1 of the PCN for 100 epochs, using the extracted palette from a ground truth image. Finally, we use the trained generators G_0 and G_1 during test time to colorize a grayscale image with generated palette $\hat{y}$ from a text input x. All networks are trained using Adam optimizer [17] with a learning rate of 0.0002. Weights were initialized from a Gaussian distribution with zero mean and standard deviation of 0.05. We set other hyper parameters as $\delta = 1, \lambda_H = 100$, and $\lambda_{KL} = 0.5$.

5 Experimental Results

This section presents both quantitative and qualitative analyses of our proposed model. We evaluate the TPN (Section 4.1) based on our PAT dataset. For the training of the PCN (Section 4.2), we use two different datasets, CUB-200-2011 (CUB) [41] and ImageNet ILSVRC Object Detection (ImageNet dataset) [34].

5.1 Analysis on Multimodality and Diversity of Generated Palettes

This section discusses the evaluation on multimodality and diversity of our generated palettes. Multimodality refers to how many different color palettes a single text input can be mapped to. In other words, if a single text can be expressed with more color palettes, the more multimodal it is. As shown in Fig. 6, our model is multimodal, while previous approaches are deterministic, meaning that it generates only a particular color palette when given a text input. Diversity within a palette refers to how diverse the colors included in a single palette are. Following the current standard for perceptual color distance measurement, we use the CIEDE2000 [35] on CIE *Lab* space to compute a model's multimodality and diversity. To measure multimodality, we compute the average minimum

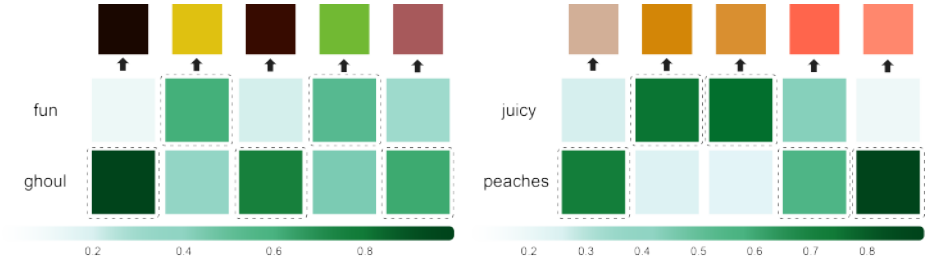

Fig. 7. Attention analysis. Attention scores measured by the TPN for two text input samples. Each box color (in green) denotes the attention score computed in producing the corresponding color shown on top. The dashed-line boxes indicate the word that each color output attended to.

distances between colors from different palettes. To measure diversity of a color palette, we measure the average pairwise distance between the five colors within a palette. All measurements are computed based on the test dataset.

Results. Table 1 shows the multimodality and diversity measurement among the variants of our model. The CA module (Section 4.1) enables our networks to suggest multiple color palettes when given the same text input. The model variant without CA (the first row in Table 1) results in zero multimodality, indicating that the networks generate identical palettes for the same text input. Another palette generation model by Heer and Stone [12] also has zero multimodality. This shows that TPN is the only existing model that can adequately express multimodality, which is crucial in the domain of colors. Although Heer and Stone's model has higher diversity than TPN, Fig. 6 shows that their palettes contain irrelevant colors that may increase diversity but decrease palette quality. On the other hand, TPN creates those palettes containing colors that well match each other. Results on the fooling rate will be further illustrated in Section 5.3.

5.2 Analysis on Attention Outputs

The attention module (Section 4.1) plays a role of attending to particular words in text input to predict the most suitable colors for the text input. Fig. 7 illustrates how the predicted colors are influenced by attention scores. The green-colored boxes show attention scores computed for each word token when predicting each corresponding color in the palette. Higher scores are indicated by dashed-line boxes. We observe that three colors generated by attending to *ghoul* are all dark and gloomy, while the other two colors attending to *fun* are bright. This attention mechanism enables our model to thoroughly reflect the semantics included in text inputs of varying lengths.

Fig. 8. Qualitative analysis on semantic context. Our model reflects subtle nuance differences in the semantic context of a given text input in the color palette outputs. Except for the first column, all the text combinations shown here are unseen data.

Table 1. Quantitative analysis results

		Palette Evaluation				User Study: Part I			
Model Variations		**Diversity**		**Multimodality**		**Fooling Rate (%)**			
Objective Function	CA	Mean	Std	Mean	Std	Mean	Std	Max	Min
Ours (TPN)	X	19.36	8.74	0.0	0.0	-	-	-	-
Ours (TPN)	O	20.82	7.43	5.43	8.11	**56.2**	12.7	**76.7**	37.1
Heer and Stone	-	35.92	12.66	0.0	0.0	39.6	10.8	58.2	25.8
Ground truth palette	-	32.60	21.84	-	-	-	-	-	-

5.3 User Study

We conduct a user study to reflect universal user opinions on the outputs of our model. Our user study is composed of two parts. The first part measures how the generated palettes match the text inputs. The second part is a survey that compares the performance of our palette-based colorization model to another state-of-the-art colorization model. 53 participants took part in our study.

Part I: Matching between Text and Generated Palettes Our goal is to generate a palette with a strong semantic connection with the given text input. A natural way to evaluate it is to quantify the degree of connection between the text input and the generated palette, in comparison to the same text input and its ground truth palette. Given a text input, its generated palette, and the ground truth palette, we ask human observers to select the palette that best suits the text input. A fooling rate (FR) in this study indicates the relative number of generated palettes chosen over ground truth palettes. More people choosing the generated palette results in a higher FR. This measure has often been used to assess the quality of colorization results [45,11]. We will use this metric to measure how much a text input matches its generated palette.

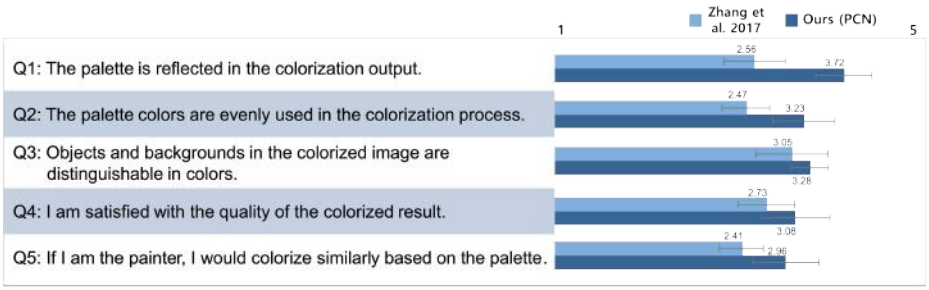

Fig. 9. Colorization performance comparisons. Mean and standard deviation values for each question are reported for the baseline [45] and our PCN. Our PCN scores higher on all of the questions, showing that users are more satisfied with PCN.

Study Procedure. Users participate in the user study over TPN and Heer and Stone's model [12]. Each consists of 30 evaluations. We randomly choose a single data item out of 992 test data and show the text input along with the generated palette and the ground truth palette.

Results. In Table 1, we measure the FR score for each person and compute the mean and the standard deviation (std) of all of the scores from participants. Max and min scores represent the highest and the lowest FR scores, respectively, recorded by a single person. While Heer and Stone's model [12] shows low FR of 39.6%, our TPN has the FR of 56.2% while maintaining a high level of diversity and multimodality. The FR of 56.2% indicates that the generated palettes are indistinguishable to human eyes and sometimes even match the input text better than the ground truth palettes. Note that the standard deviation of 12.7% implies diverse responses to the same data pairs.

Part II: Colorization Comparisons In this part of the user study, we conduct a survey on the performance of the PCN given palette inputs. Users are asked to answer five questions based on the given grayscale image, the color palette, and the colored image. For quantitative comparison, we set a state-of-the-art colorization model [45] as our baseline. This model originally contains local and global hint networks. In our implementation of the baseline model, we utilize the global hint networks to infuse our generated palette to the main colorization networks. Note that we modified the baseline model to fit our task. Our novelty is the ability to produce high-quality colorization with only five colors of a palette while our baseline [45] needs 313 bins of *ab* gamut. Our model is able to colorize with limited information due to novel components such as the conditional adversarial loss and feeding the palette into skip-connection layers.

Study Procedure. We show colorization results of our PCN and the baseline model one-by-one in a random order. Then, we ask each participant to answer

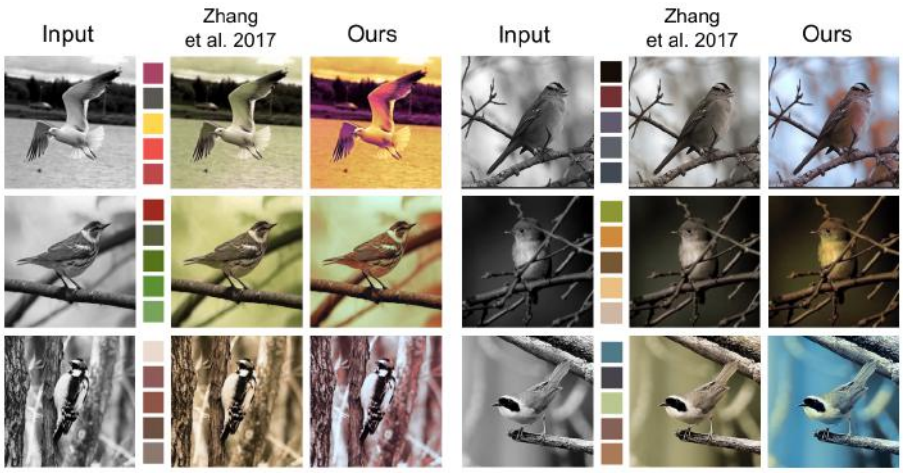

Fig. 10. We compare colorization results with previous work [45]. The five-color palette used for colorization is shown next to the input grayscale image. Note that our PCN performs better at applying various colors included in the palette.

five different questions (shown in Fig. 9) based on a five-point Likert scale. The focus of our questions is to evaluate how well the palette was used in colorizing the given grayscale image. The total number of data samples per test is 15.

Results. The resulting statistics are reported in Fig. 9. Our PCN achieves higher scores than the baseline model across all the questions. We can infer that the palettes generated by our model are preferred over palettes created by a human hand. Since our model learns consistent patterns from a large number of human-generated palette-text pairs, our model may have generated color palettes that more users could relate to.

6 Conclusions

We proposed a generative model that can produce multiple palettes from rich text input and colorize grayscale images using the generated palettes. Evaluation results confirm that our TPN can generate plausible color palettes from text input and can incorporate the multimodal nature of colors. Qualitative results on our PCN also show that the diverse colors in a palette are effectively reflected in the colorization results. Future work includes extending our model to a broader range of tasks requiring color recommendation and conducting the detailed analysis of our dataset.

Acknowledgement. This work was partially supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (No. NRF2016R1C1B2015924). Jaegul Choo is the corresponding author.

References

1. Ahn, H., Ha, T., Choi, Y., Yoo, H., Oh, S.: Text2Action: Generative adversarial synthesis from language to action. In: Proc. the IEEE International Conference on Robotics and Automation (ICRA) (2018)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: Proc. the International Conference on Learning Representations (ICLR) (2014)
3. Chang, H., Fried, O., Liu, Y., DiVerdi, S., Finkelstein, A.: Palette-based photo recoloring. ACM Transactions on Graphics (TOG) **34**(4) (2015)
4. Charpiat, G., Hofmann, M., Schölkopf, B.: Automatic image colorization via multimodal predictions. In: Proc. the European Conference on Computer Vision (ECCV) (2008)
5. Cho, J., Yun, S., Lee, K., Choi, J.Y.: PaletteNet: Image recolorization with given color palette. In: Proc. the IEEE Conference on Computer Vision and Pattern Recognition Workshops (2017)
6. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014)
7. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proc. the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
8. Chuang, J., Stone, M., Hanrahan, P.: A probabilistic model of the categorical association between colors. In: Proc. the IS&T Color and Imaging Conference (CIC). vol. 2008 (2008)
9. Crozier, W.: The psychology of colour preferences. Coloration Technology **26**(1) (1996)
10. De Bortoli, M., Maroto, J.: Colours across cultures: Translating colours in interactive marketing communications. In: Proc. the European Languages and the Implementation of Communication and Information Technologies (ELICIT) (2001)
11. Guadarrama, S., Dahl, R., Bieber, D., Norouzi, M., Shlens, J., Murphy, K.: Pixcolor: Pixel recursive colorization. In: Proc. the British Machine Vision Conference (BMVC) (2017)
12. Heer, J., Stone, M.: Color naming models for color selection, image editing and palette design. In: Proc. the SIGCHI Conference on Human Factors in Computing Systems (SIGCHI) (2012)
13. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proc. the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
14. Jahanian, A., Keshvari, S., Vishwanathan, S., Allebach, J.P.: Colors–messengers of concepts: Visual design mining for learning color semantics. ACM Transactions on Computer-Human Interaction (TOCHI) **24**(1) (2017)
15. Kawakami, K., Dyer, C., Routledge, B.R., Smith, N.A.: Character sequence models for colorful words. In: Proc. the Conference on Empirical Methods in Natural Language Processing (EMNLP) (2016)
16. Kim, T., Cha, M., Kim, H., Lee, J., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: Proc. the International Conference on Machine Learning (ICML) (2017)

17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Proc. the International Conference on Learning Representations (ICLR) (2014)
18. Kobayashi, S.: Color image scale. http://www.ncd-ri.co.jp/english/main_0104.html (2009)
19. Labrecque, L.I., Milne, G.R.: Exciting red and competent blue: the importance of color in marketing. Journal of the Academy of Marketing Science **40**(5) (2012)
20. Li, X., Zhao, H., Nie, G., Huang, H.: Image recoloring using geodesic distance based color harmonization. Computational Visual Media **1**(2) (2015)
21. Liu, Y., Cohen, M., Uyttendaele, M., Rusinkiewicz, S.: Autostyle: Automatic style transfer from image collections to users' images. Computer Graphics Forum (CGF) **33**(4) (2014)
22. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proc. the Conference on Empirical Methods in Natural Language Processing (EMNLP) (2015)
23. McMahan, B., Stone, M.: A bayesian model of grounded color semantics. Transactions of the Association of Computational Linguistics (TACL) **3**(1) (2015)
24. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)
25. Monroe, W., Hawkins, R.X., Goodman, N.D., Potts, C.: Colors in context: A pragmatic neural model for grounded language understanding. Transactions of the Association of Computational Linguistics (ACL) (2017)
26. Munroe, R.: Color survey results. Online at http://blog.xkcd.com/2010/05/03/color-surveyresults (2010)
27. Murray, N., Skaff, S., Marchesotti, L., Perronnin, F.: Toward automatic and flexible concept transfer. Computers & Graphics **36**(6) (2012)
28. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: Proc. the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
29. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proc. the Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014)
30. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: Proc. the International Conference on Learning Representations (ICLR) (2015)
31. Reed, S., Akata, Z., Lee, H., Schiele, B.: Learning deep representations of fine-grained visual descriptions. In: Proc. the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
32. Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. In: Proc. the International Conference on Machine Learning (ICML) (2016)
33. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Proc. the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI) (2015)
34. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision (IJCV) **115**(3) (2015)
35. Sharma, G., Wu, W., Dalal, E.N.: The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. Color Research & Application **30**(1) (2005)
36. Solli, M., Lenz, R.: Color semantics for image indexing. In: Proc. the Conference on Colour in Graphics Imaging and Vision (CGIV) (2010)

37. Sutskever, I., Martens, J., Hinton, G.E.: Generating text with recurrent neural networks. In: Proc. the International Conference on Machine Learning (ICML) (2011)
38. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems (NIPS) (2014)
39. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: Proc. the Conference on Empirical Methods in Natural Language Processing (EMNLP) (2015)
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems (NIPS) (2017)
41. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)
42. Xiao, Y., Zhou, P., Zheng, Y.: Interactive deep colorization with simultaneous global and local inputs. arXiv preprint arXiv:1801.09083 (2018)
43. Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X., Metaxas, D.: Stack-GAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proc. the IEEE International Conference on Computer Vision (ICCV) (2017)
44. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: Proc. the European Conference on Computer Vision (ECCV) (2016)
45. Zhang, R., Zhu, J.Y., Isola, P., Geng, X., Lin, A.S., Yu, T., Efros, A.A.: Real-time user-guided image colorization with learned deep priors. ACM Transactions on Graphics (TOG) (2017)

Supplementary Materials

7 User Study Samples

Our user study consists of two parts, one for evaluation of Text-to-palette Generation Networks (TPN) and the other for evaluation of Palette-based Colorization Networks (PCN). Fig. 11**(a)-(b)** illustrates how our data tuples were shown to the participants in **Part I** and **Part II**, respectively.

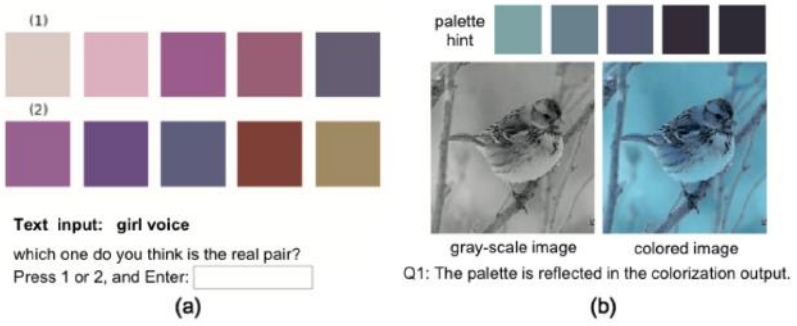

Fig. 11. UI design of our user study.

8 Text-to-Palette Generation Networks (TPN)

8.1 Model Comparisons for Learning Global Color Distributions

Fig. 12 shows comparisons of color distributions between ground truth palettes of the training data and generated palettes from our test data. For each color distribution, we quantize the ab values of every palette color into 313 color bins [44] and visualize the probability distribution of ab values. We compare three model variants of different objective functions: cGAN+Huber (λ_H=100), Huber (λ_H=100), and cGAN (λ_H=0). We also compute the Kullback-Leibler (KL) divergence between the ground truth palette distribution of the training data and that of our model variants.

As shown in the bottommost plot of Fig. 12, the Huber loss plays a critical role in producing proper colors close to the ground truth image. Without the Huber loss, the model does not only fail to recover the color distribution similar to the ground truth data but also exhibits the lowest fooling rate of 30.7% in user study results. On the other hand, the model with cGAN+Huber loss (λ_H=100) records the lowest KL divergence of 0.2299 as well as the best fooling rate of 56.2%, while the model with only the Huber loss (λ_H=100) records the second best. This is due to the fact that only using the Huber loss leads to blindly averaging over multiple ground truth palettes, resulting in slightly desaturated

palette results as shown in the second row of Fig. 13. In contrast, the model with both cGAN+Huber loss learns and preserves various ground truth colors rather than simply averaging them, resulting in bright, highly saturated results as shown in the first row of Fig. 13.

Fig. 12. Color Distribution Comparisons. Red lines correspond to color distributions of generated palettes from three model variants. The blue lines denote the ground truth color distribution of the training data. The KL divergence of the three distribution pairs are computed as 0.2299, 0.2375, and 0.5833 in order.

8.2 Additional Results

This section shows additional, diverse and detailed results from TPN.

Fig. 14 shows how our model handles phrase-level inputs. To make comparison easier, all the phrases contains the word 'love.' It is interesting to see how our model chooses to express the subtle nuance differences included in the input text. Notice how the output color palettes tend to be darker for text inputs that are negative towards 'love' (e.g., 'i thought i loved you' and 'where did our love go'). All input phrases included in this figure are unseen data.

Fig. 15 shows outputs of our model in comparison to ground truth palettes. If an input word is seen at least once in the training data, our model is able to output a color palette related to the input word. For instance, take a look at the color palette named 'mango and grapefruit' on the top left. The word

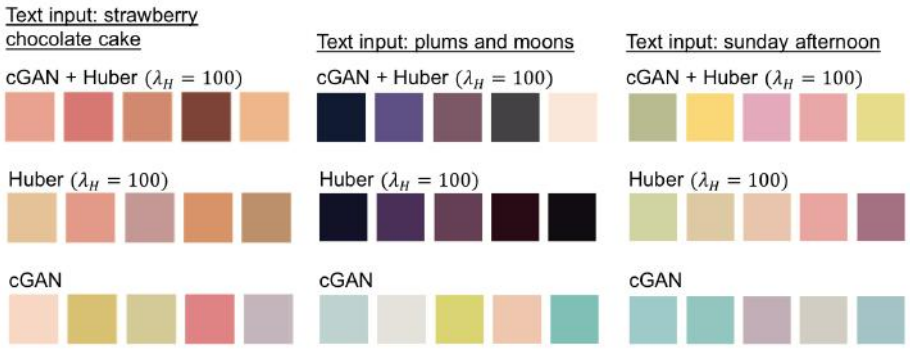

Fig. 13. Comparison of palette prediction results from different model variations.

Fig. 14. Handling phrase-level inputs about 'love'.

Fig. 15. Palette predictions from test set.

'grapefruit' is included only once in the training set. Yet, the model successfully outputs a color palette that matches the text input. Also, ground truth palettes are included for a direct comparison with generated palettes. Even if the predicted palette is not exactly identical to the ground truth palette, both can be perceived as reasonable colors. Even though our model can effectively pro-

Fig. 16. Failed results of TPN. Our model fails and outputs the same washed-out grayish-brown color palettes for unknown tokens.

duce semantically meaningful colorizations, it struggles when unknown tokens are given as input. Unknown tokens refer to words not included in the training set. it is not surprising that our model fails and outputs the same washed-out grayish-brown palettes as we can see in Fig. 16. On the other hand, our model can still produce reasonable palettes in the case of unseen, new combinations of words found in the training set. For example, 'bright life' in Fig. 15 was seen separately as 'bright' and 'life' in the training set but not together. Thus, 'bright life' is classified as unseen data, which our model has no problem in predicting color palettes from.

9 Palette-Based Colorization Networks (PCN)

We present additional colorization results on datasets including CUB-200-2011 (CUB dataset) [41], ImageNet ILSVRC Object Detection (ImageNet dataset) [34], and Graphical Pattern images (Pattern images) in Figs. 17-22. In these figures, the leftmost columns are grayscale images. Text inputs are given above the grayscale image. The vertical color palettes next to the grayscale images are palettes generated from the text input. The output has been colorized with the generated color palette. We would like to emphasize that our model effectively utilizes the generated color palettes during the colorization process. The colorized image may be different from its natural colors because our networks incorporate additional color hints. We display the original ground truth image on the right to compare how different an image becomes after applying the palettes.

9.1 CUB-200-2011

Figs. 17 and 18 show additional colorization results on the CUB dataset.

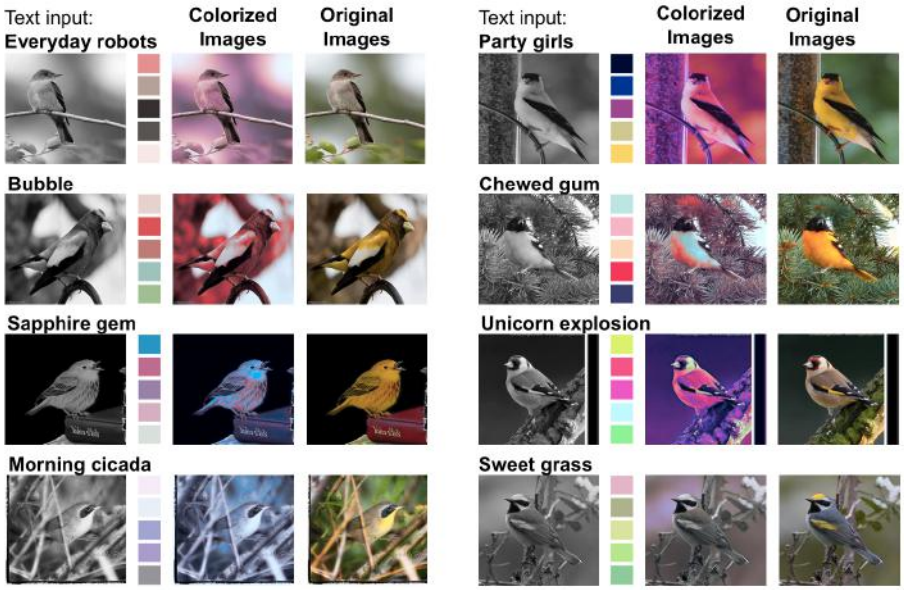

Fig. 17. Results on CUB dataset (1).

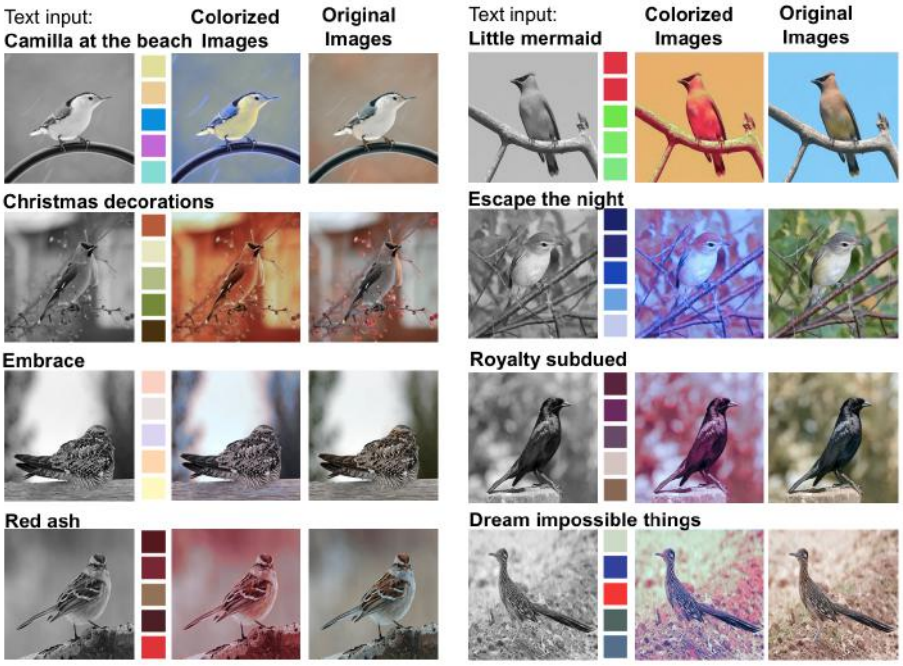

Fig. 18. Results on CUB dataset (2).

9.2 ImageNet ILSVRC Object Detection

Figs. 19 and 20 show additional colorization results on the ImageNet dataset.

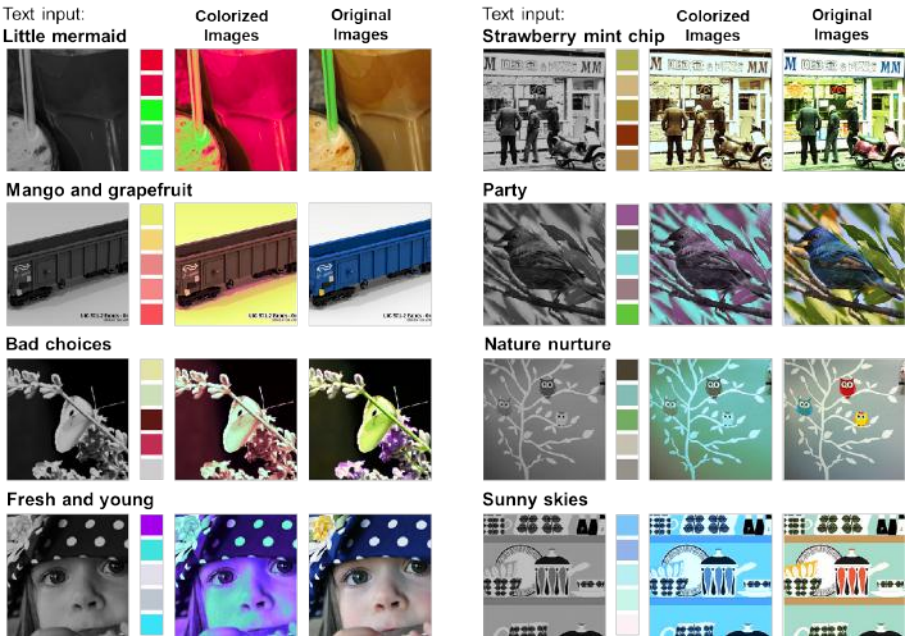

Fig. 19. Results on ImageNet dataset (1).

9.3 Graphical Pattern Images

Our PCN model generalizes surprisingly well on other types of images. Our model is trained on ImageNet dataset, which is mostly made up of natural images. Instead of natural images, we used our colorization model to colorize graphical pattern images. The graphical pattern images are crawled from Google through searching keywords such as 'pattern,' 'fabric pattern,' or 'beautiful patterns.' As seen in Figs. 21 and 22, graphical pattern images are significantly different from natural images. The colorized outputs show that our model can apply our generated color palettes on images of diverse shapes and textures. The results qualitatively show that our palette-based colorization model is transferable to other image domains.

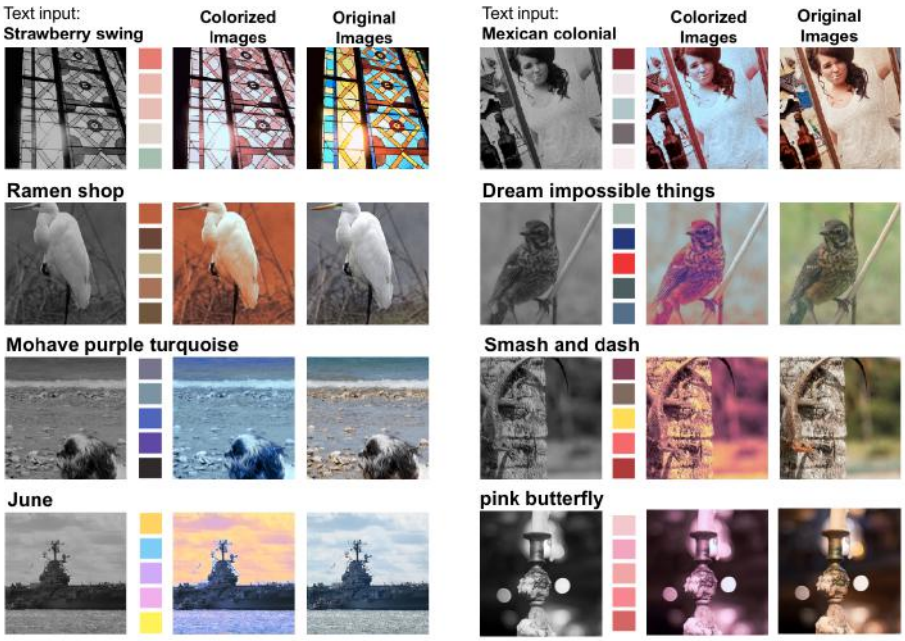

Fig. 20. Results on ImageNet dataset (2).

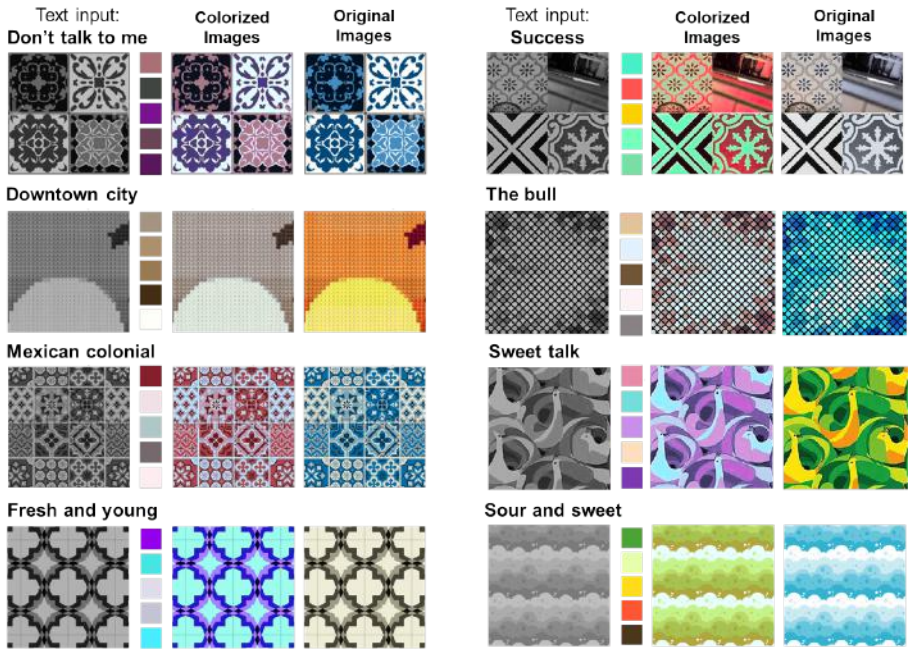

Fig. 21. Results on graphical pattern images (1).

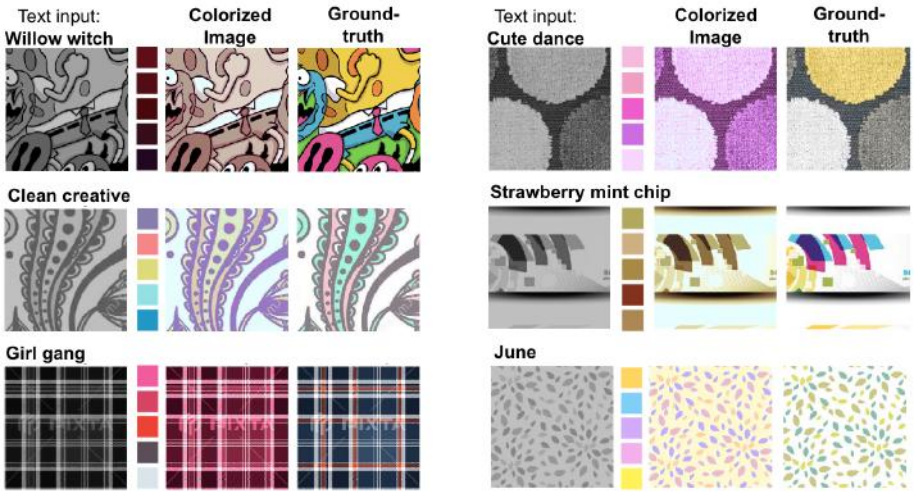

Fig. 22. Results on graphical pattern images (2).