L-CoDe: Language-based Colorization using Color-object Decoupled Conditions

Shuchen Weng[1#], Hao Wu[2#], Zheng Chang[3], Jiajun Tang[1], Si Li[3*], Boxin Shi[1, 4, 5, 6]

[1] School of Computer Science, Peking University [2] School of Software and Microelectronics, Peking University
[3] School of Artificial Intelligence, Beijing University of Posts and Telecommunications
[4] Institute for Artificial Intelligence, Peking University [5] Beijing Academy of Artificial Intelligence
[6] Peng Cheng Laboratory
{shuchenweng, haowu, jiajun.tang, shiboxin}@pku.edu.cn, {zhengchang98, lisi}@bupt.edu.cn

Abstract

Colorizing a grayscale image is inherently an ill-posed problem with multi-modal uncertainty. Language-based colorization offers a natural way of interaction to reduce such uncertainty via a user-provided caption. However, the color-object coupling and mismatch issues make the mapping from word to color difficult. In this paper, we propose L-CoDe, a Language-based Colorization network using color-object Decoupled conditions. A predictor for object-color corresponding matrix (OCCM) and a novel attention transfer module (ATM) are introduced to solve the color-object coupling problem. To deal with color-object mismatch that results in incorrect color-object correspondence, we adopt a soft-gated injection module (SIM). We further present a new dataset containing annotated color-object pairs to provide supervisory signals for resolving the coupling problem. Experimental results show that our approach outperforms state-of-the-art methods conditioned on captions.

1 Introduction

Image colorization, as the name implies, aims to add colors to a grayscale image, since there are various types of pictures that do not contain color, such as legacy photos, near-infrared images, sketch or manga, and so on. The colorization process could be fully automatic by predicting colors from large-scale data using data-driven approaches (Isola et al. 2017; Zhao et al. 2018; Su, Chu, and Huang 2020). However, the problem is inherently ambiguous since there are potentially infinite many colors that can be assigned to the gray pixels of an input image (e.g., an apple may be colorized by green, yellow, or red). Hence, human intervention often plays an important role to determine a unique solution.

With user interaction, scribble-based colorization methods (Luan et al. 2007; Zhang et al. 2017; Sangkloy et al. 2017) focus on propagating local user scribbles (e.g., color points or strokes), while example-based colorization methods (Gupta et al. 2012; He et al. 2018; Xu et al. 2020; Lu et al. 2020) colorize the input grayscale image with color statistics transferred from a similar reference image. However, these methods often require a good sense of aesthetics or a suitable reference image, which could be time-consuming for an untrained user. Language-based colorization, a new interactive approach appearing in the recent years (Manjunatha et al. 2018; Chen et al. 2018; Xie 2018), colorizes a grayscale image conditioned on a caption. Users only need to describe the object[1] category desired to be colorized and the corresponding color of that region in the form of natural language.

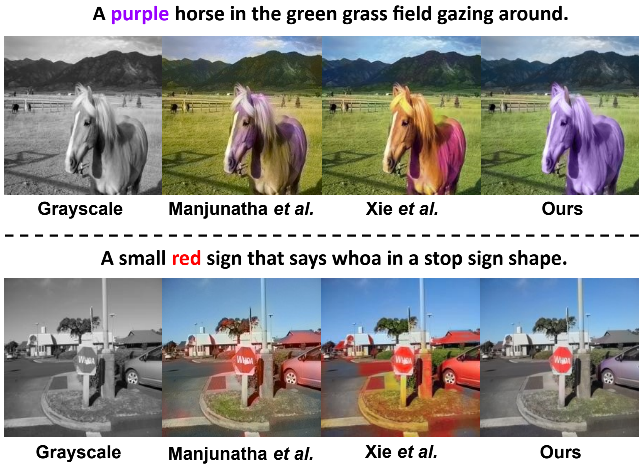

A purple horse in the green grass field gazing around.
Grayscale
Manjunatha et al.
Xie et al.
Ours
A small red sign that says whoa in a stop sign shape.
Grayscale
Manjunatha et al.
Xie et al.
Ours

Figure 1: L-CoDe conducts colorization on a grayscale image according to a given caption displayed on top of each image row. Due to challenges caused by color-object coupling (top row) and color-object mismatch (bottom row), state-of-the-art methods (Manjunatha et al. 2018; Xie 2018) fail to add the designated color according to the caption correctly, while L-CoDe succeeds.

For language-based colorization, the colorized result should be consistent with the description of the caption. To meet this requirement, a caption is usually encoded into a vector and injected into a colorization network as a condition (Manjunatha et al. 2018; Xie 2018), which makes it more difficult than automatic colorization. There are two problems with such an approach: (1) *Color-object coupling*: When the color and object combination specified in the cap-

[#] Equal contributions. * Corresponding author.

[1] In this paper, with a bit of abuse of concept, we refer to thing/instance class (e.g., person, car, elephant) and stuff class (e.g., grass, wall, sky) together as “object”.

tion is less observed in the dataset ("purple horse" in the top example of Figure 1), the colorization method may fail to change the "common sense". This is because colors and objects in the caption are treated equally without decoupling. (2) *Color-object mismatch*: An object whose color is not mentioned in the caption may take on the color from another object (e.g., the red car in the bottom example of Figure 1).

In this paper, we propose **L-CoDe**, a Language-based Colorization network using color-object Decoupled conditions. To deal with (1) color-object coupling, we adopt biaffine mechanism to predict the object-color corresponding matrix (OCCM), and use it to transfer the correspondence between visual regions and nouns to the correspondence between regions and adjectives[2], which is implemented by proposing a new attention transfer module (ATM). To further solve (2) color-object mismatch issue, we adopt a soft-gated injection module (SIM) to ensure that the color will not be applied to objects whose colors are not mentioned in the caption. L-CoDe contributes to a novel language-based colorization solution by

- decoupling colors and objects for correctly applying designated (could be unusual) color words to objects;
- improving the injection module to fulfill accurate color-object matching with a soft-gated mechanism; and
- a new dataset containing annotated ground-truth corresponding matrices with captions to supervise the training.

We demonstrate that L-CoDe provides higher quality colorization results both quantitatively and qualitatively, and validate its applicability in colorizing legacy photos.

2 Related Work

We provide a brief overview of relevant prior works according to different ways of user interaction.

Automatic colorization. Colorization could be conducted without user interaction. Relying entirely on learning to automate the colorization process has received increasing attention in recent years (Zhang, Isola, and Efros 2016; Zhao et al. 2018; Su, Chu, and Huang 2020). Cheng, Yang, and Sheng (2015) proposed a fully automatic colorization method by concatenating several pre-defined features and feeding them into a three-layer fully connected neural network. Deshpande, Rock, and Forsyth (2015) solved a linear system to colorize an image. Recently, deep learning based solutions have become the mainstream to automatically extract features and predict the colorized results. Iizuka, Simo-Serra, and Ishikawa (2016) and Zhao et al. (2018) presented a two-branch architecture that jointly learned and fused local image features and global priors (e.g., semantic labels). Isola et al. (2017) treated colorization as an image-to-image translation task and proposed a generative network. To handle multi-modal uncertainty in colorization, some works predict the color distribution of each pixel instead of a single color. Zhang, Isola, and Efros (2016) proposed a network trained with a multinomial cross entropy loss with rebalanced rare classes allowing unusual colors to appear. Larsson, Maire, and Shakhnarovich (2016) used hypercolumns (Hariharan et al. 2015) to interpret the semantic composition of the scene and the localization of objects to predict the color histograms of each pixel. Su, Chu, and Huang (2020) learned object-level semantics instead of image-level or pixel-level by training on the cropped object images and then fusing the learned object level and full-image features. Due to the multi-modal uncertainty, fully automatic solutions may not always produce satisfactory results, which could be complemented with various types of user interaction.

Scribble-based colorization. Early attempts leverage high-level user scribbles (e.g., color points or strokes) to guide the colorization process by propagating user-specified color scribbles based on some low-level similarity metrics (Levin, Lischinski, and Weiss 2004; Yatziv and Sapiro 2006; Luan et al. 2007). The pioneering work (Levin, Lischinski, and Weiss 2004) assumed that adjacent pixels with similar luminance should have similar color, and then solved an optimization problem based on this constraint. Several follow-up approaches focused on reducing color bleeding via edge detection (Huang et al. 2005) or improving the efficiency of color propagation with intrinsic distance (Yatziv and Sapiro 2006) or texture similarity (Qu, Wong, and Heng 2006; Luan et al. 2007). However, these methods predict the color of each pixel completely depending on user inputs, so they may require a lot of user scribbles. With a deep neural network (Zhang et al. 2017; Sangkloy et al. 2017), pixels not specified by the user are learned from training dataset, which alleviates the efforts.

Example-based colorization. This category of methods transfers the color from a reference image to the input image by computing the correspondences between them based on some similarity metrics (Welsh, Ashikhmin, and Mueller 2002; Liu et al. 2008; Bugeau, Ta, and Papadakis 2013; Charpiat, Hofmann, and Schölkopf 2008; Tai, Jia, and Tang 2005; Ironi, Cohen-Or, and Lischinski 2005; Chia et al. 2011; Gupta et al. 2012). The early work (Welsh, Ashikhmin, and Mueller 2002) transferred colors by matching global color statistics. For more accurate local transfer, correspondence techniques at different levels were proposed, including pixel level (Liu et al. 2008; Bugeau, Ta, and Papadakis 2013), segmented region level (Charpiat, Hofmann, and Schölkopf 2008; Tai, Jia, and Tang 2005; Ironi, Cohen-Or, and Lischinski 2005) and super-pixel level (Chia et al. 2011; Gupta et al. 2012; Dong et al. 2019, 2020).

These methods work remarkably well when the input and the reference share similar contents. However, finding an appropriate reference image is time-consuming and can be challenging for rare objects or complex scenes, even if using an automatic retrieval system (Chia et al. 2011). To solve these problems, He et al. (2018) proposed a deep learning approach, which allows controllability and is robust to reference selection. In order to further improve the robustness, Xu et al. (2020) adopted a novel two-way architecture to jointly learn faithful colorization with a related reference and plausible color prediction with an unrelated one, and Lu

[2] In this paper, nouns refer specifically to words describing the object category and adjectives refer only to words representing colors.

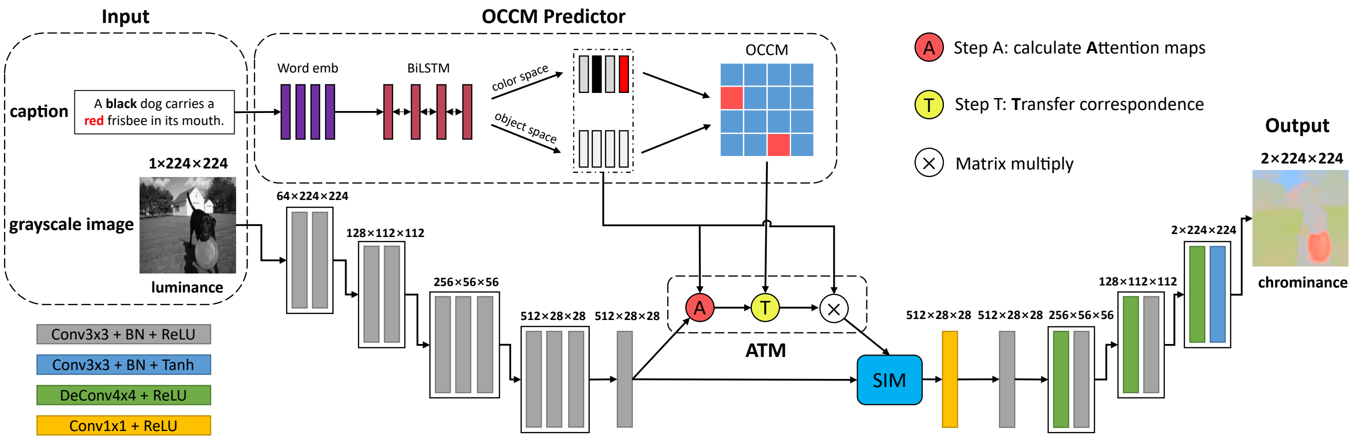


Figure 2: Pipeline of L-CoDe. L-CoDe takes as input a grayscale image (luminance) and a caption. First, the grayscale image is encoded into visual features. Then the OCCM and ATM are used to generate color-object decoupled conditions. The SIM injects color-object decoupled conditions into the visual regions described by the caption. Finally, modulated visual features are converted into chrominance by the decoder.

et al. (2020) designed an attention gating mechanism based network to fuse the semantic colors and global color distribution from the reference image, both methods achieved better colorization.

Language-based colorization. Recently, a language-based colorization method was presented to colorize grayscale images using sentences that describe objects with colors (Manjunatha et al. 2018). Context confusion and spatial inconsistency are common problems for language-based colorization. To deal with complex sentences, Chen et al. (2018) used recurrent attentive models to fuse image and language features, and to enhance the spatial consistency of colorized results. Xie (2018) adopted a semantic segmentation side-task to facilitate the learning of high-level semantics. Due to the flexibility of using natural language, language-based colorization derives many interesting applications. Zou et al. (2019) designed a system to colorize scene sketches guided by captions in a progressive way. Bahng et al. (2018) proposed 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 palettes.

3 Method

In this section, we first present an overview of the pipeline of L-CoDe. Then, we elaborate on the detailed designs of the three components: the OCCM predictor, ATM and SIM. Finally, we introduce the loss function and training details.

3.1 Overview

The pipeline of L-CoDe is illustrated in Figure 2. It works in the CIE Lab color space, which is perceptually linear. In the CIE Lab color space, each image is separated into a luminance channel L and two chrominance channels a and b. L-CoDe predicts two missing color channels of a gray-scale image conditioned on a caption.

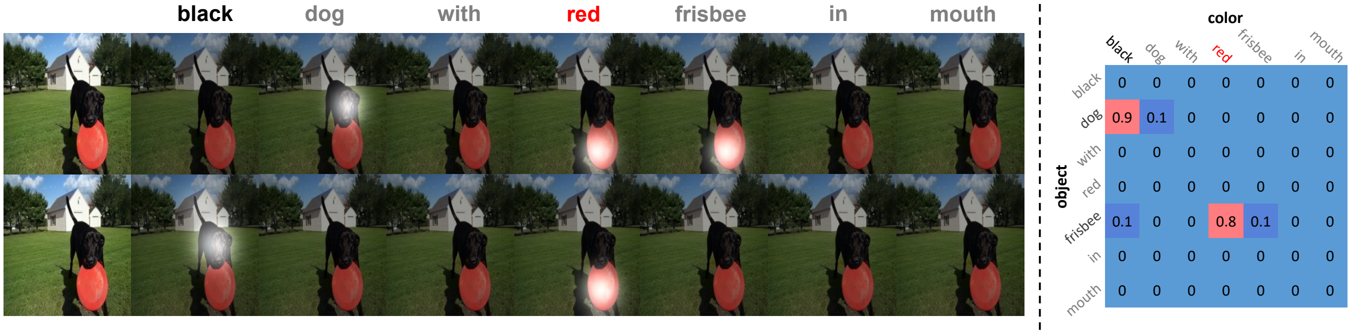
L-code adopts an encoder-decoder backbone. The encoder first extracts visual features $V \in \mathbb{R}^{D_v \times H \times W}$ from an input grayscale image, where D_v donates the number of channels, H and W are the height and width of the feature maps. Then three modules are proposed to generate the decoupled conditions and modulate the visual features. Finally, the decoder converts modulated visual features $V' \in \mathbb{R}^{D_v \times H \times W}$ into the colorized result.

Here we briefly introduce the proposed key modules, which will be explained in detail in the following subsections: (1) The object-color corresponding matrix (OCCM) is predicted by the OCCM predictor; (2) the attention transfer module (ATM) transfers the correspondence between visual regions and nouns to the correspondence between regions and adjectives using OCCM to provide color-object decoupled conditions; and (3) the soft-gated injection module (SIM) modulates visual features with decoupled conditions.

3.2 OCCM predictor

The previous methods (Manjunatha et al. 2018; Xie 2018) encode the caption into a single vector which mixes the nouns and adjectives together, resulting in stronger coupling issue. In order to distinguish the nouns and adjectives, we encode each word in the caption into a vector. Specifically, given a caption with N words, we use Bi-LSTM (Schuster and Paliwal 1997) to extract context-aware feature matrix $W \in \mathbb{R}^{D_w \times N}$. Each column of W represents a word vector, the dimension of which is D_w. Moreover, we predict the OCCM to find the correspondence between nouns and adjectives in the caption, which is required by the following ATM. Inspired by the biaffine attention in dependency parsing (Dozat and Manning 2016; Fernández-González and Gómez-Rodríguez 2020), we adopt two MLPs (Dozat and Manning 2016) f^{col} and f^{obj} to convert the word vectors into "object space" and "color space" respectively:

$$H_i^{\text{col}} = f^{\text{col}}(W_i), \quad H_i^{\text{obj}} = f^{\text{obj}}(W_i), \qquad (1)$$



	black	dog	with	red	frisbee	in	mouth
black	0	0	0	0	0	0	0
dog	0.9	0.1	0	0	0	0	0
with	0	0	0	0	0	0	0
red	0	0	0	0	0	0	0
frisbee	0.1	0	0	0.8	0.1	0	0
in	0	0	0	0	0	0	0
mouth	0	0	0	0	0	0	0

Figure 3: Visualization of attention maps before/after (top/bottom on the left) Step T and the OCCM. We visualize the attention maps by expanding them to 224 × 224 with Gaussian filters. The brighter the region in the visualization, the greater the probability that the object is located in that region.

where $H_i^{\text{obj}} \in \mathbb{R}^{D_h \times 1}$ and $H_i^{\text{col}} \in \mathbb{R}^{D_h \times 1}$ donate the i-th vector in "object space" and "color space" respectively. W_i is the i-th column vector of W. The corresponding relationship between color and object in the OCCM is predicted by:

$$T_{i,j} = \sigma((H_i^{\text{obj}})^\top U H_j^{\text{col}} + (H_i^{\text{obj}})^\top u). \tag{2}$$

If the i-th word is a noun whose color is described by the j-th word, the value of $T_{i,j}$ is close to 1. Otherwise, $T_{i,j}$ is close to 0. $\sigma(*)$ is the sigmoid function. $U \in \mathbb{R}^{D_h \times D_h}$ and $u \in \mathbb{R}^{D_h \times 1}$ are learned parameters.

3.3 Attention transfer module

To obtain the color-object decoupled condition, the widely used cross-modality attention (Xu et al. 2018), which synthesizes fine-grained images based on individual word representation, could be a useful reference. We briefly review the crucial steps in Xu et al. (2018) to make our following explanation self-contained: First, they project the word vectors into the semantic space of the visual vectors; then they calculate the dot product of the visual vectors (query) and the word vectors (key) to obtain the probability of each visual region corresponding to all words in the sentence; finally, the weighted sum of the word vectors (value) is calculated to generate the condition corresponding to each visual region.

However, in the colorization task, the input visual features lack color information, so the visual regions tend to correspond to nouns. Cross-modality attention (Xu et al. 2018) ignores the adjectives describing the unusual and counter-intuitive color details which finally makes the result stick to "common sense". In addition, the such an attention uses the same key and value, which also aggravates the coupling.

To overcome the problems above, we propose a new "attention-transfer" mechanism to map the correspondence between regions and nouns into the correspondence between regions and adjectives. In this way, we can use nouns to find the corresponding regions, and take the adjectives to colorize them. It has been suggested that using separate projection layers can make the key focus on matching with the visual regions, and the value is optimized towards generating a better condition (Liu et al. 2021). So we use two linear projection layers to convert word vectors from the "object space" and "color space" into the visual space obtaining $\hat{H}^{\text{obj}} \in \mathbb{R}^{D_v \times N}$ and $\hat{H}^{\text{col}} \in \mathbb{R}^{D_v \times N}$, and take them as the key and value respectively. The proposed ATM contains "A-T" as two crucial steps, which are detailed as follows.

- Step A: We obtain the correspondence between regions and nouns by calculating attention maps of visual features and words:

$$S = \tilde{V}^\top \hat{H}^{\text{obj}}, \tag{3}$$

where $\tilde{V} \in \mathbb{R}^{D_v \times M}$ is the flattened version of V, $M = H \times W$, and $S \in \mathbb{R}^{M \times N}$. Each element $S_{i,j}$ in the matrix S represents the score that the j-th words corresponds to the i-th position of visual features.

- Step T: To filter out the unreasonable pairs, we set the elements that are smaller than the threshold α in the OCCM to zero, and normalize it to $\bar{T}$ (using the l_1-norm). Then we use $\bar{T}$ to transfer the correspondence between regions and nouns to the correspondence between regions and adjectives, which obtains S'. An example for visualization of S/S' and OCCM is shown in Figure 3.

$$\bar{T}_{i,j} = \frac{T_{i,j}}{\sum_{k=0}^{N-1}(T_{i,k})}, \quad S' = S\bar{T}. \tag{4}$$

We use softmax to normalize S' to $\bar{S}'$, and obtain the decoupled conditions $C \in \mathbb{R}^{D_v \times M}$ by $C_i = \sum_{j=0}^{N-1} \bar{S}'_{i,j} \hat{H}_j^{\text{col}}$, where C_i is the i-th column of C and corresponds to the i-th position of visual features.

3.4 Soft-gated injection module

When describing an image using natural language, people may only want to mention the objects they are interested in. An example is given in the top left corner of Figure 2. Only "black dog" and "red frisbee" are provided in the input caption, while other objects like the sky, trees, grass, and house are not provided with designated color labels. To prevent these unmentioned objects from taking colors appearing in the caption, we propose the SIM module to deal with such color-object mismatch.

Observing that attention module focuses on visual regions whose color is mentioned in the caption (bottom left of Figure 3), we consider using S' to ensure that the color would not be applied to objects whose color are not mentioned in the caption. Specifically, we reshape S' to $N \times H \times W$ and applying 1×1-conv operation with sigmoid activation f^{conv} on S' to obtain the soft-gated mask $m \in \mathbb{R}^{1\times H\times W}$:

$$m = f^{\text{conv}}(S'). \tag{5}$$

The mask m indicates which visual regions have corresponding adjectives.

We adopt scale and shift parameters to modulate the visual features. The decoupled condition C is reshaped to $D_v \times H \times W$, and converted to scale and shift parameters by two 1×1-conv layers. Then we use the soft-gated mask to further constrain the modulation parameters:

$$\begin{aligned} \gamma' &= \gamma(C) \odot m + (1-m) \odot \mathbb{1}, \\ \beta' &= \beta(C) \odot m + (1-m) \odot \mathbb{0}, \end{aligned} \tag{6}$$

where $\gamma' \in \mathbb{R}^{D_v\times H\times W}$, $\beta' \in \mathbb{R}^{D_v\times H\times W}$, "$\odot$" donates element-wise multiplication, $\gamma(*)$ and $\beta(*)$ represent the convolution layers that convert C to the scale and shift parameters respectively. $\mathbb{1}$ and $\mathbb{0}$ denote tensors of ones and zeros respectively. Finally, the modulated feature V' can be defined as:

$$V' = \gamma' \odot \frac{V-\mu}{\sigma} + \beta'. \tag{7}$$

μ and σ are the estimated mean and standard deviation from aggregating both batch and spatial dimensions:

$$\begin{aligned} \mu &= \frac{1}{BHW}\sum_{b,h,w} V_{b,c,h,w}, \\ \sigma &= \sqrt{\frac{1}{BHW}\sum_{b,h,w}(V^2_{b,c,h,w} - \mu^2)}, \end{aligned} \tag{8}$$

where B donates the batch size.

3.5 Loss function and training

We adopt a smooth-l_1 loss (Huber 1992) with $\delta = 1$ as a robust estimator to train colorization network:

$$\begin{aligned} \ell_\delta(x,y) =& \frac{1}{2}(x-y)^2 \mathbb{1}_{\{|x-y|<\delta\}} + \\ & \delta(|x-y| - \frac{1}{2}\delta)\mathbb{1}_{\{|x-y|\geq\delta\}}. \end{aligned} \tag{9}$$

In addition, we optimize the binary cross entropy between the estimated OCCM and the ground-truth matrix:

$$\ell_{BCE}(x,y) = -(y\log(x) + (1-y)\log(1-x)). \tag{10}$$

The model is trained in an end to end manner. For the input, we resize the grayscale image to $1 \times 224 \times 224$ and repeat it as $3 \times 224 \times 224$. In the encoding stage, the visual feature maps in the first 4 convolutional blocks containing 2 (or 3) convolution layers are progressively halved spatially while doubling the feature channel number. In the decoding stage, the modulated feature maps are progressively doubled spatially while halving the feature channel number. All down-sampling layers use MaxPool with a stride of 2, while all upsampling layers use deconvolution with a stride of 2. BatchNorm layers (Ioffe and Szegedy 2015) are added after each convolutional block. All the convolutional layers use ReLU as the activation function, and only the last layer uses Tanh to constrain the output within a meaningful bound. The first four convolutional blocks in the network are initialized with pre-trained weights from a VGG16-BN model. The rest of the colorization network is initialized with the Xavier method.

We set the batch size to 16, $\alpha = 0.1$ in the ATM. We minimize our objective loss using Adam optimizer with learning rate set as 2×10^{-4} and momentum parameters $\beta_1 = 0.99$ and $\beta_2 = 0.999$. Experiments were conducted on two NVIDIA GTX 1080Ti GPUs.

4 Experiments

In this section, we present experimental results to validate the advantages of L-CoDe and demonstrate its applications. We start by describing the evaluation datasets and metrics. Then we compare our results with the state-of-the-art language-based colorization methods and carry out an ablation study to validate our ATM and SIM. We also demonstrate applications of our L-CoDe on colorizing legacy black and white photos. Finally, we conclude this section with an example of failure case.

4.1 Experimental setting

Datasets. Learning-based image colorization methods could be trained using the ImageNet dataset (Russakovsky et al. 2015). However, language-based colorization requires a caption describing the grayscale image, so existing language-based methods choose to use image datasets with captions, such as the COCO-Stuff (Caesar, Uijlings, and Ferrari 2018). For the captions in COCO-Stuff, there are many objects whose colors are absent. According to Xie (2018), we keep the images whose captions contain adjectives and have 59,371 training images and 2,486 validation images left. We further annotate the correspondence between objects and colors by hand as the basis for generating ground-truth matrices.

Evaluation metrics. Following the experimental protocol by Su, Chu, and Huang (2020), we report the PSNR and SSIM numbers to quantify the colorization quality. To compute the SSIM on color images, we average the SSIM values computed from individual channels. The recently proposed perceptual metric LPIPS by Zhang et al. (2018) (version 0.1; with VGG backbone) is also calculated and compared.

4.2 Comparisons with state-of-the-arts methods

The method proposed by Chen et al. (2018) is a conceptual design, so a fair comparison is difficult because it requires a separate implementation per dataset. The methods proposed by Manjunatha et al. (2018) and Xie (2018) are the most relevant language-based solutions, with which we hope to compare. However, both of them have their own training and testing datasets based on COCO-Stuff (Caesar, Uijlings, and Ferrari 2018). To make a fair comparison, we retrain these

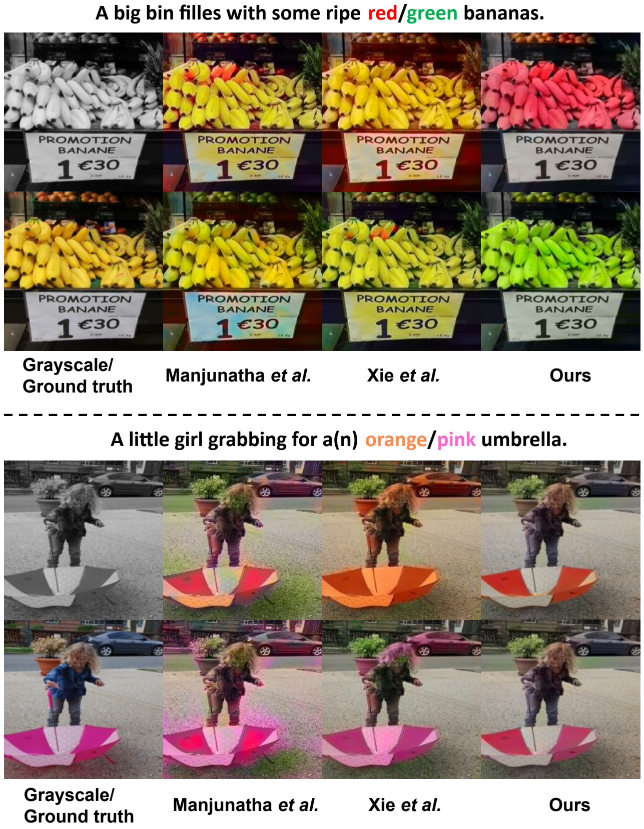


Figure 4: Qualitative comparisons with state-of-the-art methods (Manjunatha et al. 2018; Xie 2018). By successfully resolving the color-object mismatch and coupling issues, L-CoDe (ours) correctly modifies the color of designated objects (top) and avoids wrongly colorizing objects not mentioned in the caption (bottom).

networks using their publicly available code on the reorganized dataset. Note that Manjunatha et al. (2018) and our method take a grayscale image and a caption as input, while Xie (2018) also needs the segmentation of the grayscale image.

Quantitative comparisons. We show quantitative comparisons of colorization results conditioned on corresponding captions in Table 1. L-CoDe performs favorably against other methods on all three metrics, indicating that our colorization results to be more similar to ground truth images on these metrics.

Qualitative comparisons. Qualitative comparisons also demonstrate the effectiveness of our approach, as shown in Figure 4. We generate different color images of a single image by swapping out different colors in the caption to evaluate whether each method is able to colorize the corresponding object precisely. These examples show that L-CoDe is particularly effective in dealing with color-object coupling and color-object mismatch.

Table 1: Quantitative comparison result. L-CoDe (ours) performs best in three metrics. Throughout this paper, ↑ (↓) means higher (lower) is better.

Method	PSNR↑	SSIM%↑	LPIPS↓
(Manjunatha et al. 2018)	21.055	85.333%	0.282
(Xie 2018)	21.407	84.016%	0.298
Ours	**24.965**	**91.657%**	**0.169**

Table 2: User study result. We conduct two user study experiments to evaluate whether our colorization results are favored by human observers. L-CoDe (ours) achieves obviously higher scores in both experiments.

Method	Experiment-1	Experiment-2
(Manjunatha et al. 2018)	18.32%	16.68%
(Xie 2018)	18.68%	32.64%
Ours	**63.00%**	**50.68%**

As shown in the top two rows in Figure 4, methods of Manjunatha et al. (2018) and Xie (2018) have difficulties in changing the color of the bananas, which is strongly coupled to yellow in the dataset, to red or green, while our method successfully deals with this. In the bottom two rows, although the object mentioned in the caption is painted with the correct color, the regions not specified wrongly take on the color from the caption. For instance, the input caption specifies only the color of the umbrella, but the little girl's hair, the flowerpot, and the car are also colorized using the same color as the umbrella.

User study. In addition to quantitative and qualitative comparisons, we conduct two user study experiments to evaluate whether our colorization results are favored by human observers. We design two experiments: Experiment-1: In this task, we provide captions describing a ground truth color image, participants are shown with the ground truth image and three colorized results from three different methods: Manjunatha et al. (2018), Xie (2018), and ours, and asked to choose the most visually pleasing result with respect to the ground truth. But in this experiment, even if a colorized result is similar to the ground truth image, it cannot prove whether the caption correctly plays the restrictive role in the colorization process. Therefore, we design Experiment-2: In this task, we replace the color word in a caption with another random color word. Participants are shown a new caption and three colorized results from three different methods and asked to choose an image that matches best with the given caption.

Each experiment is composed of 100 tasks, and the image in each task is randomly selected from the testing dataset. We publish these two experiments on Amazon Mechanical Turk (AMT) and each experiment is completed by 25 participants. As shown in Table 2, our method achieves higher scores in both experiments, which confirms the subjective advantages of L-CoDe.

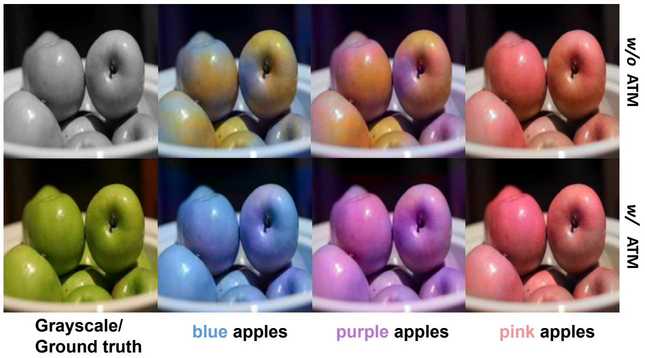


Figure 5: An example of ablation study by replacing the ATM module with attention in Xu et al. (2018). It becomes less effective to apply unusual colors to objects.

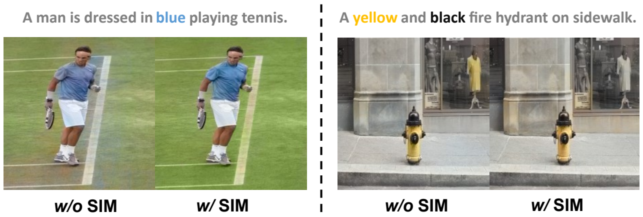

A man is dressed in blue playing tennis.
A yellow and black fire hydrant on sidewalk.
w/o SIM
w/ SIM
w/o SIM
w/ SIM

Figure 6: An example of ablation study by replacing the SIM module. There is dispersion and mismatch in the colorized images.

Table 3: Quantitative comparisons with replacing ATM (1), replacing SIM (2) and our complete method (Ours).

Method	PSNR↑	SSIM↑	LPIPS↓
(1)	24.778	91.643%	0.175
(2)	23.497	89.869%	0.207
Ours	**24.965**	**91.657%**	**0.169**

4.3 Ablation study

We evaluate the effectiveness of two new components — the ATM and the SIM in two ablation settings: (1) We replace the ATM with attention module in Xu et al. (2018) which takes the outputs of BiLSTM as inputs and discards the OCCM; (2) we replace the SIM by modulating the visual feature with γ and β rather than γ' and β'. Quantitative comparisons are shown in Table 3, where our full method achieves higher scores on all three metrics. After replacing the ATM, there is no significant difference in the quantitative results, because the colors in the test captions are the common colors of the objects. The qualitative comparison shown in Figure 5 demonstrates that, without the ATM the network becomes rather ineffective in changing colors according to captions due to the coupling of color-object pairs. After replacing the SIM, there is dispersion and mismatch in the colored images without the constraint of mask m. As shown in Figure 6, blue color spread to the grass (left), and yellow color was wrongly applied to the clothes behind (right). This demonstrates that our SIM is effective.

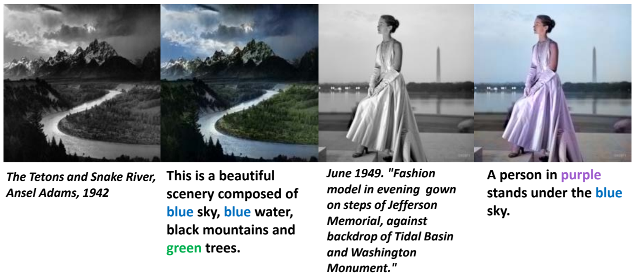

The Tetons and Snake River, Ansel Adams, 1942
This is a beautiful scenery composed of blue sky, blue water, black mountains and green trees.
June 1949. "Fashion model in evening gown on steps of Jefferson Memorial, against backdrop of Tidal Basin and Washington Monument."
A person in purple stands under the blue sky.

Figure 7: An example of colorization for legacy black and white photos.

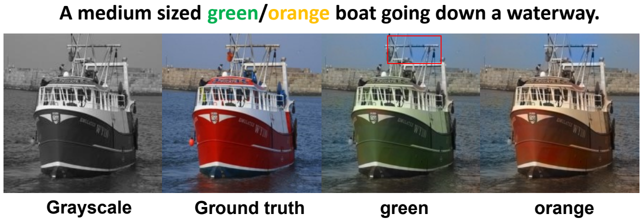


Figure 8: An example of failure case. Our method may produce bleeding across object boundaries when handling objects with fine structures along edges.

4.4 Colorizing legacy photos

Since L-CoDe is trained on "synthetic" grayscale images by removing the chrominance channels from natural color images, it is readily to be applied to add colors to legacy black and white photos. We show some example results in Figure 7, which demonstrates the generalization capability of the proposed method.

4.5 Failure case

We show an example of failure cases in Figure 8. It is difficult to handle edges of objects with fine structures, because of the limited resolution of image feature maps and attention maps. As a result, our method may produce visible artifacts such as color bleeding across object boundaries.

5 Conclusions

We propose L-CoDe, a Language-based Colorization network using color-object Decoupled conditions. L-CoDe successfully deals with the color-object coupling and color-object mismatch issues that result in incorrect caption-color correspondence. Although experimental results show that our work achieves state-of-the-art performance, we believe that there's still plenty of room for improvement. In our future work, we will consider adopting features and attention representations of higher resolution to achieve colorization with finer details.

Acknowledgements

This work is supported by National Natural Science Foundation of China under Grant No. 62136001, 62088102, 61872012.

References

Bahng, H.; Yoo, S.; Cho, W.; Keetae Park, D.; Wu, Z.; Ma, X.; and Choo, J. 2018. Coloring with words: Guiding image colorization through text-based palette generation. In *ECCV*.

Bugeau, A.; Ta, V.-T.; and Papadakis, N. 2013. Variational exemplar-based image colorization. *TIP*.

Caesar, H.; Uijlings, J.; and Ferrari, V. 2018. Coco-stuff: Thing and stuff classes in context. In *CVPR*.

Charpiat, G.; Hofmann, M.; and Schölkopf, B. 2008. Automatic image colorization via multimodal predictions. In *ECCV*.

Chen, J.; Shen, Y.; Gao, J.; Liu, J.; and Liu, X. 2018. Language-based image editing with recurrent attentive models. In *CVPR*.

Cheng, Z.; Yang, Q.; and Sheng, B. 2015. Deep colorization. In *ICCV*.

Chia, A. Y.-S.; Zhuo, S.; Gupta, R. K.; Tai, Y.-W.; Cho, S.-Y.; Tan, P.; and Lin, S. 2011. Semantic colorization with internet images. *ACM TOG*.

Deshpande, A.; Rock, J.; and Forsyth, D. 2015. Learning large-scale automatic image colorization. In *ICCV*.

Dong, X.; Li, W.; Wang, X.; and Wang, Y. 2019. Learning a deep convolutional network for colorization in monochrome-color dual-lens system. In *AAAI*.

Dong, X.; Li, W.; Wang, X.; and Wang, Y. 2020. Cycle-CNN for colorization towards real monochrome-color camera systems. In *AAAI*.

Dozat, T.; and Manning, C. D. 2016. Deep biaffine attention for neural dependency parsing. *CoRR*.

Fernández-González, D.; and Gómez-Rodríguez, C. 2020. Discontinuous constituent parsing with pointer networks. In *AAAI*.

Gupta, R. K.; Chia, A. Y.-S.; Rajan, D.; Ng, E. S.; and Zhiyong, H. 2012. Image colorization using similar images. In *ACM MM*.

Hariharan, B.; Arbeláez, P.; Girshick, R.; and Malik, J. 2015. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*.

He, M.; Chen, D.; Liao, J.; Sander, P. V.; and Yuan, L. 2018. Deep exemplar-based colorization. *ACM TOG*.

Huang, Y.-C.; Tung, Y.-S.; Chen, J.-C.; Wang, S.-W.; and Wu, J.-L. 2005. An adaptive edge detection based colorization algorithm and its applications. In *ACM MM*.

Huber, P. J. 1992. Robust estimation of a location parameter. In *Breakthroughs in statistics*.

Iizuka, S.; Simo-Serra, E.; and Ishikawa, H. 2016. Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM TOG*.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Ironi, R.; Cohen-Or, D.; and Lischinski, D. 2005. Colorization by Example. In *EGSR*.

Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. In *CVPR*.

Larsson, G.; Maire, M.; and Shakhnarovich, G. 2016. Learning representations for automatic colorization. In *ECCV*.

Levin, A.; Lischinski, D.; and Weiss, Y. 2004. Colorization using optimization. *ACM TOG*.

Liu, B.; Song, K.; Zhu, Y.; de Melo, G.; and Elgammal, A. 2021. TIME: Text and image mutual-translation adversarial networks. In *AAAI*.

Liu, X.; Wan, L.; Qu, Y.; Wong, T.-T.; Lin, S.; Leung, C.-S.; and Heng, P.-A. 2008. Intrinsic colorization. *ACM TOG*.

Lu, P.; Yu, J.; Peng, X.; Zhao, Z.; and Wang, X. 2020. Gray2ColorNet: Transfer more colors from reference image. In *ACM MM*.

Luan, Q.; Wen, F.; Cohen-Or, D.; Liang, L.; Xu, Y.-Q.; and Shum, H.-Y. 2007. Natural image colorization. In *EGSR*.

Manjunatha, V.; Iyyer, M.; Boyd-Graber, J.; and Davis, L. 2018. Learning to color from language. In *NAACL*.

Qu, Y.; Wong, T.-T.; and Heng, P.-A. 2006. Manga colorization. *ACM TOG*.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; C. Berg, A.; and Fei-Fei., L. 2015. ImageNet large scale visual recognition challenge. *IJCV*.

Sangkloy, P.; Lu, J.; Fang, C.; Yu, F.; and Hays, J. 2017. Scribbler: Controlling deep image synthesis with sketch and color. In *CVPR*.

Schuster, M.; and Paliwal, K. 1997. Bidirectional recurrent neural networks. *TSP*.

Su, J.-W.; Chu, H.-K.; and Huang, J.-B. 2020. Instance-aware image colorization. In *CVPR*.

Tai, Y.-W.; Jia, J.; and Tang, C.-K. 2005. Local color transfer via probabilistic segmentation by expectation-maximization. In *CVPR*.

Welsh, T.; Ashikhmin, M.; and Mueller, K. 2002. Transferring color to greyscale images. *ACM TOG*.

Xie, Y. 2018. *Language-guided image colorization*. Master's thesis, ETH Zurich, Departement of Computer Science.

Xu, T.; Zhang, P.; Huang, Q.; Zhang, H.; Gan, Z.; Huang, X.; and He, X. 2018. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*.

Xu, Z.; Wang, T.; Fang, F.; Sheng, Y.; and Zhang, G. 2020. Stylization-based architecture for fast deep exemplar colorization. In *CVPR*.

Yatziv, L.; and Sapiro, G. 2006. Fast image and video colorization using chrominance blending. *TIP*.

Zhang, R.; Isola, P.; and Efros, A. A. 2016. Colorful image colorization. In *ECCV*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*.

Zhang, R.; Zhu, J.-Y.; Isola, P.; Geng, X.; Lin, A. S.; Yu, T.; and Efros, A. A. 2017. Real-time user-guided image colorization with learned deep priors. *ACM TOG*.

Zhao, J.; Liu, L.; Snoek, J., C.G.M; and Shao, L. 2018. Pixel-level semantics guided image colorization. In *BMVC*.

Zou, C.; Mo, H.; Gao, C.; Du, R.; and Fu, H. 2019. Language-based colorization of scene sketches. *ACM TOG*.