L-CoDer: Language-based Colorization with Color-object Decoupling Transformer

Zheng Chang[#1], Shuchen Weng[#2], Yu Li[3], Si Li[*1], and Boxin Shi[2]

[1] School of Artificial Intelligence, Beijing University of Posts and Telecommunications
[2] NERCVT, School of Computer Science, Peking University
[3] International Digital Economy Academy
{zhengchang98,lisi}@bupt.edu.cn
{shuchenweng,shiboxin}@pku.edu.cn
liyu@idea.edu.cn

Abstract. Language-based colorization requires the colorized image to be consistent with the the user-provided language caption. A most recent work proposes to decouple the language into color and object conditions in solving the problem. Though decent progress has been made, its performance is limited by three key issues. (i) The large gap between vision and language modalities using independent feature extractors makes it difficult to fully understand the language. (ii) The inaccurate language features are never refined by the image features such that the language may fail to colorize the image precisely. (iii) The local region does not perceive the whole image, producing global inconsistent colors. In this work, we introduce transformer into language-based colorization to tackle the aforementioned issues while keeping the language decoupling property. Our method unifies the modalities of image and language, and further performs color conditions evolving with image features in a coarse-to-fine manner. In addition, thanks to the global receptive field, our method is robust to the strong local variation. Extensive experiments demonstrate our method is able to produce realistic colorization and outperforms prior arts in terms of consistency with the caption.

1 Introduction

Image colorization, the task of converting a grayscale image into a plausible color image, has been widely used in black-and-white image restoration, artist assistance, and advertising/film industry. However, automatic image colorization [21,23,30] is inherently an ill-posed problem, as there are multiple reasonable colors suitable for the grayscale image. Thus an interactive supervised signal is required to determine the unique solution. Commonly used signals (*e.g.*, user scribble [20,32] and reference example [10,18,28]) require either high-level artistic skill or time-consuming search, while taking language as the guidance of

[#] Equal contributions. * Corresponding author.

Fig. 1. We demonstrate three typical issues in existing language-based colorization approaches (*e.g.*, L-Code [25]) and how our method improves them correspondingly. **Left**: With modality unification, our method understands the intrinsic color properties behind words, therefore it could colorize images with word that describes abstract appearance (colorful) rather than specific colors (red for tomato and green for salad). **Middle**: The results of our method present more accurate color than previous work (the hat tends to be gray rather than purple), which benefits from the color representation evolving with image features. **Right**: Thanks to the transformer architecture, our method is robust to the locally strong texture variation (upper part of the sea) in grayscale image.

colorization is recently demonstrated to be a friendly way of user interaction for colorization [3,19,25,27].

Language-based colorization requires the colorized results to be consistent with the description of language. Though natural language shows superior potential as the conditions of colorization, the usage of it is the rose among thorns because of the huge semantic chasm between grayscale image and language inputs, which causes color-object coupling and color-object mismatch [3,19,27]. To handle these problems, researchers pay special attention to decoupling language conditions [25]. However, these existing methods [3,19,27,25] have several commonly remaining issues to be solved. *(i)* Large gap between modalities - previous methods leverage different architectures independently to encode two input sources into embedding features (CNN for image and LSTM for caption). There is an inevitable gap between these two types of features, and they are not easy to be fully and precisely unified at a later stage, which prevents the model from understanding the semantics of the language in depth (Fig. 1 left). *(ii)* Inaccurate language representation - by directly taking the outputs of LSTM as conditions to inject into image features, fixed embedding limits the representation power of language, which causes inaccurate colorization (Fig. 1 middle). *(iii)* Local vulnerability - color bleeding occurs when there is a strong variation of texture or luminance in the local context, as the semantics is cued from the local regions in grayscale images (Fig. 1 right).

We for the first time introduce transformer into language-based colorization and propose **L-CoDer**, a **L**anguage based **Co**lorization with color-object **De**coupling transforme**r**, to deal with aforementioned issues as follows: *(i)* Thanks to the color-object decoupling transformer we proposed, both image and decou-

pled language conditions are unified in one modality as tokens, which is beneficial to understand intrinsic color properties behind the word (Fig. 1 left). *(ii)* Every transformer block extracts and fuses the self-modality and corresponding cross-modality token features so that the language conditions evolve from coarse to fine instead of being kept with fixed semantic, which provides adaptive supervisory signals to generate accurate and plausible colors (Fig. 1 middle). *(iii)* Benefited from the global receptive field of transformer, L-CoDer has stronger robustness to locally strong variation of texture or luminance, which further improves the colorization quality (Fig. 1 right). Our contributions are summarized as:

- We propose the color-object decoupling transformer to deal with the large gap between modalities, inaccurate language representation, and local vulnerability issues that reside in language-based colorization.
- We design the decoupling attention to make images globally and bidirectionally interact with language features while maintaining the decoupled properties of language conditions.
- We organize the decoupled language tokens into a coarse-to-fined representation, which provides adaptive supervisory signals evolving with image features and bringing in accurate and plausible colorized results.

We conduct our experiments in the extended COCO-Stuff dataset [25] and demonstrate that our transformer model achieves state-of-the-art performance with better colorized quality and condition consistency in both quantitative and qualitative results.

2 Related Works

Automatic colorization. Automatic colorization approaches colorize grayscale image without any external hints, which learns data distribution on large-scale datasets and directly estimate the proper colors. A part of works [4,12,15,30] focus on feature engineering to explore the effective architecture. Some works pay attention to advanced generative models, *e.g.*, VAE [5], GAN [23], cINN [2], and transformer [14]. Recently, to further extract semantic information, prior knowledge is widely introduce into automatic colorization methods, *e.g.*, detection boxes [21], segmentation masks [33], pretrained GAN [26]. However, automatic colorization methods suffer from multi-modal uncertainty and may fail to generate results satisfying users' expectations.

language-based colorization. Language-based colorization is to colorize gray images under the guidance of user-given caption, which is presented by Manjunatha *et al.* [19]. After that, researchers start to explore the way to fuse image and language features spatially by adopting recurrent attentive model [3]. Generating segmentation mask as the side-task is later introduced to jointly optimize the colorization results [27]. A more recent work decouples language conditions into object space and color space, which solves color-object coupling and color-object mismatch problem [25]. Although noticeable progress has been made in the task, the modality differences still largely affect the performance of colorization.

Vision transformer. Transformer is firstly proposed and demonstrated to make great success in natural language processing (NLP) to model sequence data [22]. Computer vision researchers have find it also performs excellent on various visual computing problems, *e.g.*, classification [9], detection [36], segmentation [34], super resolution [16], inpainting [17], to name a few. Benefit by the generality of transformer, it works well in cross-modality task, like referring segmentation [7], image-text retrieval [13], VQA [35] and text-to-image generation [8,29]. Vision transformer has also been applied to automatic colorization [14], but it remains space for exploration in language-guided interactions.

3 Method

To make this paper self-contained, we first review the language condition decoupling proposed by L-CoDe [25] to clarify the necessity of language decoupling. Next, we present the overview of L-CoDer and elaborate on the detailed designs of modules. In our approach, we address the issues of large gap between modalities, inaccurate language representation, and local vulnerability, which previous methods suffer from. Finally, we introduce the loss functions and training details.

3.1 Language Condition Decoupling for Colorization

Language condition decoupling mainly solves two main problems: *(i)* Color-object coupling: When the specified color and object combination is less observed in the dataset, the model may fail to apply the corresponding color to the object. *(ii)* Color-object mismatch: The model may incorrectly colorize the object whose color is not mentioned in the caption with the color of another object. For example, given language "green bananas on the plate" and a corresponding image, the model without decoupling may fail to colorize bananas with green because the banana object is always combined with yellow (color-object coupling); or it may incorrectly colorize plate with green, which is used to describe the bananas (color-object mismatch).

Instead of encoding caption into a single vector, L-CoDe decouples caption into adjective vectors that represent colors (*e.g.*, green) and noun vectors that describe the object category (*e.g.*, bananas and plate). After that, it predicts the combination of adjective vectors and noun vectors and constructs an object-color corresponding matrix (OCCM). A binary cross entropy (BCE) loss is used to supervise the optimization of predicted OCCM [25], which ensures the word "green" is combined with the word "bananas".

It is required to design unique modules to apply decoupled language conditions into colorization networks. In L-CoDe [25], the attention transfer module is presented for dealing with color-object coupling problem. Specifically, it uses noun vectors to find the corresponding image regions with a standard attention operator, and transfers the correspondence between regions and nouns into the correspondence between regions and adjectives with predicted OCCM. Therefore, regions corresponding to noun vectors are colorized with the adjective vectors in the combination. For avoiding color-object mismatch issue, the soft-gated

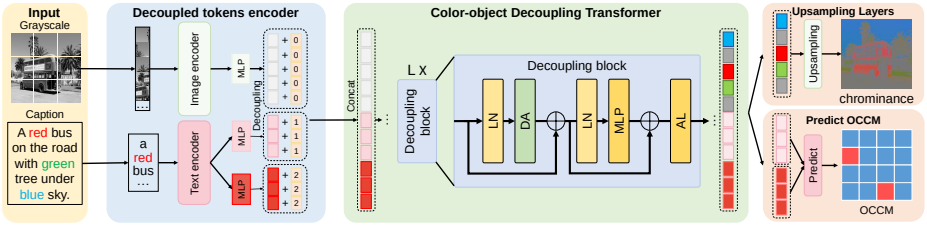

Fig. 2. Framework of L-CoDer. The **decoupled tokens encoder** maps the input grayscale image and language into the same modality, but they are represented in disparate feature spaces: Each word is decoupled as a noun token and an adjective token, and each image patch is encoded as an image token (Sec. 3.3). After concatenating all the tokens, the **color-object decoupling transformer** makes global interaction between tokens of the same or different modalities to supervise the evolution of tokens bidirectionally from coarse to fine with several decoupling blocks (Sec. 3.4). The **decoupling attention** (DA) in decoupling block is presented to avoid color-object coupling and color-object mismatch problems, which makes image tokens interact with decoupled language tokens while maintaining the decoupled properties of language (Sec. 3.5). The **upsampling layers**, composed of a stack of convolutions, are used to convert the image tokens at the finest grain into two missing chrominance channels with user-desired resolution (Sec. 3.6). The OCCM [25] (briefly recalled in Sec. 3.1) is calculated by noun tokens and adjective tokens, which is used to maintain decouple property in decoupling blocks.

injection module is designed. In detail, a soft-gated mask is constructed using the transferred attention maps as input, which guides the injection of decoupled language conditions to ensure that colors are not applied to objects not mentioned in the caption.

However, there remain difficult problems in language-based colorization task, *e.g.*, large gap between modalities, inaccurate language representation, and local vulnerability, which are shown in Fig. 1 from left to right, respectively. Transformer could unify the modality of image and language, dynamically represent language conditions, and have robustness to locally strong variation of texture or luminance, which motivates us to design L-CoDer to solve these problems.

3.2 L-Coder Framework

We illustrate the framework of L-CoDer in Fig. 2. It works in the CIE *Lab* color space, which requires models to generate two missing chrominance channels corresponding to the input grayscale image (as the luminance channel) under the guidance of user-specified language. L-CoDer is composed of a decoupled tokens encoder, a color-object decoupling transformer with decoupling attentions, and several upsampling layers. Two loss functions are used during training: *(i)* the ground truth image as a candidate colorized results, and *(ii)* the ground truth OCCM to constraint the semantic decoupling of object tokens and color tokens. The parameter settings are shown in the supplementary.

3.3 Decoupled Tokens Encoder

This module is proposed to encode both image and language as tokens so that they could be unified in one modality, which helps bridge the large gap between modalities. The decoupling design also avoids color-object coupling and color-object mismatch issues mentioned in Sec. 3.1. The decoupled tokens encoder is composed of an image encoder, a language encoder, and a decoupling module. Given a grayscale image $I \in \mathbb{R}^{H\times W}$ where H and W are the height and width of the input image, the image encoder first repeats the grayscale image for adapting to the input of ViT [9] as $(I, I, I) \in \mathbb{R}^{H\times W\times 3}$, and then reshapes it into N patches as $I_{\mathrm{P}} = [I_{\mathrm{P}}^1, \ldots, I_{\mathrm{P}}^N] \in \mathbb{R}^{N\times P^2\times 3}$, where (P, P) is the patch resolution and $N = HW/P^2$. After that, image encoder feeds patches into a standard ViT [9] to extract global features and generate image tokens $T_{\mathrm{I}} = [T_{\mathrm{I}}^1, \ldots, T_{\mathrm{I}}^N] \in \mathbb{R}^{N\times C_{\mathrm{I}}}$, where C_{I} is the channel number of image tokens.

We use BERT [6] as our language encoder, which encodes input caption $[w_1, \ldots, w_M]$ into language tokens $T_{\mathrm{L}} = [T_{\mathrm{L}}^1, \ldots, T_{\mathrm{L}}^{\mathrm{M}}] \in \mathbb{R}^{M\times C_{\mathrm{L}}}$, where M is the number of words in the caption and C_{L} is the channel number of language tokens. Note that we build our dictionary based on BERT, so that it includes a large vocabulary and the isolated words never appear in the training dataset will also be assigned with a pretrained embedded vector.

After the image and language are encoded, we decouple language conditions to handle color-object coupling and color-object mismatch problems [25] mentioned in Sec. 3.1. Specifically, we adopt a Multi-Layer Perceptron (MLP) to map image tokens into latent space, and another two MLPs to convert language tokens into object space and color space separately, written as:

$$T_{\mathrm{img}} = f^{\mathrm{img}}(T_{\mathrm{I}}), \qquad T_{\mathrm{obj}} = f^{\mathrm{col}}(T_{\mathrm{L}}), \qquad T_{\mathrm{col}} = f^{\mathrm{col}}(T_{\mathrm{L}}), \tag{1}$$

where $T_{\mathrm{img}} = [T_{\mathrm{img}}^1, \ldots, T_{\mathrm{img}}^N] \in \mathbb{R}^{N\times C_{\mathrm{s}}}$ are image tokens in latent space, $T_{\mathrm{obj}} = [T_{\mathrm{obj}}^1, \ldots, T_{\mathrm{obj}}^M] \in \mathbb{R}^{M\times C_{\mathrm{s}}}$ are noun tokens represented in object space, and $T_{\mathrm{col}} = [T_{\mathrm{col}}^1, \ldots, T_{\mathrm{col}}^M] \in \mathbb{R}^{M\times C_{\mathrm{s}}}$ are adjective tokens represented in color space, and C_{s} is the number of embedding channels.

Inspired by ViLT [13], we introduce the modal-type embedding vectors $T_{\mathrm{type}} \in \mathbb{R}^{C_{\mathrm{s}}}$ to distinguish token modalities. We add modal-type embedding vectors $[T_{\mathrm{type}}^0, T_{\mathrm{type}}^1, T_{\mathrm{type}}^2]$ to image tokens, noun tokens, and adjective tokens separately:

$$\hat{T}_{\mathrm{img}}^i = T_{\mathrm{img}}^i + T_{\mathrm{type}}^0, \qquad \hat{T}_{\mathrm{obj}}^j = T_{\mathrm{obj}}^j + T_{\mathrm{type}}^1, \qquad \hat{T}_{\mathrm{col}}^j = T_{\mathrm{col}}^j + T_{\mathrm{type}}^2, \tag{2}$$

where $i \in \{1, \ldots, N\}$ and $j \in \{1, \ldots, M\}$.

3.4 Color-object Decoupling Transformer

With the color-object decoupling transformer, the semantic of decoupled language tokens evolves with image features from coarse to fine, which avoids inaccurate language representation issue. We show the attention maps at different layers to illustrate the focus of colorization changes with the evolution of tokens

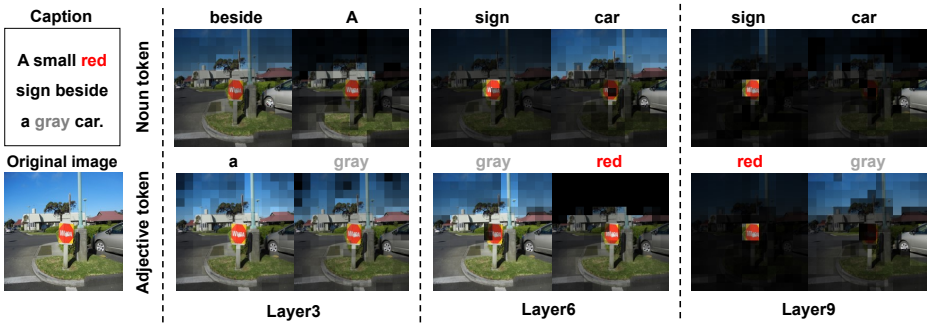

Fig. 3. We show the top-2 most attended words by the color-object decoupling transformer at layers 3, 6, and 9. As the number of layers increases, the model gradually finds the most important noun tokens (sign and car) and adjective tokens (red and gray). Meanwhile, the semantic of tokens evolves towards more accurate representation in a coarse-to-fine manner, *e.g.*, nouns tokens find regions of corresponding objects, and adjective tokens find regions with similar colors.

in Fig. 3. In addition, thanks to the ability of transformer to capture global dependencies, our method has stronger robustness to locally strong variation of texture or luminance. The Z^1_{img}, Z^1_{obj}, and Z^1_{col} generated from $\hat{T}_{\text{img}}$, $\hat{T}_{\text{obj}}$, and $\hat{T}_{\text{col}}$ by separate fully connected layers are the initial input of the proposed color-object decoupling transformer, which we introduce in detail next.

The color-object decoupling transformer is made up of L decoupling blocks (depending on the model variant), and each decoupling block contains a decoupling layer (DL) and an adaptive layer (AL). Given the input of the i-th block that includes image tokens Z^i_{img}, and decoupled language tokens Z^i_{obj} and Z^i_{col}, we formulate the process of extraction intermediate feature as:

$$[Z^{i+1}] = \text{AL}(\text{DL}([Z^i])), \qquad i \in \{1, \ldots, L\} \tag{3}$$

where $[Z^i]$ is the abbreviation of $[Z^i_{\text{img}}; Z^i_{\text{obj}}; Z^i_{\text{col}}] \in \mathbb{R}^{(N+2M)\times C_z}$, $Z^i_{\text{img}} \in \mathbb{R}^{N\times C_z}$, Z^i_{obj}, $Z^i_{\text{col}} \in \mathbb{R}^{M\times C_z}$, and C_z is the channel number.

Decoupling layer (DL) is modified from standard transformer block, which contains an MLP that has two fully connected layers, a decoupling attention (DA) (Sec. 3.5), and a LayerNorm (LN) layer added before MLP and DA. This process is formulated as:

$$[\bar{Z}^i] = \text{DA}\ (\text{LN}([Z^i])) + [Z^i], \qquad i \in \{1, \ldots, L\} \tag{4}$$

$$[\hat{Z}^i] = \text{MLP}(\text{LN}([\bar{Z}^i])) + [\bar{Z}^i]. \qquad i \in \{1, \ldots, L\} \tag{5}$$

The operator of adaptive layer (AL) depends on the modality of tokens. Specifically, for image tokens Z^i_{img}, AL reshapes them into spatial space followed by a convolution operator; but for decoupled language tokens Z_{obj} and Z_{col}, AL performs a fully connected operation, written as:

$$[Z^{i+1}] = [\text{F}_{\text{conv}}(\hat{Z}^i_{\text{img}}); \text{F}_{\text{fc}}([\hat{Z}^i_{\text{obj}}; \hat{Z}^i_{\text{col}}])], \qquad i \in \{1, \ldots, L\} \tag{6}$$

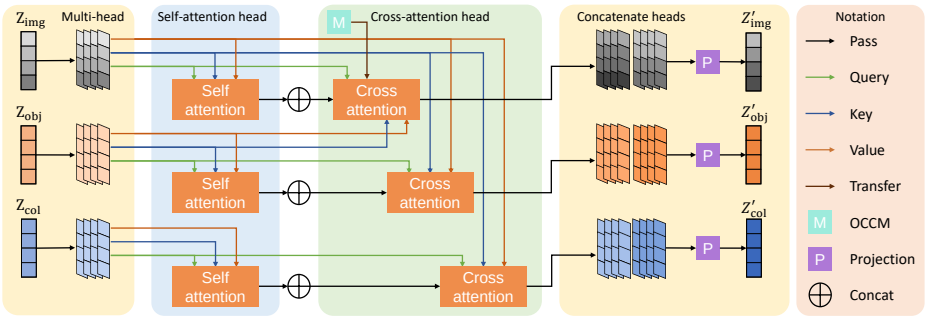

Fig. 4. Illustration of decoupling attention. The tokens are firstly projected into multi-head feature space. Next, tokens in each space are fed into self-attention and cross-attention heads separately to extract high-level semantic. Finally, we concatenate the output of self-attention and cross-attention heads in the multi-head feature space, and project them into common feature space to fuse semantic from various perspectives as the output of decoupling attention.

where $\mathrm{F}_{\mathrm{conv}}$ is a 3×3 convolution layer and F_{fc} is a fully connected layer. AL is similar to RTSB [16], which is designed to take convolution operator to enhance the translational equivariance of our transformer.

3.5 Decoupling Attention

To maintain the decoupled properties of language conditions, we propose decoupling attention to ensure the interaction between decoupled language tokens and image tokens. As shown in Fig. 4, decoupling attention is composed of H self-attention heads, H cross-attention heads, and a projection layer.

For self-attention heads, tokens of each modality calculate similarity with themselves to extract high-level global features; while for cross-attention heads, tokens cue semantic from other modal tokens: *(i)* We locate objects in image tokens Z_{img} with noun tokens Z_{obj}, and inject color with adjective tokens Z_{col}; *(ii)* Inspired by VLT [7], we integrate features of image tokens Z_{img} into noun tokens Z_{obj} to help understand perspectives and emphasis of the sentence and further improve the accuracy of location; *(iii)* As the image tokens Z_{img} are gradually colorized, we also fuse image features into adjective tokens Z_{col}, which keeps the accuracy of color semantic by adjusting the features of tokens. For each modality, a projection layer is designed to fuse the results of H self-attention heads and H cross-attention heads to output the refined tokens. We formulate the decoupling attention as three steps:

Step 1): For each attention head, we project $[Z_{\mathrm{img}}; Z_{\mathrm{obj}}; Z_{\mathrm{col}}]$ into query, key, and value feature space by fully connected layers, and calculate attention maps to measure the relevance between tokens and themselves (self-attention head) or corresponding tokens of another modality (cross-attention head):

$$A^{\mathrm{slf}}_{i,h} = Z^{\mathrm{qry}}_{i,h} \times (Z^{\mathrm{key}}_{i,h})^{\top}, \ \bar{A}^{\mathrm{crs}}_{\mathrm{img},h} = Z^{\mathrm{qry}}_{\mathrm{img},h} \times (Z^{\mathrm{key}}_{\mathrm{obj},h})^{\top}, \ A^{\mathrm{crs}}_{j,h} = Z^{\mathrm{qry}}_{j,h} \times (Z^{\mathrm{key}}_{\mathrm{img},h})^{\top}, \quad (7)$$

where $i \in \{\text{img}, \text{obj}, \text{col}\}$, $j \in \{\text{obj}, \text{col}\}$, and $h \in \{1, \ldots, H\}$ is the index of attention heads.

Step 2): We use the predicted OCCM $M_{\text{occm}} \in \mathbb{R}^{M \times M}$ to transfer cross-modality attention map of object tokens $\bar{A}^{\text{crs}}_{\text{img},h}$ to apply correct colors to corresponding objects position:

$$A^{\text{crs}}_{\text{img},h} = \bar{A}^{\text{crs}}_{\text{img},h} \times M_{\text{occm}}. \tag{8}$$

M_{occm} is calculated by decoupled language tokens Z^{l-1}_{obj} and Z^{l-1}_{col} at last block:

$$M_{\text{occm}} = \text{Norm}(\sigma(Z^{l-1}_{\text{obj}} U (Z^{l-1}_{\text{col}})^\top + Z^{l-1}_{\text{obj}} u), 0.1), \tag{9}$$

where $l \in \{1, \ldots, L\}$ is the index of decoupling blocks, σ is the sigmoid function, $U \in \mathbb{R}^{C_{\text{in}} \times C_{\text{in}}}$, $u \in \mathbb{R}^{C_{\text{in}} \times M}$ are learnable parameters, and $\text{Norm}(A, v)$ is a function to normalize matrix with ℓ_1 normalization after setting elements in matrix A smaller than v as zero. The process of applying OCCM is in the supplementary.

Step 3): We further use the softmax to normalize all attention maps $\hat{A} = \text{Softmax}(A/C_{\text{in}})$, which are used as the soft gate to inject assigned tokens:

$$Z^{\text{slf}}_{i,h} = \hat{A}^{\text{slf}}_{i,h} \times Z^{\text{val}}_{i,h}, \qquad Z^{\text{crs}}_{\text{img},h} = \hat{A}^{\text{crs}}_{\text{img},h} \times Z^{\text{val}}_{\text{col},h}, \qquad Z^{\text{crs}}_{j,h} = \hat{A}^{\text{crs}}_{j,h} \times Z^{\text{val}}_{\text{img},h}. \tag{10}$$

We finally concatenate H self-attention heads and H cross-attention heads and projected them into the high-level feature space:

$$Z'_i = ([Z^{\text{slf}}_{i,0} \quad Z^{\text{crs}}_{i,0} \quad \ldots \quad Z^{\text{slf}}_{i,H} \quad Z^{\text{crs}}_{i,H}])W^{\text{proj}}_i, \qquad i \in \{\text{img}, \text{obj}, \text{col}\} \tag{11}$$

where $W^{\text{proj}}_i \in \mathbb{R}^{C_z \times C_z}$ is the parameter matrix.

3.6 Upsampling Layers

Upsampling layers use colorized image tokens Z^{L+1}_{img} to generate colorful image with user-desired resolution. Specifically, the colorized image tokens are first reshaped into spatial resolution $\hat{Z}_{\text{img}} \in \mathbb{R}^{\sqrt{N} \times \sqrt{N} \times C_z}$, and then fed into a stack of transposed convolutions to upsample the resolution and predict the two chrominance channels. We concatenate the grayscale image I_{L} and predict two chrominance channels to obtain the colorized result in the CIE *Lab* space, written as:

$$I_{\text{Lab}} = \text{Concat}(I_{\text{L}}, \text{G}_{\text{deconv}}(\hat{Z}_{\text{img}})), \tag{12}$$

where G_{deconv} is compose of 4 upsampling blocks with transposed convolution layers to achieve 16 times larger resolution.

3.7 Learning

There are two losses we use to supervise the optimization as L-CoDe [25]: *(i)* a smooth-ℓ_1 loss with $\delta = 1$ to supervise the colorized images:

$$L_\delta(x, y) = \frac{1}{2}(x - y)^2 \mathbb{1}_{\{|x-y|<\delta\}} + \delta(|x - y| - \frac{1}{2}\delta)\mathbb{1}_{\{|x-y|\geq\delta\}}, \tag{13}$$

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

Category	Method	PSNR↑	SSIM↑	LPIPS↓
Automatic	CIC [30]	22.156	89.705%	0.224
	DeOldify [1]	21.708	86.451%	0.255
	InstColor [21]	23.914	90.618%	0.194
	ChromaGAN [23]	22.085	84.161%	0.275
Language-based	LBIE [3]	22.092	85.197%	0.265
	ML2018 [19]	21.055	85.333%	0.282
	Xie2018 [27]	21.407	84.016%	0.298
	L-CoDe [25]	24.965	91.657%	0.169
Ablation	W/o decouple	25.014	90.724%	0.173
	W/o evolution	25.141	91.155%	0.167
	W/o bidirection	25.135	91.131%	0.168
	W/o upsample	25.305	91.464%	0.161
Ours	L-CoDer	**25.504**	**91.963**%	**0.159**

and *(ii)* a binary cross entropy loss to optimize last predicted OCCM towards the ground truth matrix:

$$L_{\text{BCE}}(x, y) = -(y \log(x) + (1 - y) \log(1 - x)). \tag{14}$$

We jointly optimize L_δ and L_{BCE} as:

$$L = \alpha L_\delta + \beta L_{\text{BCE}}, \tag{15}$$

where we set $\alpha = 1$ and $\beta = 0.0001$.

We train L-CoDer 40 epochs with batchsize 32 for 12 hours on 4 NVIDIA TITAN RTX GPUs. We use AdamW optimizer to minimize our losses with learning rate as 1×10^{-5}, momentum parameters $\beta_1 = 0.99$ and $\beta_2 = 0.999$.

4 Experiments

Dataset. We conduct our experiments on the extended COCO-Stuff dataset proposed in L-CoDe [25], which includes 59K training images and 2.4K evaluation images of 224×224 resolution, with annotated correspondence between objects and colors as the basis for generating ground truth of OCCM.

Evaluation Metrics. Following L-CoDe [25], we report Peak Signal-to-Noise Ratio (PSNR) [11], Structural Similarity Index Measure (SSIM) [24], and Learned Perceptual Image Patch Similarity (LPIPS) [31] to quantify the colorization quality. We further conduct user studies to evaluate whether our results are favored by human observers.

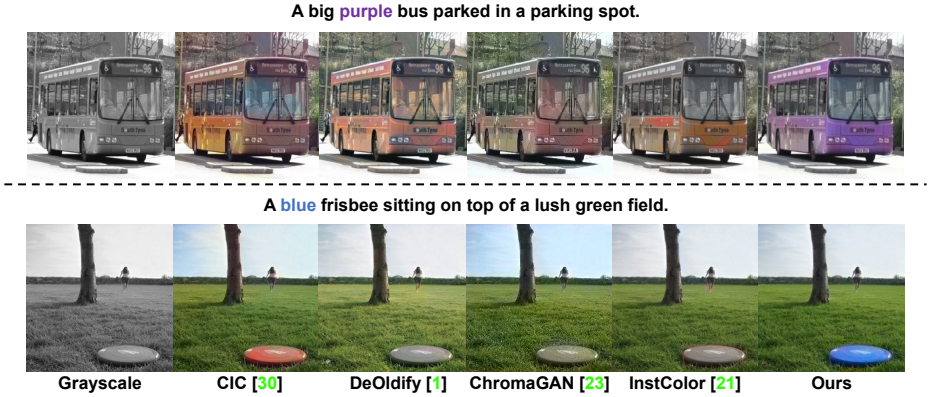

Fig. 5. Comparison with automatic colorization methods. Without the interactive supervised signal, automatic colorization methods cannot change the color of the bus in the top and the frisbee in the bottom, if the users have special request.

Table 2. User study results. Our method outperforms other approaches with the highest scores on both experiments.

Experiment	LBIE [3]	ML2018 [19]	Xie2018 [27]	L-CoDe [25]	Ours
Reality	10.68%	10.56%	15.72%	25.76%	**37.28**%
Corresponding	10.80%	13.56%	22.00%	23.40%	**30.24**%

4.1 Comparisons with State-of-the-art Methods

We make comparisons with four automatic colorization methods to demonstrate the necessity of language conditions as supervisory signal of colorization task, including CIC [30], Deoldify [1], ChromaGAN [23], and InstColorization [21]. We also make comparisons with another four language-based colorization approaches to show the improvement by overcoming large gap between modalities, inaccurate language representation, and local vulnerability problems, which contains LBIE [3], ML2018 [19], Xie2018 [27], and L-CoDe [25].

Qualitative comparisons. The automatic methods suffer from multi-modal uncertainty, which degrades the quality of colorized images. We show the comparison with automatic methods in Fig. 5, where the bus and frisbee could be colorized as any reasonable color, so the model cannot figure out the most appropriate color and may not meet the users' requirement. The comparison with language-based approaches is shown in Fig. 6, where we show overall quality improvement, word ontology understanding, accurate color representation, and local variation robustness from top to the bottom separately.

Quantitative comparisons. We show the quantitative comparison results in Tab. 1. Our method (L-CoDer) outperforms all compared methods of both automatic and language-based approaches, and achieves the best PSNR, SSIM, and LPIPS scores.

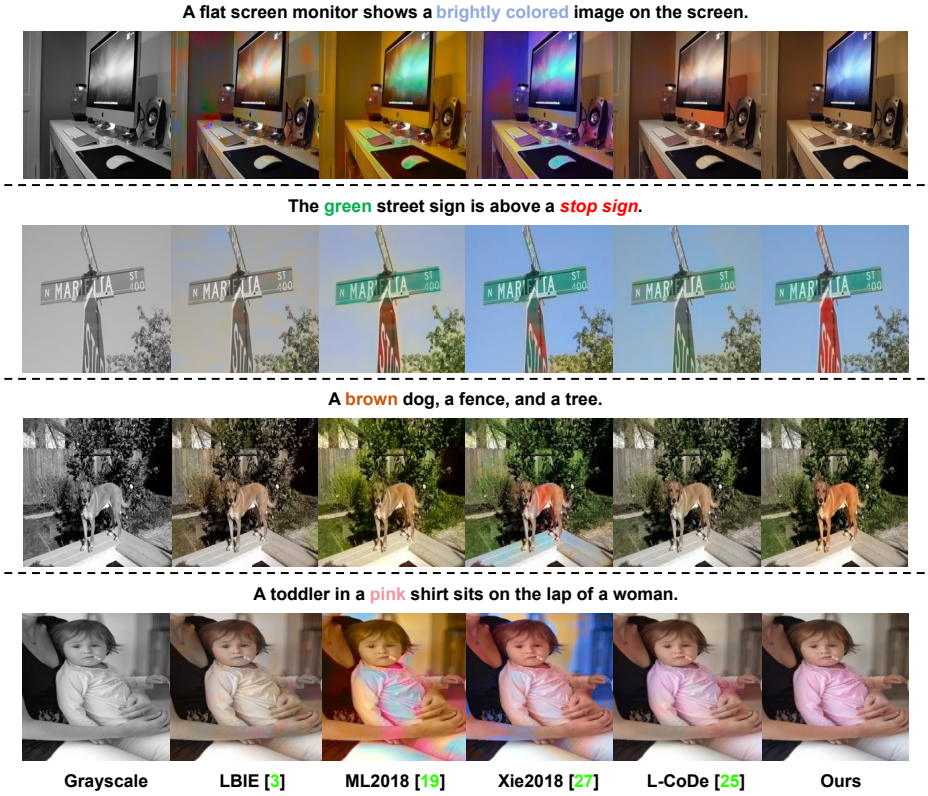

Fig. 6. Comparison with language-based colorization methods. **First row**: Our method could generate more realistic images. **Second row**: Our method cues intrinsic color property behind words (inferring red from words "stop sign"). **Third row**: Our method colorizes objects with more accurate color (dog with accurate brown). **Fourth row**: Our method shows robustness to locally strong texture variation (correctly colorizing pink shirt near the arm).

4.2 User Study

We conduct two user study experiments to evaluate whether our colorization results are more favored by human observers rather than other language-based colorization approaches. We perform two experiments following the setting of L-CoDe [25]: *(i)* Reality experiment: Participants are shown a ground truth image and five language-based colorization results, along with a caption that describes the ground truth color image, and asked to choose the result that is most visually pleasing with respect to the ground truth. *(ii)* Corresponding experiment: We randomly replace a word that describes the appearance of objects in the caption with another one, and use the modified caption to re-colorized five results. Shown modified caption and re-colorized results, participants are asked to choose an image that matches best with the given caption.

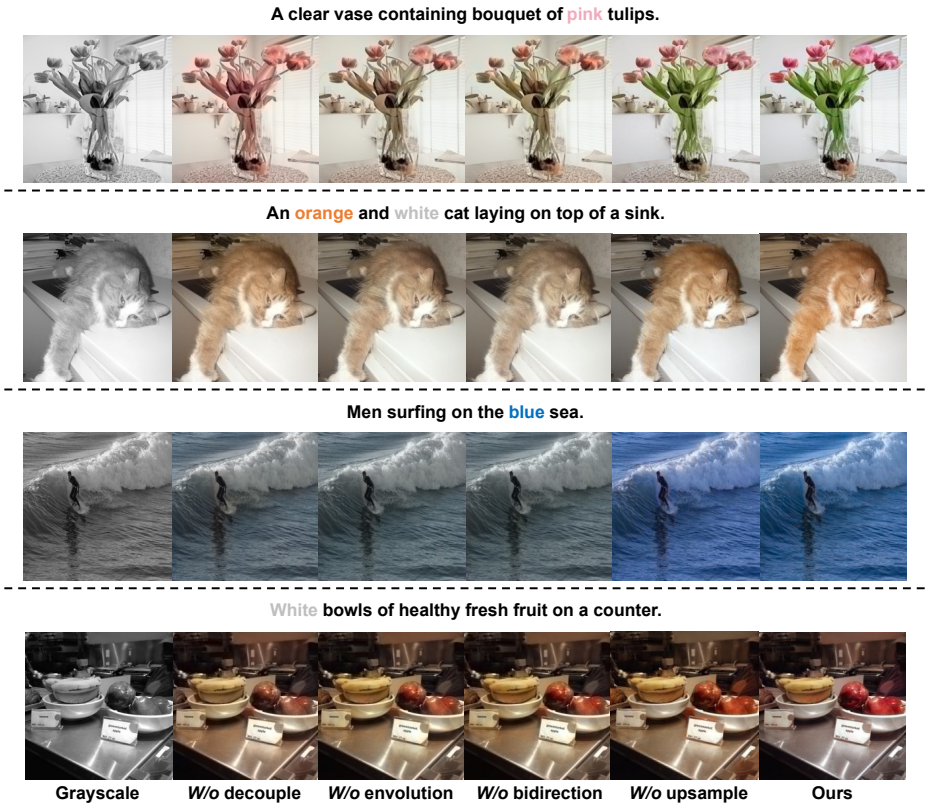

Fig. 7. Ablation study with different variants of the proposed method. **First row**: Without decoupling, the stem of flower is colorized with the pink color that describes the petals. **Second row**: Disabling evolution, the cat becomes grayish. **Third row**: Removing bidirection, the saturation of blue sea decreases. **Fourth row**: After removing upsampling layers, the red color of left apple is presented as a square patch.

In each experiment, there are 100 images randomly selected from the testing set. Experiments are published on Amazon Mechanical Turk (AMT) and each experiment is completed by 25 participants. As shown in Tab. 2, our method achieves the highest scores in both experiments.

4.3 Ablation Study

We disable various modules to create four baselines to study the impact of our proposed modules. The evaluation scores and synthetic images of the ablation study are shown in Tab. 1 and Fig. 7.

W/o decouple. We remove the decoupling module in decoupled tokens encoder, replace decoupling attention with conventional cross attention, and take out all the relevant modules about OCCM. In this way, we meet color-object mismatch problem mentioned in L-CoDe [25] (first row in Fig. 7).

Fig. 8. Examples of colorizing legacy photos with user-given captions.

W/o evolution. We use fixed decoupled tokens as language conditions and inject them into every decoupling block. As the result, the colorized image becomes grayish and color representation becomes inaccurate (second row in Fig. 7).
W/o bidirection. We modify the decoupling attention by controlling information flow only from decoupled language tokens to image tokens. This strategy prevents language tokens from capturing image semantics, and further interferes with the evolution of language tokens. The colorization results are undersaturated without bidirectional interaction between tokens (third row in Fig. 7).
W/o upsample. We remove the upsampling layers that are composed of a series of convolutions. Instead, a matrix multiplication is performed to project image tokens into pixels. Then, pixels are reshaped as 2D image. In this ablation, obvious patch artifacts occur (fourth row in Fig. 7).

4.4 Application

We show our application on colorizing legacy black and white photos under the guidance of language in Fig. 8.

5 Conclusion

We propose L-CoDer to deal with language-based colorization task. We introduce transformer into the task for three advantages: *(i)* The input image and language could be unified in one modality as tokens, which narrows the gap between modalities and helps the model understand intrinsic color property behind the word. *(ii)* The semantic of decoupled language tokens is organized coarse-to-fine evolving with image feature, which makes color representation more accurate. *(iii)* The global receptive field of transformer makes our method locally robust, which further improves the colorization quality. We conduct our experiments on the extended COCO-Stuff dataset, and our method achieves significantly higher scores than other compared methods in PSNR, SSIM, and LPIPS metrics.
Limitation. The transformer architecture also brings drawbacks, *e.g.*, when our method is expanded to train the high resolution version, it will require significantly more computing resources and longer training time than CNN-based methods. These could be improved with the development of transformer.
Acknowledgements. This project is supported by National Natural Science Foundation of China under Grant No. 62136001.

References

1. Antic, J.: A deep learning based project for colorizing and restoring old images (and video!), https://github.com/jantic/DeOldify 10, 11
2. Ardizzone, L., Lüth, C., Kruse, J., Rother, C., Köthe, U.: Guided image generation with conditional invertible neural networks. arXiv preprint arXiv:1907.02392 (2019) 3
3. Chen, J., Shen, Y., Gao, J., Liu, J., Liu, X.: Language-based image editing with recurrent attentive models. In: CVPR (2018) 2, 3, 10, 11
4. Cheng, Z., Yang, Q., Sheng, B.: Deep colorization. In: ICCV (2015) 3
5. Deshpande, A., Lu, J., Yeh, M.C., Jin Chong, M., Forsyth, D.: Learning diverse image colorization. In: CVPR (2017) 3
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL (2019) 6
7. Ding, H., Liu, C., Wang, S., Jiang, X.: Vision-language transformer and query generation for referring segmentation. In: ICCV (2021) 4, 8
8. Ding, M., Yang, Z., Hong, W., Zheng, W., Zhou, C., Yin, D., Lin, J., Zou, X., Shao, Z., Yang, H., et al.: Cogview: Mastering text-to-image generation via transformers. In: NIPS (2021) 4
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021) 4, 6
10. He, M., Chen, D., Liao, J., Sander, P.V., Yuan, L.: Deep exemplar-based colorization. ACM TOG (2018) 1
11. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of psnr in image/video quality assessment. Electronics letters (2008) 10
12. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. ACM ToG (2016) 3
13. Kim, W., Son, B., Kim, I.: Vilt: Vision-and-language transformer without convolution or region supervision. In: ICML (2021) 4, 6
14. Kumar, M., Weissenborn, D., Kalchbrenner, N.: Colorization transformer. In: ICLR (2021) 3, 4
15. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. In: ECCV (2016) 3
16. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: ICCV (2021) 4, 8
17. Liu, R., Deng, H., Huang, Y., Shi, X., Lu, L., Sun, W., Wang, X., Dai, J., Li, H.: Fuseformer: Fusing fine-grained information in transformers for video inpainting. In: ICCV (2021) 4
18. Lu, P., Yu, J., Peng, X., Zhao, Z., Wang, X.: Gray2colornet: Transfer more colors from reference image. In: ACM MM (2020) 1
19. Manjunatha, V., Iyyer, M., Boyd-Graber, J., Davis, L.: Learning to color from language. In: NAACL (2018) 2, 3, 10, 11
20. Sangkloy, P., Lu, J., Fang, C., Yu, F., Hays, J.: Scribbler: Controlling deep image synthesis with sketch and color. In: CVPR (2017) 1
21. Su, J.W., Chu, H.K., Huang, J.B.: Instance-aware image colorization. In: CVPR (2020) 1, 3, 10, 11
22. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NIPS (2017) 4

23. Vitoria, P., Raad, L., Ballester, C.: Chromagan: Adversarial picture colorization with semantic class distribution. In: WACV (2020) 1, 3, 10, 11
24. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. TIP (2004) 10
25. Weng, S., Wu, H., Chang, Z.C., Tang, J., Li, S., Shi, B.: L-code: Language-based colorization using color-object decoupled conditions. In: AAAI (2022) 2, 3, 4, 5, 6, 9, 10, 11, 12, 13
26. Wu, Y., Wang, X., Li, Y., Zhang, H., Zhao, X., Shan, Y.: Towards vivid and diverse image colorization with generative color prior. In: ICCV (2021) 3
27. Xie, Y.: Language-guided image colorization. Master's thesis, ETH Zurich, Departement of Computer Science (2018) 2, 3, 10, 11
28. Xu, Z., Wang, T., Fang, F., Sheng, Y., Zhang, G.: Stylization-based architecture for fast deep exemplar colorization. In: CVPR (2020) 1
29. Zhang, H., Koh, J.Y., Baldridge, J., Lee, H., Yang, Y.: Cross-modal contrastive learning for text-to-image generation. In: CVPR (2021) 4
30. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV (2016) 1, 3, 10, 11
31. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018) 10
32. 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 TOG (2017) 1
33. Zhao, J., Liu, L., Snoek, C.G., Han, J., Shao, L.: Pixel-level semantics guided image colorization. In: BMVC (2018) 3
34. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: CVPR (2021) 4
35. Zhou, Y., Ren, T., Zhu, C., Sun, X., Liu, J., Ding, X., Xu, M., Ji, R.: Trar: Routing the attention spans in transformer for visual question answering. In: ICCV (2021) 4
36. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. In: ICLR (2020) 4