# Conditional Image Repainting

Shuchen Weng and Boxin Shi*, *Senior Member, IEEE*

**Abstract**—A number of advanced image editing technologies have demonstrated impressive performance in synthesizing visually pleasing results in accordance with user instructions. In this paper, we further extend the practicalities of image editing technology by proposing the conditional image repainting (CIR) task, which requires the model to synthesize realistic visual content based on multiple cross-modality conditions provided by the user. We first define condition inputs and formulate two-phased CIR models as the baseline. After that, we further design unified CIR models with novel condition fusion modules to improve the performance. For allowing users to express their intent more freely, our CIR models support both attributes and language to represent colors of repainted visual content. We demonstrate the effectiveness of CIR models by collecting and processing four datasets. Finally, we present a number of practical application scenarios of CIR models to demonstrate its usability.

**Index Terms**—Image synthesis, image editing, cross-modality, generative adversarial networks

## 1 Introduction

IMAGE editing technologies have been widely explored to achieve texture [10], [23], [25], color [44], [56], [66], and object contour [30], [57], [62] manipulation for various application scenarios. Recently, it has been demonstrated that computer-assisted image editing model [40] could perform impressively to create a complex and realistic scene image under the guidance of cross-modality conditions. These works push the frontier of image editing and lower the skill barriers for non-professional users.

To further extend the flexibility of image editing technology, the concept of *conditional image repainting* (CIR) was proposed [45], [54], [55], where "repainting" means that the model is trained to generate some visual content that can be seamlessly composited with the original image in specific regions, and "conditional" means that the visual content should be generated under the guidance of several user-provided cross-modality conditions. As such, CIR models could "free" the users from professional skills while maintaining the "freedom" to realize their idea for editing an image (Fig. 1 (a)). Specifically, CIR models separately adopt the texture condition (random noise), the color condition (attribute or language, Fig. 1 (b)), and the geometry condition (parsing mask, Fig. 1 (c)) to describe the appearance of the repainted visual content, and the regions of the original image that have not been repainted are denoted as the background condition (RGB image, Fig. 1 (d)). Note that CIR models support both attributes [45], [55] and language [54] as the color condition to edit images with different properties: *(i)* For image regions that are clearly divided (*e.g.*, the man's upper and pants), CIR models adopt the attribute to directly construct the correspondences between the color condition and image regions (Fig. 1 top row). *(ii)* For images that have continuous structure or complex appearance (*e.g.*, the colorful bird), CIR models could instead take the language as the color condition to enable higher descriptive flexibility (Fig. 1 bottom row).

The CIR models can be categorized according to different architectures and conditions used. The first CIR model [55] focusing on synthesizing person images, formulates the CIR task to design the *two-phased architecture* and adopts *attributes* as the color condition, denoted as **TP+A** (Fig. 1 (e) top row). The two phases are *(i)* generation phase: generating the fine-grained visual content following conditions that describe the appearance as precisely as possible; *(ii)* composition phase: adjusting the color tone of the generated visual content adaptively towards the background condition to harmonize the repainted image. Another two-phased CIR model [54] introduces the *language* as a user-friendly option to represent the color condition, denoted as **TP+L** (Fig. 1 (e) bottom row). Since the diverse language representation makes it infeasible to predefine the correspondence between image regions and color conditions in advance, this model designs the SEmantic-BridgE (SEBE) module to provide an approximated estimate. Despite that two-phased CIR models [54], [55] could produce reasonable results, there remain several issues: *(i)* They assume the dependency of two phases: the compositing model could perform only after the generation phase, which limits its "play space" and reduces synthetic quality. *(ii)* They directly discard the meaningless background generated in the generation phase, which makes the variance of the stochastic gradient among samples large and causes low convergence. To address these issues, the recently proposed CIR model adopts the *unified architecture* by designing advanced modules to take the background condition into account when generating the visual content [45]. It also uses the attribute color condition, and we denoted it as **UF+A** (Fig. 1 (f) top row).

In this paper, we extend the UF+A CIR model to support the language color condition (Fig. 1 (f) bottom row), which enables users to flexibly represent an unlimited number of color combinations without having to specify predefined correspondences between image regions and the color condition. Additionally, it takes advantages of unified CIR models, which break the two-phased dependency limitation

- * *Corresponding author (shiboxin@pku.edu.cn).*
- *S. Weng and B. Shi are with the National Key Laboratory for Multimedia Information Processing and National Engineering Research Center of Visual Technology, School of Computer Science, Peking University, Beijing 100871, China.*

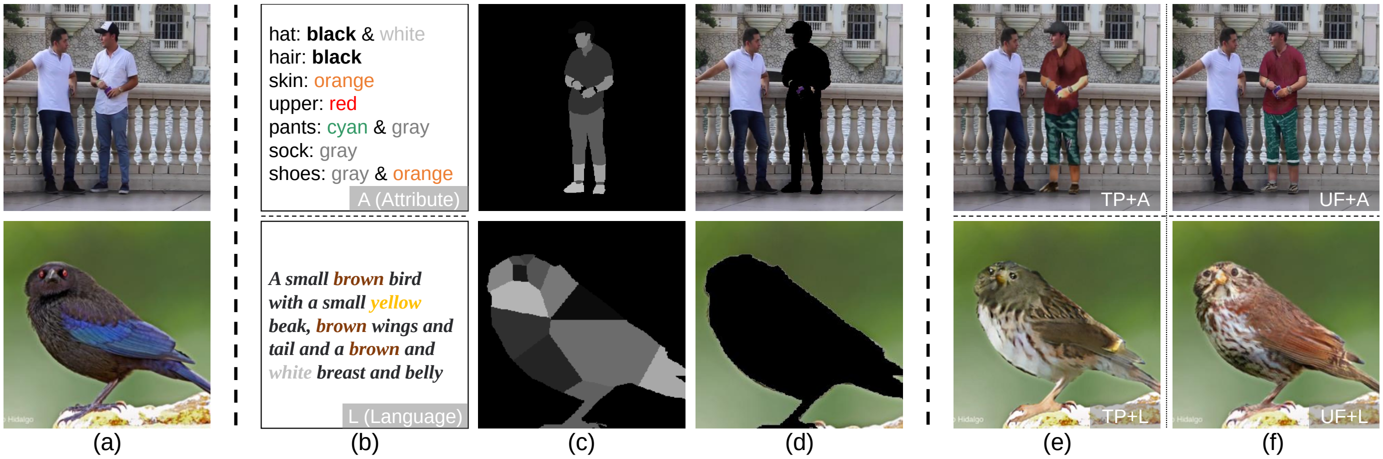


Fig. 1. Illustration of the CIR task. **Left:** The original images (a) that are required to be repainted. **Middle:** The user-provided multiple cross-modality conditions, including (b) attributes (A)/language (L) color conditions in the top/bottom row, (c) the geometry condition, and (d) the background condition. The texture condition (random noise) is omitted here. **Right:** The repainted results by two framework: (e) two-phased (TP) CIR models and (f) unified (UF) CIR models.

and have a smaller variance of stochastic gradients among samples, so that images could be repainted with higher synthetic quality, better condition consistency, and more realistic compositing effect. We denote this model as **UF+L**. Specifically, it is achieved via two improvements to the condition fusion module in the UF+A CIR model: *(i)* We design Layout Alignment attention (LA attention) mechanism to decouple the language color condition into adjective-noun pairs and further estimate the reliable correspondence between image features and the language color condition. *(ii)* We propose Color Semantic Smoother (CSS) to maintain the color distribution of estimated correspondences in every condition fusion module spatially similar, which enhances the internal edges of visual content and prevents color bleeding. Compared to TP+L, we apply the unified architecture and redesigning its novel condition fusion module by taking the background condition into account.

The TP+L CIR model [54] collects bird images and stuff images to evaluate the performance in repainting specified objects and outdoor scenarios with the language color condition. However, the stuff dataset has limited numbers of samples (7K, compared to 12K in the bird dataset) and categories (9, compared to 200 in the bird dataset). This motivates us to further expand the stuff dataset with more abundant images and diverse categories in the wild to avoid CIR models with the language color condition overfitting in typical scenarios. Specifically, the expanded stuff dataset includes 12K samples and 33 categories, each with handcrafted segmentation annotation and 10 captions that are washed out of unnecessary description about texture or geometry and only describe colors.

This paper is an extension of our preliminary works [55], [54], [45] published at CVPR 2020, ECCV 2020, and CVPR 2022 that introduced TP+A, TP+L, and UF+A CIR models, respectively. By completing the last remaining quadrant in Fig. 1 (f) bottom row, this paper has made the following contributions:

- We extend the unified architecture to support the language color condition, which provides higher descriptive flexibility to edit sophisticated structure and appearance.
- We expand the stuff dataset [54] in numbers of images and categories to comprehensively evaluate the performance of CIR models with the language color condition.
- We demonstrate a broad range of applications for CIR models to extended tasks such as damaged image restoration and image colorization.

## 2 Related Works

### 2.1 Conditional generative adversarial networks

The conditional generative adversarial network (cGANs) [29] is a conditional extension of the generative adversarial network [11], which is widely used in advanced image editing to synthesize user-required results under additional conditions. Under the guidance of the language description, cGANs [4], [59], [64], [69] stack multiple generators in sequence to generate images in a coarse-to-fine manner. Additionally, cGANs [5], [21], [28] could edit image with higher controllability and practicability by using the skeleton, reference image or attribute as the conditions. cGANs [15], [32], [53] could also reversibly transfer the domain of image-like data according to predefined rules. In this paper, we adopt multiple cross-modality conditions as the guidance to further exploit the potential of cGANs.

### 2.2 Condition injection

In addition to traditional operators that concatenate or add conditions directly to image features, novel condition injection modules have been developed for a variety of condition representations and modalities. Adaptive Instance Normalization (AdaIN) [14] is best known for image style transfer and it is widely used in vector condition fusion, *e.g.*, StyleGAN [16] injects the latent code with it. SPatially-Adaptive DEnormalization (SPADE) [33] is proposed to inject image-like data (*e.g.*, the segmentation mask) to preserve semantic information in uniform or flat regions. SPADE is further used as a basis for the design of advanced spatial tensor injection modules, *e.g.*, Affine Combination Module (ACM) module in ManiGAN [19]. Feature-wise Linear Modulation (FiLM) [37] has been demonstrated effective in visual reasoning tasks for sequence data, which shows generalization on language based colorization task [26]. Semantic Region-Adaptive Normalization (SEAN) [70] is designed to control both image semantic layout and the style of each semantic region. On the basis of these previous works, we explore condition injection module to separately control texture, color, geometry, and background conditions.

### 2.3 Cross-modality attention

The cross-modality attention bridges the gap between the data from various modalities. The traditional cross-modality attention modules [39], [47], [59], [60] project data from different modalities into the same latent space and estimate the relevance between them, which are trained in a data-driven fashion and supervised by a cross-modality retrieval [59] or reconstruction loss [39]. Recent cross-modality attention modules consider the semantic of the specific condition. Structure-driven attention [57] is an improvement on contextual attention [61], which avoids inaccurate attention weights using the structure information from conditional modality cues. DAE-GAN [43] extracts the aspect-level features from sentence and integrates them with image features to update and enhance word-level features. Object-driven attention [20] uses the word embedding of the class label obtained from the detection model to construct a word context vector by querying words in the sentence. XMC-GAN [63] combines the cross-modality attention and the condition injection module as the self-modulation layer, which calculates word-context vectors as modulation parameters and injects them via the condition injection approach. In addition to these works, we further explore the appropriate cross-modality attention mechanism in the CIR task.

### 2.4 Image composition

Traditional image composition methods are based on low-level image features, *e.g.*, global color distributions [6], image gradient domain [38], and multi-resolution pyramid with matching strategy [46]. Early deep-learning based methods adjust color tone via handcraft features of images [50], segmentation mask guidance [51], and deep image features learned via GANs [2], [48], [58]. Recently, delicate modules are designed to model the difference between foreground and background, *e.g.*, spatial-separated attention [7], region-aware adaptive instance normalization [24], and illumination and reflection decomposition [12]. It is essential for the CIR task to design a proper composition module so that the generated visual content could be seamlessly composited with the background condition.

## 3 Definition of condition inputs

As shown in Fig. 1, there are four cross-modality conditions used in the CIR task and each of them represents different semantics, including:

- **Texture condition** is adopted as the Gaussian noise $z \sim \mathcal{N}(0,1)$ because of two unique properties: *(i)* It is free of annotation; and *(ii)* it could provide high variety in generating textures of repainted visual content since the latent space of the Gaussian noise is continuous.
- **Color condition** $x^{\mathrm{c}}$ have two representations: *(i)* We use **attributes** $x^{\mathrm{ca}} \in \mathbb{L}^{N_{\mathrm{v}} \times N_{\mathrm{c}}}$ as the color condition when regions of repainted image could be clearly divided (Fig. 1 (b) top row), where each attribute describes a visual region (*e.g.*, hat, pants, and shoes), and values of the attribute mean corresponding colors (*e.g.*, black, orange, and cyan). Here $\mathbb{L} \in \{0,1\}$, $N_{\mathrm{c}}$ and $N_{\mathrm{v}}$ denote the number of attributes and the number of values, respectively; *(ii)* we can also use the **language** description $x^{\mathrm{cl}} \in \mathbb{R}^{N_{\mathrm{l}}}$ for images that have continuous structure or complex appearance (Fig. 1 (b) bottom row), where $N_{\mathrm{l}}$ presents the sentence length of the language.
- **Geometry condition** is the parsing mask $x^{\mathrm{g}} \in \mathbb{L}^{N_{\mathrm{g}} \times \tilde{H} \times \tilde{W}}$ (Fig. 1 (c)), which divides the repainted regions as multiple part regions (*e.g.*, left hand, right leg, and upper). Here $N_{\mathrm{g}}$, $\tilde{H}$, and $\tilde{W}$ denote the number of part regions, height, and width of the repainted image, respectively.
- **Background condition** $y^{\mathrm{b}} \in \mathcal{R}^{3 \times \tilde{H} \times \tilde{W}}$ is an RGB image that provides the reference color tone for the repainted visual content (Fig. 1 (d)). To distinguish the repainted regions and the background condition, the repainted binary mask $M^{\mathrm{r}} \in \mathcal{L}^{\tilde{H} \times \tilde{W}}$ could be obtained under the guidance of the geometry condition, which sets value 1 for repainted regions and 0 for elsewhere.

## 4 Two-phased CIR model

In this section, we first formulate two-phased CIR models and introduce their corresponding architectures, as shown in Fig. 2 (left). Then, we present designs to concrete visually-abstract conditions (*i.e.*, the texture condition and the color condition) under the guidance of the visually-concrete condition (*i.e.*, the geometry condition). Finally, we introduce optimization losses to constrain the conditional consistency.

### 4.1 Formulation of the two-phased CIR model

Two-phased CIR models include the generation phase and the compositing phase. In the **generation phase**, the visual content is synthesized under the guidance of conditions that describe the appearance, *i.e.*, $x^{\mathrm{g}}$, $x^{\mathrm{c}}$, and $z$. Based on the definitions in Sec. 3, we could further formulate the generation phase as:

$$\hat{y}^{\mathrm{r}} = F^{\mathrm{G}}(z, x^{\mathrm{c}}, x^{\mathrm{g}}), \tag{1}$$

where $F^{\mathrm{G}}$ is the conditional generator, and $\hat{y}^{\mathrm{r}} \in \mathbb{R}^{3 \times H \times W}$ is the synthesized visual content.

In the **compositing phase**, the compositing model $F^{\mathrm{C}}$ firstly estimates color tone parameters $(\rho, \tau)$ based on $\hat{y}^{\mathrm{r}}$ and the background condition $y^{\mathrm{b}} \in \mathbb{R}^{3 \times H \times W}$ as:

$$(\rho, \tau) = F^{\mathrm{C}}(\hat{y}^{\mathrm{r}}, y^{\mathrm{b}}). \tag{2}$$

After that, an affine transformation is performed to adjust the color tone of $\hat{y}^{\mathrm{r}}$ towards $y^{\mathrm{b}}$ as:

$$y^{\mathrm{r}} = \tanh(\rho \odot \hat{y}^{\mathrm{r}} \oplus \tau), \tag{3}$$

where $y^{\mathrm{r}}$ is the adjusted visual content, $\odot$ and $\oplus$ are the element-wise multiplication and the addition, respectively. Finally, $y^{\mathrm{r}}$ is combined with to synthesize the conditional repainted image $\hat{y}$ as:

$$\hat{y} = M^{\mathrm{r}} \odot y^{\mathrm{r}} + (1 - M^{\mathrm{r}}) \odot y^{\mathrm{b}}, \tag{4}$$

where $M^{\mathrm{r}}$ is the binary mask to distinguish the repainted regions and the background condition.

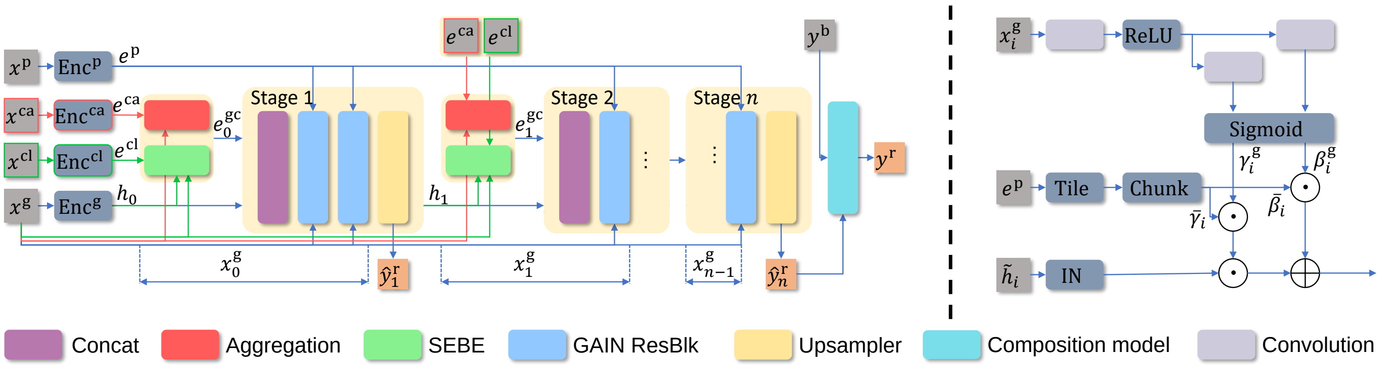


Fig. 2. Illustration of two-phased CIR models. There will be only one of the red or green blocks performed, which represent modules that are executed when the color condition is represented as attributes (TP+A) or language (TP+L), respectively. **Left:** The architecture of $n$-staged conditional generator. Only after the visual content is completely synthesized, the composition model adjusts their contrast and brightness. **Right:** The detailed design of Geometry-guided Adaptive Instance Normalization (GAIN), where $\oplus$ and $\odot$ denote the addition and the element-wise multiplication, respectively.

### 4.2 Two-phased architecture design

As shown in Fig. 2 (left), three encoders are used for embedding conditions that describe the appearance: *(i)* **texture encoder** $\text{Enc}^{\text{p}}$ to output the texture embedding $e^{\text{p}} \in \mathbb{R}^{2K}$, where $K$ is the channel number; *(ii)* **color encoder** to produce the attribute feature $e^{\text{ca}} \in \mathbb{R}^{K\times N_{\text{c}}}$ with the attribute color encoder $\text{Enc}^{\text{ca}}$ or the language feature $e^{\text{cl}} \in \mathbb{R}^{K\times N_{\text{l}}}$ with the language color encoder $\text{Enc}^{\text{cl}}$; and *(iii)* **geometry encoder** $\text{Enc}^{\text{g}}$ to generate the initial layout of repainted regions $h_0 \in \mathbb{R}^{K\times H_0\times W_0}$, where $H_0$ and $W_0$ are initial spatial resolution. With these encoded condition features, the architecture of conditional generator is designed to be $n$-staged to synthesize the visual content step by step, where each stage is composed of a concatenation layer, two GAIN ResBlks (details will be introduced in Sec. 4.5), and an upsampler. We could formulate the $n$-staged generation process as:

$$h_{i+1} = F_{i+1}^{\text{G}}(h_i, x_i^{\text{g}}, e^{\text{p}}, e_i^{\text{gc}}), \qquad i \in \{0, \ldots, n-1\} \tag{5}$$

where $F_{i+1}^{\text{G}}$ denotes the subnetwork of $F^{\text{G}}$ in the $(i+1)$-th stage, $h_i$ means the intermediate feature map produced by $F_{i+1}^{\text{G}}$, $x_i^{\text{g}}$ is the resized geometry condition that has the same spatial resolution of $h_i$, and $e_i^{\text{gc}}$ is the spatially-specific color condition calculated based on $x_i^{\text{g}}$ and the color condition ($e^{\text{ca}}$ or $e^{\text{cl}}$, details will be introduced in Sec. 4.3 and Sec. 4.4). At the end of each stage, the output $h_i$ is projected by a convolution block to generate an intermediate visual content $\hat{y}_i^{\text{r}}$, which is used to supervise the generation process. We set the stage number $n = 3$ following the previous work [59].

The composition model is designed based on the recently proposed pixel transformation method [2]. Given both the repainted visual content and the background condition, it estimates the contrast and brightness transformation parameters to adjust the color tone of repainted regions. Additionally, a two-fold improvement has been made to address limitations of this spatially-invariant transformation in the CIR task: *(i)* using the tanh function to confine the value range of adjusted visual content in $[-1, 1]$ and avoid the over-saturation issue; and *(ii)* predicting spatially-adaptive color tone parameters for each repainted region pixel, *i.e.*, $\rho, \tau \in \mathbb{R}^{3\times H\times W}$ rather than a single set of shared transformation parameters to avoid gradient vanishing.

Two discriminators are used to guide the optimization: *(i)* joint-conditional-unconditional patch discriminators [20] to make repainted images realistic and conditional consistency; *(ii)* a three-layer convolutional neural network as a compositing discriminator to determine whether the color tone of repainted regions are indistinguishable from the background.

### 4.3 Attribute color representation

the TP+A CIR model adopts attributes [55] as the color condition when image regions could be clearly divided (Fig. 1 (e) top row) so that they could directly predefine the correspondences between the color condition and image regions under the guidance of the geometry condition. We denote correspondences as $A \in \mathbb{L}^{N_{\text{c}}\times N_{\text{g}}}$ s.t. $\forall i, j\ A[i,j] \geq 0$, $\forall j\ \ \sum_i A[i,j] = 1$ (see Sec. 3 for definition of $x_{\text{c}}$ and $x_{\text{g}}$). Based on predefined correspondences, the TP+A CIR model could calculate the spatially-specific color condition $e_i^{\text{gc}} \in \mathbb{R}^{K\times\tilde{H}\times\tilde{W}}$ as:

$$e_i^{\text{gc}} = e^{\text{ca}} A \otimes x_i^{\text{g}}, \tag{6}$$

where $\otimes$ is the broadcast operator that aggregates attribute features for corresponding image regions. We name this process as **Aggregation**.

To guarantee the meaningfulness of attribute features, the TP+A CIR model adopts the cross-modality similarity model (CMSM) [59] to pretrain the attribute color encoder $\text{Enc}^{\text{ca}}$ and replace the attention mechanism by predefined correspondences to achieve a better matching. Improved CMSM is further used to compute a cross-modality ranking loss $\mathcal{L}_{\text{CM}}$ to bridge the semantic gap between repainted images and the color condition as:

$$\mathcal{L}_{\text{CM}} = -\sum_{i}^{N_{\text{bz}}} \log\Big(P(C_i|I_i)P(I_i|C_i)\Big), \tag{7}$$

where $N_{\text{bz}}$ represents the sample number in the batch, $P(\cdot|\cdot)$ means the posterior probability that conditions are matched, $C_i$ and $I_i$ mean the $i$-th color condition and synthesized image in the training batch, respectively.

### 4.4 Language color representation

The TP+L CIR model uses the language [54] as the color condition for images that have continuous structure or complex appearance (Fig. 1 (e) bottom row). Though the language description offers greater descriptive flexibility, it requires an additional module to estimate correspondences between image regions and the color condition. Therefore, the TP+L CIR model designs the SEmantic-BridgE (**SEBE**) attention. Specifically, SEBE firstly estimates the relevance $\beta_{p,q}$ between the $p$-th image feature $r_p$ and the $q$-th word feature in the language color condition $e_q^{\mathrm{cl}}$ as:

$$\beta_{p,q} = \frac{\exp(s_{p,q})}{\sum_{k=1}^{N_1} \exp(s_{p,k})}, \quad s_{p,q} = (r_p)^\top \phi(e_p^{\mathrm{cl}}), \tag{8}$$

where $\phi(\cdot)$ is a linear layer. Next, denoting $\hat{e}_p^+$ and $\hat{e}_q$ as the GloVe embedding [36] of the part name in the $p$-th image feature (acquired from the resized geometry condition $x_i^{\mathrm{g}}$) and the $q$-th word in the language color condition $x^{\mathrm{cl}}$, SEBE additionally calculates the auxiliary cosine similarity between them as $s_{p,q}^{\mathrm{SEBE}} \in [-1, 1]$. After that, the similarity is shifted to be non-negative and normalized by L1 Norm. This process could be formulated as:

$$\beta_{p,q}^{\mathrm{SEBE}} = \frac{s_{p,q}^{\mathrm{SEBE}} + 1}{\sum_{k=1}^{N_1} (s_{p,k}^{\mathrm{SEBE}} + 1)}, \quad s_{p,q}^{\mathrm{SEBE}} = \frac{(\hat{e}_p^+)^\top \hat{e}_q}{\|\hat{e}_p^+\| \|\hat{e}_q\|}, \tag{9}$$

where $\beta_{p,q}^{\mathrm{SEBE}} \in [0, 1]$ is the calculated auxiliary attention weight. As such, the perfect image-language matching is established when $\hat{e}_p^+$ and $\hat{e}_q$ are embedding vectors of the same word (*e.g.*, beak, breast, and belly in Fig. 1 (b) bottom row), where $\beta_{p,q}^{\mathrm{SEBE}}$ is able to stay constantly as 1.

As the philosophy of "loudness is persuasive", SEBE employs the "maximum" as the mixer of attention weights to perform the attention selection and obtains the color feature $c_p^{\mathrm{SEBE}} \in \mathbb{R}^K$ in the $p$-th image region as:

$$c_p^{\mathrm{SEBE}} = \sum_{q=1}^{N_1} \max(\beta_{p,q}^{\mathrm{SEBE}}, \beta_{p,q}) \phi(e_q^{\mathrm{cl}}). \tag{10}$$

Finally, the spatially-specific color condition $e_i^{\mathrm{gc}}$ could be obtained by concatenating all color features $c_p^{\mathrm{SEBE}}$.

Similar to the TP+A CIR model, the language color encoder $\mathrm{Enc}^{\mathrm{cl}}$ is pretrained with CMSM. Since correspondences cannot be predefined, the cross-modality ranking loss $\mathcal{L}_{\mathrm{CM}}$ with attention mechanism is applied.

### 4.5 Condition injection module

In two-phased CIR models, conditions are injected in two ways: *(i)* Considering the spatially-specific color condition $e_i^{\mathrm{gc}}$ presents the color distribution on the image plane, two-phased CIR models simply concatenate it with the input intermediate feature map $h_i$ before the subnetwork of the conditional generator $F_{i+1}^{\mathrm{G}}$ in each stage. *(ii)* The Geometry-guided Adaptive Instance Normalization Residual BLocK (**GAIN ResBlk**) is proposed to inject the texture condition. The GAIN ResBlk consists of a residual connection, several convolutions, and two GAIN modules. We present the structure of the GAIN module in Fig. 2 (right). Specifically, a shallow convolution block projects $x_i^{\mathrm{g}}$ to be geometry affine parameters $\gamma_i^{\mathrm{g}} \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$ and $\beta_i^{\mathrm{g}} \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$. After that, texture affine parameters are inferred via chunking the texture embedding $e^{\mathrm{p}} \in \mathbb{R}^{2K}$ into slope $\gamma^{\mathrm{p}} \in \mathbb{R}^K$ and bias $\beta^{\mathrm{p}} \in \mathbb{R}^K$, and then are tiled to be $\bar{\gamma}_i \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$ and $\bar{\beta}_i \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$. Finally, texture affine parameters are fused with the geometry affine parameters via element-wise multiplication to make regions within the same part have similar textures, which are used to modulate the hidden feature maps $\tilde{h}_i \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$ as:

$$\tilde{h}_i' = (\gamma_i^{\mathrm{g}} \odot \bar{\gamma}_i) \odot \mathrm{IN}(\tilde{h}_i) \oplus (\beta_i^{\mathrm{g}} \odot \bar{\beta}_i), \tag{11}$$

where $\mathrm{IN}(\cdot)$ represents the instance normalization, and $\odot$ is the element-wise multiplication.

### 4.6 Learning

A set of joint-conditional-unconditional patch discriminators [20] are introduced to make repainted images realistic and conditional consistency, as $D^{\mathrm{I}} = \{(D_1^{\mathrm{u}}, D_1^{\mathrm{c}}), \ldots, (D_n^{\mathrm{u}}, D_n^{\mathrm{c}})\}$, where $D_i^{\mathrm{u}}$ and $D_i^{\mathrm{c}}$ represent the unconditional and conditional discriminator, respectively. Given a pair of intermediate visual content and spatially-specific color condition, $D_i^{\mathrm{u}}$ and $D_i^{\mathrm{c}}$ are required to estimate the probability of being realistic and conditional consistency for each patch, which could be formulated as:

$$\begin{aligned} \mathcal{L}_{\mathrm{g}}^{\mathrm{m}} = \sum_{i=1}^{n} \frac{1}{2N_i^{\mathrm{pat}}} \sum_{j}^{N_i^{\mathrm{pat}}} \lambda^{\mathrm{u}} (\log D_i^{\mathrm{u}}(\hat{y}_i^{\mathrm{r}})_j + \log(1 - D_i^{\mathrm{u}}(y_i^{\mathrm{gt}})_j) \\ + \log D_i^{\mathrm{c}}(\hat{y}_i^{\mathrm{r}}, e_i^{\mathrm{gc}})_j + \log(1 - D_i^{\mathrm{c}}(y_i^{\mathrm{gt}}, e_i^{\mathrm{gc}})_j), \end{aligned} \tag{12}$$

where $\lambda^{\mathrm{u}}$ is a balancing hyperparameter, $N_i^{\mathrm{pat}}$ is the pixel number of intermediate visual content $\hat{y}_i^{\mathrm{r}}$, and $y_i^{\mathrm{gt}}$ is the resized ground truth image. The last discriminators $D_n^{\mathrm{u}}$ and $D_n^{\mathrm{c}}$ are further adopted for the adjusted visual content $y^{\mathrm{r}}$ as:

$$\begin{aligned} \mathcal{L}_{\mathrm{g}}^{\mathrm{r}} = & \frac{1}{2N_n^{\mathrm{pat}}} \sum_{j=1}^{N_n^{\mathrm{pat}}} \lambda^{\mathrm{u}} (\log D_n^{\mathrm{u}}(y^{\mathrm{r}})_j + \log(1 - D_n^{\mathrm{u}}(y^{\mathrm{gt}})_j)) \\ & + \log D_n^{\mathrm{c}}(y^{\mathrm{r}}, e_n^{\mathrm{gc}})_j + \log(1 - D_n^{\mathrm{c}}(y^{\mathrm{gt}}, e_n^{\mathrm{gc}})_j). \end{aligned} \tag{13}$$

The compositing discriminator $D^{\mathrm{C}}$ is a three-layer neural network, designed to distinguish the adjusted repainted regions and the background regions as:

$$\mathcal{L}_{\mathrm{c}} = -\sum_{i=1}^{N^{\mathrm{pix}}} (\log(1 - D^{\mathrm{C}}(y^{\mathrm{r}})_i) + \log D^{\mathrm{C}}(y^{\mathrm{gt}})_i) \odot M^{\mathrm{r}}, \tag{14}$$

where $N_{\mathrm{pix}}$ denotes the pixel number, $M^{\mathrm{r}}$ is the binary mask that sets 1 for repainted regions and 0 for otherwise

A pixel-wise loss $\mathcal{L}_{\mathrm{R}}$ is used to regularize the training, which anchors the repainted regions of images after the compositing phase to that before the compositing phase as:

$$\mathcal{L}_{\mathrm{R}}(F^{\mathrm{C}}) = \|(y^{\mathrm{r}} - \hat{y}^{\mathrm{r}}) \odot M_{\mathrm{r}}\|_1. \tag{15}$$

$\mathcal{L}_{\mathrm{CM}}$ is a cross-modality ranking loss to make the repainted image consistent with the color condition, as described in Eq. (7).

Finally, the total loss is the weighted sum of losses mentioned above as:

$$\min_{D^{\mathrm{I}}, D^{\mathrm{C}}} \max_{F^{\mathrm{G}}, F^{\mathrm{C}}} \mathcal{L}_{\mathrm{g}}^{\mathrm{m}} + \mathcal{L}_{\mathrm{g}}^{\mathrm{r}} + \lambda_{\mathrm{CM}} \mathcal{L}_{\mathrm{CM}} + \lambda_{\mathrm{c}} \mathcal{L}_{\mathrm{c}} + \lambda_{\mathrm{R}} \mathcal{L}_{\mathrm{R}}, \tag{16}$$

where $\lambda_{\mathrm{CM}} = 20$, $\lambda_{\mathrm{c}} = 0.03$, $\lambda_{\mathrm{R}} = 1.0$, and $\lambda_{\mathrm{u}} = 4.0$, based on experiments using a held-out validation set.

## 5 Unified CIR model

The pipeline that explicitly divides the task into two phases as shown in Fig. 3 (a-c), has an inherent limitation: only after the first generation phase, the compositing model could adjust the color tone of repainted regions, which narrows the two-phase models' "play space" and leads to color tone gaps between repainted regions and the background condition. In addition, two-phased CIR models directly discard the generated meaningless background (*e.g.*, the purple area in Fig. 3 (b)), which masks the gradient of background region, makes the variance of the stochastic gradient among samples large, causes the low convergence, and finally reduces the quality of repainted images. As shown in Fig. 3 (d), the two-phased CIR model (TP+A) produces a reasonable result, but there are obvious color tone gaps between regions, due to its assumptions on the dependency between the generation phase and compositing phase.

In this section, we present the unified CIR models to solve these issues with novel architecture and condition injection modules. As shown in Fig. 3 (e), the well-adjusted color tone makes the repainted person appear more realistic. Additionally, we represent the redesigned training losses for unified CIR models.

### 5.1 Formulation of the unified CIR model

Unified CIR models are proposed to break the two-phased dependency limitation, which achieve higher synthetic quality, better condition consistency, and more realistic compositing effect. Unified CIR models discard the composition model and take the background condition into account when synthesizing the visual content so that their color tone could be adaptively adjusted. Following the definition in Sec. 3, the one-step generation process of the unified CIR model could be formulated to replace Eq. (1), Eq. (2), and Eq. (3) as:

$$y^{\mathrm{r}} = F^{\mathrm{G}}(z, x^{\mathrm{c}}, x^{\mathrm{g}}, y^{\mathrm{b}}). \tag{17}$$

Then, unified CIR models obtain the finally synthesized repainted image $\hat{y}$ by Eq. (4).

Similar to two-phased CIR models, unified CIR models also support both attributes and the language as the color condition, abbreviated as UF+A and UF+L. Novel condition fusion and injection modules have been designed for these two color representations.

### 5.2 Unified architecture model design

As show the unified architecture in Fig. 4 (left), unified CIR models separately use the texture encoder $\mathrm{Enc}^{\mathrm{p}}$, the color encoder $\mathrm{Enc}^{\mathrm{ca}}$ or $\mathrm{Enc}^{\mathrm{cl}}$, and the geometry encoder $\mathrm{Enc}^{\mathrm{g}}$ to embedding corresponding conditions, where the background condition $y^{\mathrm{b}}$ and the geometry condition $x^{\mathrm{g}}$ are separately resized as as $y_i^{\mathrm{b}}$ and $x_i^{\mathrm{g}}$ before fused with other conditions. Following the configuration of GauGAN [33], the conditional generator $F^{\mathrm{G}}$ of unified CIR model consists of a stack of Con-Fusion ResBlks (details will be introduced in Sec. 5.5) and Upsamplers. Compared to the $n$-staged conditional generator in Fig. 2, this one-step conditional generator synthesizes repainted images starting from a coarser resolution ($4 \times 4$ *vs.* $64 \times 64$) and performs more upsamplers (7 *vs.* 3), which allows the model to enrich the image details more gradually. This one-step generator also discards all intermediate visual contents $\hat{y}_i^{\mathrm{r}}$, which requires the model to design novel supervision signals to constrain the optimization process.

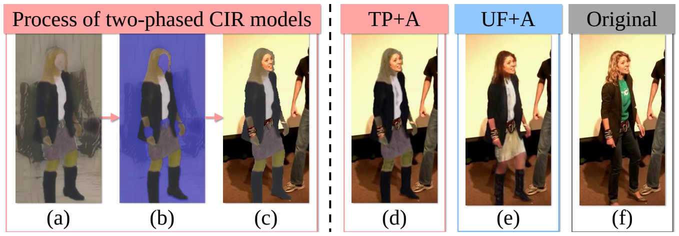


Fig. 3. Comparison between the two-phased CIR model (TP+A) and the unified CIR model (UF+A). **Left:** The process of two-phased CIR models: (a) the output of the generation phase; (b) the visualization of meaningless background with the purple mask; (c) the input of the compositing model. **Right:** (d), (e), and (f) represent the repainted results of the TP+A CIR model, the UF+A CIR model, and the original image, respectively.

Unified CIR models adopt three discriminators to achieve the adversarial training: *(i)* the last one of joint-conditional-unconditional patch discriminators [20] to guide the realistic and conditional consistency of the adjusted visual content $y^{\mathrm{r}}$; *(ii)* the same compositing discriminator as two-phased CIR models to push the color tone of repainted regions towards the background condition; *(iii)* the feature matching discriminator [53] to estimate the intermediate feature map and supervise the optimization.

### 5.3 Attribute color representation

The UF+A CIR model [45] is adopted for the image whose regions are clearly divided, as shown in Fig. 1 (f) top row. With predefined correspondences between image features and the color condition (defined in Sec. 4.3), the spatially-specific color condition $e_0^{\mathrm{gc}}$ could be directly calculated and inputted to the one-step conditional generator $F^{\mathrm{G}}$ to initial the semantic layout of repainted images.

Unified CIR models design hierarchical architectures for CMSM [59] to obtain the multi-grained features of synthesized images, which are used to pretrain the color encoder $\mathrm{Enc}^{\mathrm{ca}}$ to cover the lack of intermediate feature maps as supervision signals. Based on this design, both the image encoder and the color encoder extract $N_{\mathrm{m}}$ intermediate feature maps at different layers. When calculating the cross-modality ranking loss $\mathcal{L}_{\mathrm{HCM}}$, posterior probabilities of intermediate image feature being matching with the corresponding intermediate color feature at different encoder layers are summarized as:

$$\mathcal{L}_{\mathrm{HCM}} = -\sum_{i}^{N_{\mathrm{m}}} \sum_{j}^{N_{\mathrm{bz}}} \log \Big( P(C_{i,j}|I_{i,j}) P(I_{i,j}|C_{i,j}) \Big), \tag{18}$$

where $C_{i,j}$ and $I_{i,j}$ mean the $j$-th color condition and synthesized image in the training batch at the $i$-th layer of the encoder, respectively.

### 5.4 Language color representation

As clarified in Sec. 4.4, the UF+L CIR model also has to estimate multiple correspondences between image features

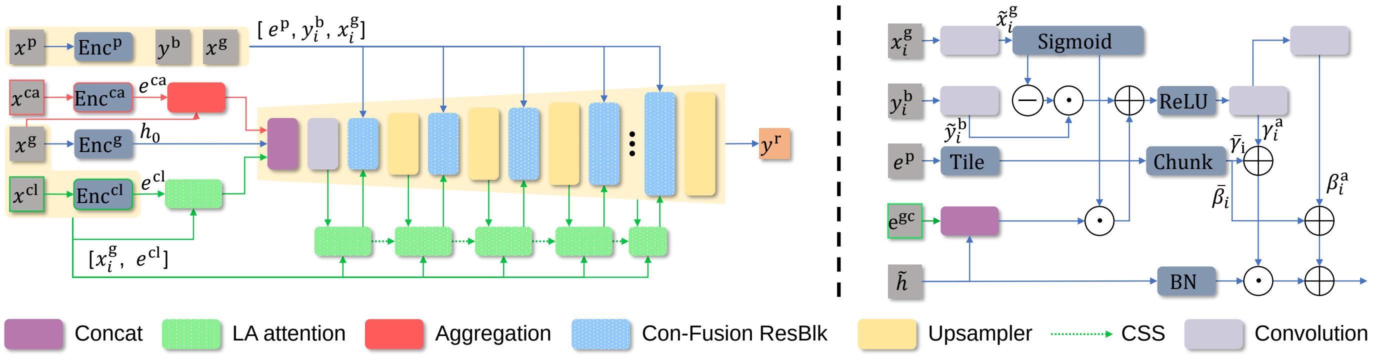


Fig. 4. Illustration of unified CIR models. There will be only one of the red or green blocks performed, which represent modules that are executed when the color condition is represented as attributes (UF+A) or language (UF+L), respectively. **Left:** The unified architecture, which takes the background condition into account so that it could adjust the color tone of repainted regions during the generation.**Right:** The detailed design of the Condition-Fusion (Con-Fusion) module, which constructs the relationship between the background condition and conditions that describe the appearance. $\oplus$, $\ominus$, and $\odot$ denote the addition, the function $\mathbb{1} - (\cdot)$, and the element-wise multiplication, respectively.

of different spatial resolution and the color condition for sophisticated images, *e.g.*, Fig. 1 (f) bottom row. As such, in addition to initializing the semantic layout, the language color condition should also be injected into intermediate layers in the conditional generator to supervise the generation process. However, it has been observed in the TP+L CIR model that excessive estimations (*i.e.*, more than 3) may cause training to become unstable and eventually result in mode collapse. To overcome this challenge and to meet the requirement for more correspondence estimations (*i.e.*, 7) in the UF+L CIR model, we newly propose the Layout Alignment attention (**LA attention**) and the Color Semantic Smoother (**CSS**) module.

The LA attention is designed to estimate the reliable spatially-specific color condition, which includes an estimation branch and a reference branch. The estimation branch estimates the relevance between image feature and word feature as Eq. (8), and aggregates color features to corresponding image regions as: $c_p^{\rm eb} = \sum_{q=1}^{N_{\rm l}} \beta_{p,q}\phi(e_q^{\rm cl})$. The reference branch provides the reliable color layout reference in three steps: *(i)* using the Stanford CoreNLP toolkit [27] to identify all adjectival modifiers about color and their corresponding nouns (or pronouns) in the language color condition; *(ii)* calculating the relevance between the GloVe embedding [36] of the part name that covers the image region (acquired from $x_i^{\rm g}$) and word in the language color condition; *(iii)* aggregating color features by placing the Glove embeddings [36] of adjectival modifiers about color to the image region whose part name is relevant with the corresponding nouns (or pronouns) as:

$$c_p^{\rm rb} = \sum_{q=1}^{N_{\rm rb}} \frac{\exp(s_{p,q}^{\rm LA})}{\sum_{k=1}^{N_{\rm rb}} \exp(s_{p,k}^{\rm LA})} (\hat{e}_q^{\rm a})^\top, \quad s_{p,q}^{\rm LA} = (\hat{e}_p^{+})^\top \hat{e}_q^{\rm n}, \tag{19}$$

where $\hat{e}_p^{+}$ is the GloVe embedding of part name that covers the $p$-th image region, $\hat{e}_q^{\rm a}$ and $\hat{e}_q^{\rm n}$ are separately the $q$-th adjectival modifier about color and corresponding nouns (or pronouns), and $N_{\rm rb}$ is the number of adjectival modifiers about color. After separately concatenating all $c_{\rm p}^{\rm eb}$ and $c_{\rm p}^{\rm rb}$, the LA attention calculates referenced spatially-specific color condition $\tilde{e}_i^{\rm gc}$ by fusing outputs of two branches via a 3 × 3 convolution layer.

Compared to SEBE module mentioned in Sec. 4.4, LA attention has two advantages: *(i)* It decouples the the language color condition into adjective-noun pairs so that only adjectival modifiers about color are correctly placed in the corresponding image region, which avoids the use of irrelevant words and further generates reliable reference layout; *(ii)* instead of using GloVe embeddings to estimate relevance, it uses aggregated GloVe embeddings as a reference to preserve the prior knowledge within a pretrained GloVe large model, which is further integrated by the following convolution layer. This facilitates the model to learn deeper semantics beyond the dataset.

CSS module is proposed to maintain a similar color distribution of color features across each spatially-specific color condition and further prevent color bleeding at the boundary of part regions. Specifically, based on previous coarser estimates, CSS calculates fine-grained spatially-specific color conditions as:

$$e_i^{\rm gc} = \frac{\lambda - 1}{\lambda^{i+1} - \lambda} \sum_{j=1}^{i} \lambda^j \ {\rm UP}(\tilde{e}_j^{\rm gc}), \tag{20}$$

where the hyperparameter $\lambda = 2$ represents the weight of previous estimations, and ${\rm UP}(\cdot)$ is an upsampler layer to increase the spatial resolution of previous estimations.

Similarly to the UF+A CIR model, CMSM and hierarchical cross-modality ranking loss $\mathcal{L}^{\rm HCM}$ are employed for pretraining the color encoder $\rm Enc^{cl}$ and supervising conditional consistency.

### 5.5 Condition injection module

In addition to adopting the geometry condition and spatially-specific color condition to initialize the semantic layout, the cross-modality conditions are also injected and fused in every novel Condition-Fusion Residual BLocK (**Con-Fusion ResBlk**), which consists of a residual connection, several convolution layers, and a Con-Fusion module.

As shown in Fig. 4 (right), the Con-Fusion module constructs the interaction and dependency relationship between the background condition and the other conditions so that unified CIR models could discard the compositing model while adjusting the color tone of the visual content.

Specifically, it separately extracts the feature of the resized background condition $y_i^{\mathrm{b}}$ and the geometry condition $x_i^{\mathrm{g}}$, denoted as $\tilde{y}_i^{\mathrm{b}}$ and $\tilde{x}_i^{\mathrm{g}}$ for the $i$-th Con-Fusion ResBlk. After that, the geometry feature is used as the gate to fuse the background feature and the hidden image feature $\tilde{h}_i$ as:

$$e^{\mathrm{a}} = \tilde{y}_i^{\mathrm{b}} \odot \mathrm{Sg}(\tilde{x}_i^{\mathrm{g}}) \oplus \tilde{h}_i \odot (\mathbb{1} - \mathrm{Sg}(\tilde{x}_i^{\mathrm{g}})), \tag{21}$$

where $e_i^{\mathrm{a}} \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$ is denoted as the appearance condition, $\mathrm{Sg}(\cdot)$ means the Sigmoid function, and $\mathbb{1}$ is the tensor of one. Next, the appearance condition is chunked to produce appearance parameters $\gamma^{\mathrm{a}} \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$ and $\beta^{\mathrm{a}} \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$ with a convolution layer. Similarly, the texture embedding is chunked and tiled as texture parameters $\bar{\gamma} \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$ and $\bar{\beta} \in \mathbb{R}^{K \times \tilde{H} \times \tilde{W}}$. These parameters are mixed through the addition operator and injected into the one-step conditional generator as:

$$\tilde{h}_i' = (\bar{\gamma}_i \oplus \gamma_i^{\mathrm{a}}) \odot \mathrm{BN}(\tilde{h}_i) \oplus (\bar{\beta}_i \oplus \beta_i^{\mathrm{a}}), \tag{22}$$

where $\mathrm{BN}(\cdot)$ means batch normalization, which could avoid the semantic information being washed away during generation [33]. Note that when taking language as the color condition, the estimated spatially-specific color condition will be concatenated with the hidden image feature as $[\tilde{h}_i; e_i^{\mathrm{gc}}]$ to replace $\tilde{h}_i$ in Eq. (21) so that it could adaptively guide the distribution of color features.

### 5.6 Learning

Since unified CIR models discard all the intermediate visual contents, only the last one ($D_n^{\mathrm{I}}$) of the joint-conditional-unconditional patch discriminators [20] is adopted to supervise repainted images to be realistic and condition-consistent, formulated the same as $\mathcal{L}_{\mathrm{g}}^{\mathrm{r}}$ in Eq. (13).

The feature matching loss [53] and perceptual loss [10] are introduced into unified CIR models to estimate the intermediate feature maps and cover the lack of intermediate visual content. The feature matching loss calculates the mean L1 distance of feature pairs extracted by feature discriminators:

$$\mathcal{L}_{\mathrm{FM}} = \sum_{i=1}^{T_{\mathrm{FM}}} \left\| D_i^{\mathrm{FM}}(y^{\mathrm{r}}) - D_i^{\mathrm{FM}}(y^{\mathrm{gt}}) \right\|_1, \tag{23}$$

where $D_i^{\mathrm{FM}}(\cdot)$ is the function to extract the feature map in the $i$-th layer of the feature discriminator, and $T_{\mathrm{FM}}$ is the layer number. And the perceptual loss is performed using a well-pretrained VGG network as:

$$\mathcal{L}_{\mathrm{p}} = \sum_{i}^{T_{\mathrm{p}}} \frac{1}{C_i H_i W_i} \left\| \varphi_i(y^{\mathrm{r}}) - \varphi_i(y^{\mathrm{gt}}) \right\|_2^2, \tag{24}$$

where $\varphi_i(\cdot)$ is the function to obtain feature in the $i$-th layer of the VGG network, $T_{\mathrm{p}}$ is the layer number, $C_i$, $H_i$, and $W_i$ are corresponding channel, height, and width, respectively.

The background L1 loss $\mathcal{L}_{\mathrm{b}}$ is designed to maintain the semantic of the background feature as:

$$\mathcal{L}_{\mathrm{b}} = \left\| (y^{\mathrm{r}} - y^{\mathrm{b}}) \odot (\mathbb{1} - M^{\mathrm{r}}) \right\|_1. \tag{25}$$

The compositing loss $\mathcal{L}_{\mathrm{c}}$ in Eq. (14) and cross-modality ranking loss $\mathcal{L}_{\mathrm{HCM}}$ in Eq. (18) are adopted to make repainted images more harmonious and bridge the semantic gap between the repainted image and the color condition, respectively.

Finally, we combine these losses as the total loss:

$$\begin{aligned} \min_{D^{\mathrm{I}}, D^{\mathrm{C}}, D^{\mathrm{FM}}} \max_{F^{\mathrm{G}}} \mathcal{L}_{\mathrm{g}}^{\mathrm{r}} + \lambda_{\mathrm{FM}} \mathcal{L}_{\mathrm{FM}} + \lambda_{\mathrm{p}} \mathcal{L}_{\mathrm{p}} + \lambda_b \mathcal{L}_{\mathrm{b}} \\ + \lambda_{\mathrm{c}} \mathcal{L}_{\mathrm{c}} + \lambda_{\mathrm{CM}} \mathcal{L}_{\mathrm{HCM}}, \end{aligned} \tag{26}$$

where $\lambda_{\mathrm{FM}} = 10.0$, $\lambda_{\mathrm{p}} = 10.0$, $\lambda_{\mathrm{b}} = 1.0$, $\lambda_{\mathrm{c}} = 0.03$, and $\lambda_{\mathrm{CM}} = 2.0$, based on experiments using a held-out validation set.

## 6 Datasets

We collect and process 4 datasets for different application scenarios of the CIR task, listed as follows:

- Person-CIR-A dataset [55] is built upon the Video Instance-level Parsing (VIP) person parsing dataset [68], which includes 404 videos with several people covering various scenarios, pixel-wise semantic part categories (*e.g.*, hair, pants, and socks), and instance-level identification. We further annotate colors for each semantic part as attributes and crop the images into sub-images with each containing one major person. The training and test splits are created with 42K and 6K samples, respectively.
- Landscape-CIR-A dataset [45] is collected from the Flickr website to concentrate on landscape generation, which contains 28K training images and 3K test images. We annotate the attribute color condition following the protocol of the Person-CIR-A dataset. We obtain the parsing mask by running the pretrained segmentation network DeepLabV2 [3]. Images are resized into $256 \times 256$ and $512 \times 512$ resolutions to evaluate the performance of landscape image repainting (objects covering a large area) and high-resolution image repainting.
- Bird-CIR-L dataset [54] is the CIR-extended version of Caltech UCSD Birds (CUB) [52] dataset. The original CUB dataset contains 11788 images covering 200 species, each with a coarse outline and 15 locations of the part (*e.g.*, beak, belly, and breast). To obtain the rough parsing mask, we label each pixel within the outline as the class of the nearest part location in Manhattan distance. Previous work [41] provides the language descriptions. Following AttnGAN [59], we split the dataset into 8.8K training samples and 2.9K testing samples.
- Stuff-CIR-L dataset focuses on repainting 33 common stuff (*e.g.*, sand, water, and dirt), including 10K training images and 2.2K testing images collected from the COCO-Stuff [1] dataset, each with a pixel-level segmentation mask. We reannotate each image with 10 captions and ensure that the color description appears in sentences and wash out unnecessary information about texture or geometry. This dataset is an extension of the stuff dataset in our preliminary work [54], where more stuff categories and complex images are included to increase the difficulty of repainting common stuff.

Among these, Person-CIR-A and Landscape-CIR-A datasets are provided with the attribute color condition, and Bird-CIR-L and Stuff-CIR-L datasets are annotated with the language color condition. They are separately used to evaluate CIR models with different representation of the color condition.

TABLE 1
Quantitative comparison experiments among unified CIR models, two-phased CIR models, and related methods on all 4 CIR datasets. Throughout the paper, ↑ (↓) indicates larger (smaller) values are better. The best performances are highlighted in **bold**.

| Dataset (attributes) | Score | Ours | | Comparison | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | UF+A | TP+A | CSC | TDANet-R | TDANet-F | SGENet-R | SGENet-F | Imagic |
| PERSON-CIR-A | FID ↓ | **11.45** | 16.09 | 18.82 | 44.64 | 23.97 | 39.25 | 15.58 | 44.34 |
| | R-prcn (%) ↑ | **88.71** | 85.22 | 84.88 | 42.28 | 66.57 | 40.57 | 62.07 | 17.73 |
| | M-score ↓ | **3.56** | 3.86 | 19.20 | 35.34 | 7.36 | 12.73 | 15.62 | 86.48 |
| LANDSCAPE-CIR-A (256 × 256) | FID ↓ | **9.96** | 18.12 | 17.70 | 52.94 | 34.63 | 34.79 | 19.96 | 32.05 |
| | R-prcn (%) ↑ | **93.04** | 85.43 | 83.95 | 74.35 | 83.84 | 67.39 | 68.08 | 74.28 |
| | M-score ↓ | **3.14** | 7.45 | 50.49 | 6.31 | 5.44 | 3.40 | 4.67 | 36.24 |
| LANDSCAPE-CIR-A (512 × 512) | FID ↓ | **18.63** | 26.97 | 21.47 | 59.07 | 26.85 | 30.99 | 24.24 | 35.47 |
| | R-prcn (%) ↑ | **91.07** | 75.33 | 73.65 | 75.45 | 86.47 | 56.93 | 67.79 | 77.34 |
| | M-score ↓ | **14.42** | 54.96 | 90.91 | 20.66 | 16.78 | 23.31 | 29.79 | 42.29 |

| Datasets (language) | Score | Ours | | Comparison | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | UF+L | TP+L | CSC | TDANet-R | TDANet-F | SGENet-R | SGENet-F | Imagic |
| BIRD-CIR-L | FID ↓ | **10.54** | 12.16 | 17.34 | 36.84 | 21.14 | 47.58 | 16.61 | 49.51 |
| | R-prcn (%) ↑ | **49.52** | 33.79 | 38.67 | 29.79 | 34.43 | 12.60 | 18.85 | 16.41 |
| | M-score ↓ | **2.14** | 5.83 | 91.73 | 14.13 | 8.67 | 14.82 | 15.72 | 11.17 |
| STUFF-CIR-L | FID ↓ | **15.27** | 18.37 | 19.31 | 28.59 | 18.24 | 19.75 | 16.96 | 112.33 |
| | R-prcn (%) ↑ | **36.39** | 32.65 | 26.06 | 6.86 | 35.85 | 12.39 | 23.44 | 15.71 |
| | M-score ↓ | **10.29** | 13.96 | 67.46 | 19.24 | 15.34 | 14.01 | 17.42 | 39.66 |

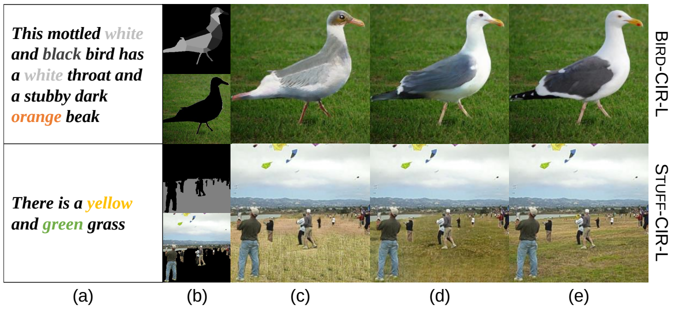

This mottled white and black bird has a white throat and a stubby dark orange beak
There is a yellow and green grass
BIRD-CIR-L
STUFF-CIR-L
(a)
(b)
(c)
(d)
(e)

Fig. 5. Comparison between different architectures. (a) The color conditions. (b) The geometry conditions and the background conditions. (c) The TP+L CIR model [54]. (d) The UF+L CIR model. (e) Original images.

## 7 EXPERIMENTS

### 7.1 Quantitative evaluation metrics

We use three evaluation metrics to evaluate performance of synthetic results: *(i)* We use the Fréchet inception distance (FID) [13] score to evaluate the synthetic image quality through measuring the distance between the distribution of synthetic results and the distribution of real images. *(ii)* We use R-precision [59] to evaluate whether the synthetic images are well conditioned on the user-given color condition, where 31 random negative samples are mixed to confuse the color-image retrieval. *(iii)* The M-score [49] is the output by a manipulation detection model [67], used to detect whether the image has been manipulated. We randomly pick 100 synthetic images to calculate the average M-score.

### 7.2 Two-phased CIR models *vs.* unified CIR models

In our preliminary research [45], we have shown that the UF+A CIR model produces a more satisfying synthetic quality than the TP+A CIR model. As shown in Fig. 5, the UF+L CIR model could more accurately estimate correspondences between image regions and the color condition (top row), and synthesize the user-specified color (bottom row). Additionally, Tab. 1 shows the UF+L CIR model outperforms the TP+L CIR model in every evaluation score. As their input conditions are different, we cannot compare TP+A with UF+L or TP+L with UF+A.

### 7.3 Comparison with related methods

Based on our demonstration in Sec. 7.2 that unified CIR models outperform two-phased CIR models (in attribute and language color conditions, respectively), we only conduct comparison experiments with unified CIR models in this subsection. Besides existing CIR models (TP+A [55], TP+L [54], UF+A [45]) in our preliminary works, we make modifications for related methods (*i.e.*, CSC [35], TdaNet [65], SGE-Net [22], and Imagic [17]) to make quantitative and qualitative comparisons with unified CIR models across all 4 CIR datasets in a more comprehensive manner.

We show synthetic images in Fig. 6 and evaluation scores in Tab. 1, which demonstrates unified CIR models achieve the best performance in synthetic quality, condition consistency, and compositing effect among related methods.

**CSC** [35] is proposed to synthesize complex scenes under the guidance of user-provided attributes and masks, which consists of a background generator and a foreground generator. Considering that the background is specified by users in the CIR task, we remove the background generator and replace its output with the user-provided background condition. However, lacking color tone adjustment, a clear boundary exists between the repainted visual content and background regions, *e.g.*, first and second rows in Fig. 6, resulting in negative impact on M-score in Tab. 1.

**TDANet** [65] is designed to restore damaged images using the user-provided language. To apply this model to the CIR task, we replace all attention modules with aggregation operators on datasets annotated by attributes (PERSON-CIR-A and LANDSCAPE-CIR-A). Following the setting in [65], we randomly generate damaged masks during training and build the variant as **TDANet-R**. Since the CIR task focus on describing concrete foreground objects (*e.g.*, person and grass), we create an additional variant (**TDANet-F**) that only samples regions of foreground objects as damaged masks. As a result, TDANet-R cannot determine the boundary of the foreground objects, as shown in third and fourth rows of Fig. 6. Although Tab. 1 shows TDANet-F could improve the visual quality, it is still difficult to synthesize fine-grained details, *e.g.*, first and third rows in Fig. 6. This is because these variants repaint visual content without the geometry condition guidance.

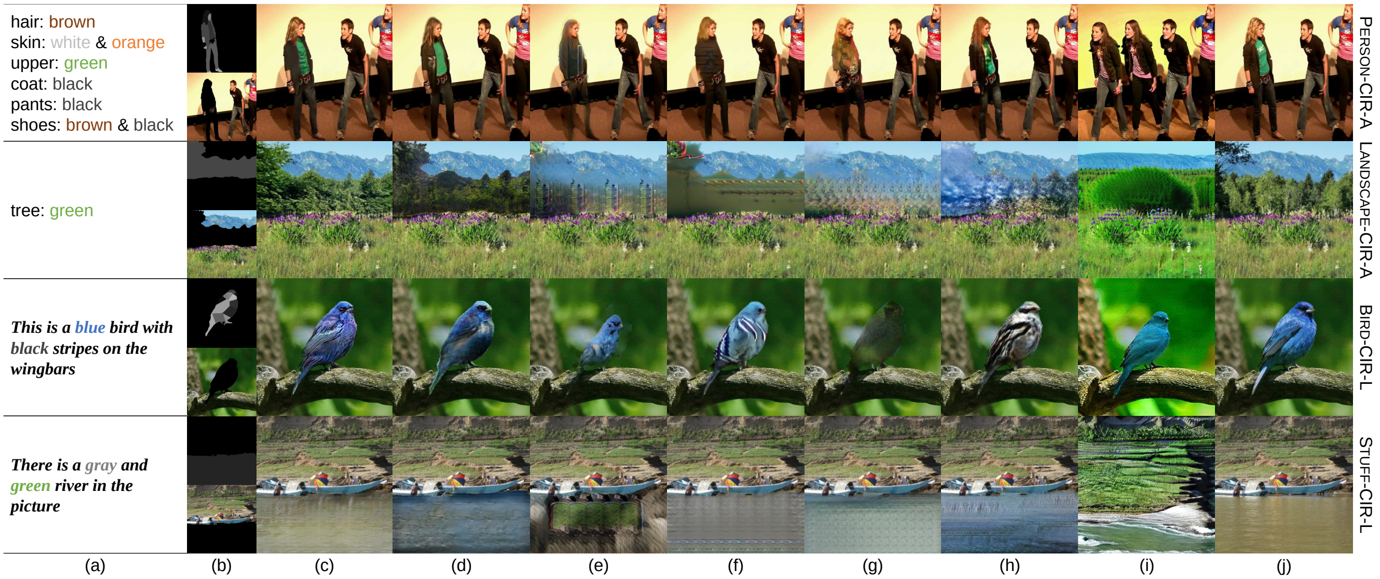


Fig. 6. Qualitative comparison results with state-of-the-art methods. (a) The color conditions. (b) The geometry conditions and the background conditions. (c) The UF+A CIR model (first and second rows) or the UF+L CIR model (third and fourth rows). (d) CSC [35]. (e) TDANet-R [65]. (f) TDANet-F [65]. (g) SGENet-R [22]. (h) SGENet-F [22]. (i) Imagic [17]. (j) Original images.

**SGENet** [22] adopts the estimated parsing mask as the guidance to restore damaged regions of images. When applying it to the CIR task, we discard all estimation modules and replace estimated results with the user-provided geometry condition. Similar to TDANet [65], we also build **SGENET-R** and **SGENET-F** by adopting different sampling strategies. In spite of the geometry condition providing the reliable structure information of the damaged regions, SGENet-R tends to make the foreground objects transparent to produce harmonious results, *e.g.*, second and third rows in Fig. 6. SGENet-F also suffers from unnatural colors since the color condition has not been taken into account, *e.g.*, second and fourth rows in Fig. 6.

**Imagic** [17] is the state-of-the-art diffusion-based image editing technique, enabling users to edit the image in accordance with the specified language description. For datasets annotated with the attribute color condition, we create language descriptions by concatenating them. We use the Stable Diffusion [42] as prerequisite pretrained text-to-image diffusion models. Although Imagic demonstrates remarkable capabilities in generating realistic repainted results, notable modifications to the background regions are observed, *e.g.*, first and fourth rows in Fig. 6. These could possibly be attributed to its text embedding and fine-tuning processes that potentially damage specific details of the original image. Consequently, it does not stand out in Tab. 1.

**User study.** In addition to qualitative and quantitative comparisons, we further conduct user study experiments for all CIR dataset to measure whether synthetic results of our unified CIR models are favored by human observers. For consistency of experimental settings, we exclude Imagic [17] since it notably modifies background regions and generates deviating results from other methods. We show all input conditions, original images and synthetic results and ask participants to choose the most visually pleasing results. We publish these experiments on Amazon Mechanical Turk (AMT). Each experiment includes 100 examples, which are completed by 25 participants.As shown in Tab. 2, our unified CIR models achieve obvious higher scores on all datasets, which confirms the subjective advantages of our methods.

TABLE 2
User study results of unified CIR models on all 4 CIR datasets.

| CSC | TDANet-R | TDANet-F | SGENet-R | SGENet-F0 | Ours/UF+A |
|---|---|---|---|---|---|
| PERSON-CIR-A | | | | | |
| 13.04 % | 3.56 % | 4.24 % | 3.76 % | 18.44 % | **56.96 %** |
| LANDSCAPE-CIR-A (256 × 256) | | | | | |
| 13.12 % | 3.84 % | 4.80 % | 6.56 % | 12.68 % | **59.00 %** |
| LANDSCAPE-CIR-A (512 × 512) | | | | | |
| 21.80 % | 3.76 % | 5.96 % | 6.72 % | 16.36 % | **45.40 %** |

| CSC | TDANet-R | TDANet-F | SGENet-R | SGENet-F | Ours/UF+L |
|---|---|---|---|---|---|
| BIRD-CIR-L | | | | | |
| 12.92 % | 4.52 % | 11.6 % | 5.08 % | 9.16 % | **56.72 %** |
| STUFF-CIR-L | | | | | |
| 12.75 % | 6.52 % | 8.76 % | 7.92 % | 13.63 % | **50.42 %** |

### 7.4 Ablation study

We study the effectiveness of LA attention and CSS by ablation experiments, and further show the evaluation scores and repainted results in Tab. 3 and Fig. 7, respectively.

**W/o EB.** We measure the necessity of the adaptive correspondence estimation by disabling the estimation branch in LA attention. As such, repainted visual contents lose some texture and color detail, *e.g.*, the second row in Fig. 7.

**W/o RB.** This ablation performs cross-modality attention to estimating correspondences as GPT-based models, resulting in the repainted visual content being clearly mismatched with the color condition, *e.g.*, the first row in Fig. 7.

**SEBE.** We replace the LA attention module with the semantic bridge attention proposed in TP+L [54]. As a result, estimated correspondences become unstable, and the visual quality suffers greatly, *e.g.*, the second row in Fig. 7.

**W/o CSS.** This ablation discards all CSS modules and breaks the similarity of distribution of color features across each spatially-specific color condition, resulting in incorrect boundaries of the part regions in the repainted visual content, *e.g.*,the first row in Fig. 7.

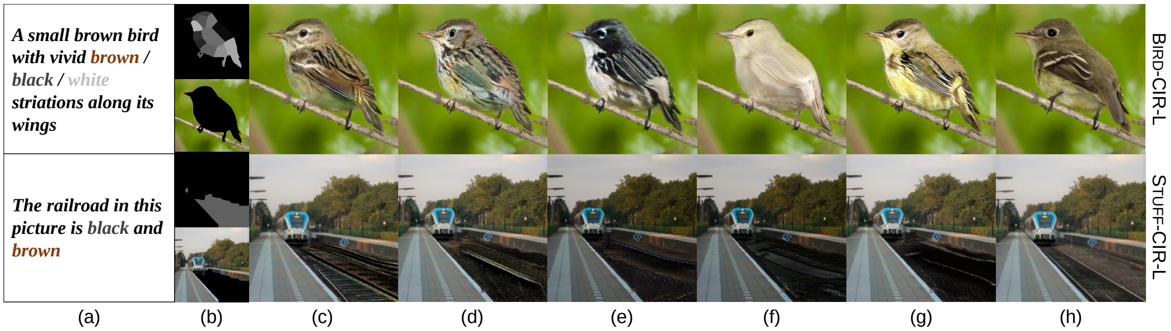

A small brown bird with vivid brown / black / white striations along its wings
The railroad in this picture is black and brown
BIRD-CIR-L
STUFF-CIR-L
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)

Fig. 7. The UF+L CIR model ablation study on CIR datasets with the language color condition. (a) The color conditions. (b) The geometry conditions and the background conditions. (c) The UF+L CIR model. (d) W/o EB. (e) W/o RB. (f) SEBE. (g) W/o CSS. (h) Original images.

TABLE 3
Quantitative ablation of the UF+L CIR model on datasets with the language color condition.

| Dataset | Score | Ours | Ablation | | | |
|---|---|---|---|---|---|---|
| | | UF+L | W/o EB | W/o RB | SEBE | W/o CSS |
| BIRD-CIR-L | FID ↓ | **10.54** | 11.40 | 13.63 | 13.47 | 11.81 |
| | R-prcn (%) ↑ | **49.52** | 38.60 | 28.50 | 29.58 | 33.47 |
| | M-score ↓ | **2.14** | 3.56 | 2.73 | 4.13 | 3.37 |
| Stuff-CIR-L | FID ↓ | **15.27** | 15.95 | 16.28 | 16.12 | 16.07 |
| | R-prcn (%) ↑ | **36.39** | 33.20 | 30.12 | 31.55 | 31.01 |
| | M-score ↓ | **10.29** | 11.49 | 11.62 | 13.25 | 11.78 |

## 8 Application

In this section, we introduce the various application scenarios of the CIR task and show results of our unified CIR models in Fig. 8. We first show the color interpolation and iteratively editing by modified input conditions to demonstrate the robustness of our model. Next, we apply the CIR model to meet the practical needs of users by utilizing the knowledge gained from different datasets, *i.e.*, object insertion [9] (trained on BIRD-CIR-L dataset) and fashion editing [8] (trained on PERSON-CIR-A dataset). After that, we further explore the feasibility of the CIR model to synthesize virtual video: using a video clip as the background condition and inserting a virtual person who performs a meaningful action. To maintain the appearance consistency of the inserted person, we fix the color condition and texture condition, and use the optical flow to smooth the repainted results. Finally, we use two examples to demonstrate the potential for CIR models to be extended to other related tasks in future research as:

- The damaged image restoration task [34]. We randomly generate damaged masks instead of sampling masks that represent regions of foreground objects and overlay them on the original images in order to simulate damaged images during training. To adopt CIR models, we use damaged image as the background condition, and provide the texture condition, color condition, and the geometry condition to guide the restoration. We additionally use the L1 loss to push the restored image regions in the direction of the ground truth.
- The image colorization task [66]. We modify the model to output $ab$ channels as the predicted chromatic features, which are then concatenated with the grayscale image to synthesize the colorized result. In addition, We concrete the texture condition with the grayscale image rather than the default Gaussian noise to colorize images under the guidance of the grayscale image. We use the color condition and the geometry condition to describe the expectations of colorization results. The background condition is set to all zeros in this task. Since the grayscale image has spatial dimension, we use convolution layers to encode the texture condition and extract its parameters. Furthermore, a L1 loss is applied to make predicted colors similar to the ground truth.

We further provide detailed illustrations of the application in the supplementary material.

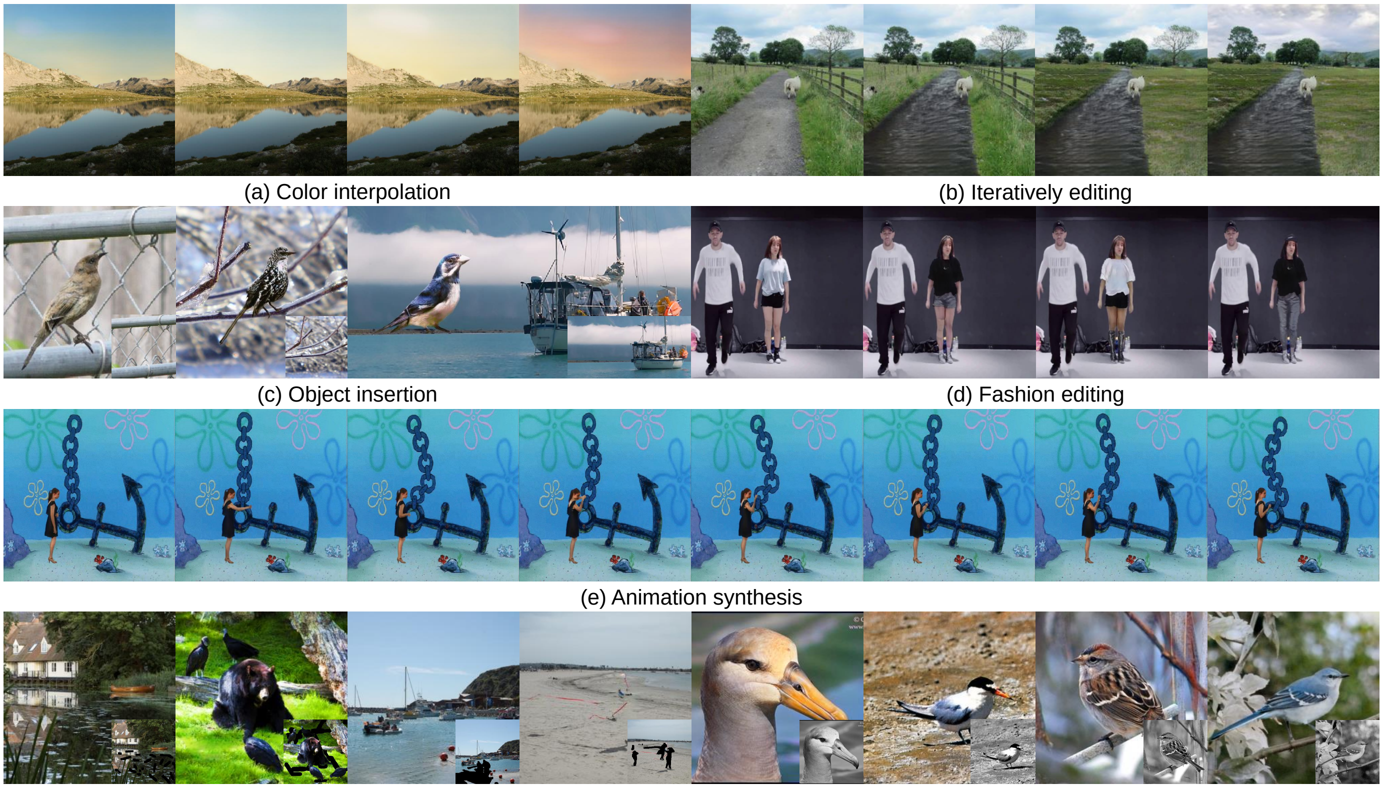
## 9 Conclusion

In this paper, we propose the conditional image repainting models to repaint some visual content with several user-provided cross-modality conditions and seamlessly composite it with the original image. We first define condition inputs and introduce two-phased CIR models, and then present unified CIR models that break the two-phased dependency limitation and achieve improved visual quality. Based on four our collected and processed datasets, it has been demonstrated that CIR models are effective in supporting attributes or language used to describe color. We further introduce various application scenarios of the CIR task to present its potential to meet user's practical requirements.

**Limitation.** While experiments across four datasets have showcased the potential of the CIR model in synthesizing general repainted results, the training of a large-scale CIR model for broader applications remains challenging due to the scarcity of well-annotated training data. Insufficient training data also result in our repainted images being less photo-realistic compared to those produced by larger generative models. An alternative approach is to distill the high-level semantic knowledge from recently emerging large-scale pretrained models, *e.g.*, Stable Diffusion [42], SAM [18], and GPT-4 [31]. Additionally, considering to integrate advanced generative frameworks and scaling up our model to improve feature representation for superior repainting results are left as our future work.

## Acknowledgments

This work was supported by the National Key R&D Program of China (2021ZD0109800), the National Natural Science Foundation of China under Grant No. 62136001, 62088102.



(a) Color interpolation

(b) Iteratively editing

(c) Object insertion

(d) Fashion editing

(e) Animation synthesis

(f) Damaged image restoration

(g) Image colorization

Fig. 8. The various application scenarios of the CIR model. (a) As the attribute color condition interpolates linearly from blue to orange, the color of the sky smoothly changes. (b) By modifying the geometry condition and color condition, our model could iteratively repaint the image and eventually synthesize completely different results. (c) We are able to insert objects with user-provided geometry and color condition into any image (shown in the right bottom) using our model. (d) Our CIR model allows us to edit the style (*i.e.*, color, texture, and length) of clothes that people wear. (e) By modifying the geometry condition and the background condition, while keeping the color condition and texture condition unchanged, we could synthesize the dramatic video (a girl in a dance costume fixes chains in the underwater world). (f) We could use the CIR model to perform damaged image restoration task by setting the damaged image (shown in the right bottom) as the background condition. (g) The CIR model could be used to achieve image colorization by adopting the grayscale image (shown in the right bottom) as the texture condition.

## References

[1] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and stuff classes in context. In *CVPR*, 2018.

[2] Bor-Chun Chen and Andrew Kae. Toward realistic image compositing with adversarial learning. In *CVPR*, 2019.

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *TPAMI*, 2017.

[4] Jun Cheng, Fuxiang Wu, Yanling Tian, Lei Wang, and Dapeng Tao. RiFeGAN: Rich feature generation for text-to-image synthesis from prior knowledge. In *CVPR*, 2020.

[5] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, 2018.

[6] Daniel Cohen-Or, Olga Sorkine, Ran Gal, Tommer Leyvand, and Ying-Qing Xu. Color harmonization. In *SIGGRAPH*, 2006.

[7] Xiaodong Cun and Chi-Man Pun. Improving the harmony of the composite image by spatial-separated attention module. *TIP*, 2020.

[8] Haoye Dong, Xiaodan Liang, Yixuan Zhang, Xujie Zhang, Xiaohui Shen, Zhenyu Xie, Bowen Wu, and Jian Yin. Fashion editing with adversarial parsing learning. In *CVPR*, 2020.

[9] Oran Gafni and Lior Wolf. Wish you were here: Context-aware human generation. In *CVPR*, 2020.

[10] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016.

[11] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[12] Zonghui Guo, Haiyong Zheng, Yufeng Jiang, Zhaorui Gu, and Bing Zheng. Intrinsic image harmonization. In *CVPR*, 2021.

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a nash equilibrium. In *NIPS*, 2017.

[14] Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017.

[15] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[16] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.

[17] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *CVPR*, 2023.

[18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

[19] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip HS Torr. ManiGAN: Text-guided image manipulation. In *CVPR*, 2020.

[20] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-driven text-to-image synthesis via adversarial training. In *CVPR*, 2019.

[21] Yuheng Li, Krishna Kumar Singh, Utkarsh Ojha, and Yong Jae Lee. MixNMatch: Multifactor disentanglement and encoding for conditional image generation. In *CVPR*, 2020.

[22] Liang Liao, Jing Xiao, Zheng Wang, Chia-Wen Lin, and Shin'ichi Satoh. Guidance and evaluation: Semantic-aware image inpainting for mixed scenes. In *ECCV*, 2020.

[23] Tianwei Lin, Zhuoqi Ma, Fu Li, Dongliang He, Xin Li, Errui Ding, Nannan Wang, Jie Li, and Xinbo Gao. Drafting and revision: Laplacian pyramid network for fast high-quality artistic style transfer. In *CVPR*, 2021.

[24] Jun Ling, Han Xue, Li Song, Rong Xie, and Xiao Gu. Region-aware adaptive instance normalization for image harmonization.

In *CVPR*, 2021.

[25] Songhua Liu, Tianwei Lin, Dongliang He, Fu Li, Meiling Wang, Xin Li, Zhengxing Sun, Qian Li, and Errui Ding. Revisit attention mechanism in arbitrary neural style transfer. In *ICCV*, 2021.

[26] Varun Manjunatha, Mohit Iyyer, Jordan L Boyd-Graber, and Larry S Davis. Learning to color from language. In *NAACL*, 2018.

[27] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *ACL: System Demonstrations*, 2014.

[28] Yifang Men, Yiming Mao, Yuning Jiang, Wei-Ying Ma, and Zhouhui Lian. Controllable person image synthesis with attribute-decomposed gan. In *CVPR*, 2020.

[29] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[30] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z. Qureshi, and Mehran Ebrahimi. EdgeConnect: Generative image inpainting with adversarial edge learning. In *ICCVW*, 2019.

[31] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[32] Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *ECCV*, 2020.

[33] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019.

[34] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[35] Dario Pavllo, Aurelien Lucchi, and Thomas Hofmann. Controlling style and semantics in weakly-supervised image generation. In *ECCV*, 2020.

[36] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[37] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

[38] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *SIGGRAPH*, 2003.

[39] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription. In *CVPR*, 2019.

[40] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[41] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *CVPR*, 2016.

[42] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[43] Shulan Ruan, Yong Zhang, Kun Zhang, Yanbo Fan, Fan Tang, Qi Liu, and Enhong Chen. DAE-GAN: Dynamic aspect-aware gan for text-to-image synthesis. In *ICCV*, 2021.

[44] Jheng-Wei Su, Hung-Kuo Chu, and Jia-Bin Huang. Instance-aware image colorization. In *CVPR*, 2020.

[45] Jimeng Sun, Shuchen Weng, Zheng Chang, Si Li, and Boxin Shi. UniCoRN: A unified conditional image repainting network. In *CVPR*, 2022.

[46] Kalyan Sunkavalli, Micah K Johnson, Wojciech Matusik, and Hanspeter Pfister. Multi-scale image harmonization. *TOG*, 2010.

[47] Hongchen Tan, Xiuping Liu, Xin Li, Yi Zhang, and Baocai Yin. Semantics-enhanced adversarial nets for text-to-image synthesis. In *ICCV*, 2019.

[48] Shashank Tripathi, Siddhartha Chandra, Amit Agrawal, Ambrish Tyagi, James M. Rehg, and Visesh Chari. Learning to generate synthetic data via compositing. In *CVPR*, 2019.

[49] Shashank Tripathi, Siddhartha Chandra, Amit Agrawal, Ambrish Tyagi, James M. Rehg, and Visesh Chari. Learning to generate synthetic data via compositing. In *CVPR*, 2019.

[50] Yi-Hsuan Tsai, Xiaohui Shen, Zhe Lin, Kalyan Sunkavalli, and Ming-Hsuan Yang. Sky is not the limit: semantic-aware sky replacement. *TOG*, 2016.

[51] Yi-Hsuan Tsai, Xiaohui Shen, Zhe Lin, Kalyan Sunkavalli, Xin Lu, and Ming-Hsuan Yang. Deep image harmonization. In *CVPR*, 2017.

[52] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[53] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018.

[54] Shuchen Weng, Wenbo Li, Dawei Li, Hongxia Jin, and Boxin Shi. Conditional image repainting via semantic bridge and piecewise value function. In *ECCV*, 2020.

[55] Shuchen Weng, Wenbo Li, Dawei Li, Hongxia Jin, and Boxin Shi. MISC: Multi-condition injection and spatially-adaptive compositing for conditional person image synthesis. In *CVPR*, 2020.

[56] Shuchen Weng, Jimeng Sun, Yu Li, Si Li, and Boxin Shi. $\text{CT}^2$: Colorization transformer via color tokens. In *ECCV*, 2022.

[57] Shuchen Weng, Yi Wei, Ming-Ching Chang, and Boxin Shi. Instance contour adjustment via structure-driven CNN. In *ECCV*, 2022.

[58] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. GP-GAN: towards realistic high-resolution image blending. In *ACM MM*, 2019.

[59] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018.

[60] Guojun Yin, Bin Liu, Lu Sheng, Nenghai Yu, Xiaogang Wang, and Jing Shao. Semantics disentangling for text-to-image generation. In *CVPR*, 2019.

[61] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Generative image inpainting with contextual attention. In *CVPR*, 2018.

[62] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Free-form image inpainting with gated convolution. In *ICCV*, 2019.

[63] Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. Cross-modal contrastive learning for text-to-image generation. In *CVPR*, 2021.

[64] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.

[65] Lisai Zhang, Qingcai Chen, Baotian Hu, and Shuoran Jiang. Text-guided neural image inpainting. In *ACM MM*, 2020.

[66] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

[67] Peng Zhou, Xintong Han, Vlad I. Morariu, and Larry S. Davis. Learning rich features for image manipulation detection. In *CVPR*, 2018.

[68] Qixian Zhou, Xiaodan Liang, Ke Gong, and Liang Lin. Adaptive temporal encoding network for video instance-level human parsing. In *ACM MM*, 2018.

[69] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. DM-GAN: dynamic memory generative adversarial networks for text-to-image synthesis. In *CVPR*, 2019.

[70] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. SEAN: Image synthesis with semantic region-adaptive normalization. In *CVPR*, 2020.

**Shuchen Weng** is a PhD student in School of Computer Science, Peking University. His research interests include cross-modality (mainly language-based) content creation and manipulation.

**Boxin Shi** received the BE degree from the Beijing University of Posts and Telecommunications, the ME degree from Peking University, and the PhD degree from the University of Tokyo, in 2007, 2010, and 2013. He is currently a Boya Young Fellow Assistant Professor and Research Professor at Peking University, where he leads the Camera Intelligence Lab. Before joining PKU, he did research with MIT Media Lab, Singapore University of Technology and Design, Nanyang Technological University, National Institute of Advanced Industrial Science and Technology, from 2013 to 2017. His papers were awarded as Best Paper Runner-Up at International Conference on Computational Photography 2015 and selected as Best Papers from ICCV 2015 for IJCV Special Issue. He is an associate editor of TPAMI/IJCV and an area chair of CVPR/ICCV. He is a senior member of IEEE.

# Supplementary Material: Conditional Image Repainting

Shuchen Weng and Boxin Shi*, *Senior Member, IEEE*

## 10 Reduced stochastic gradient variance

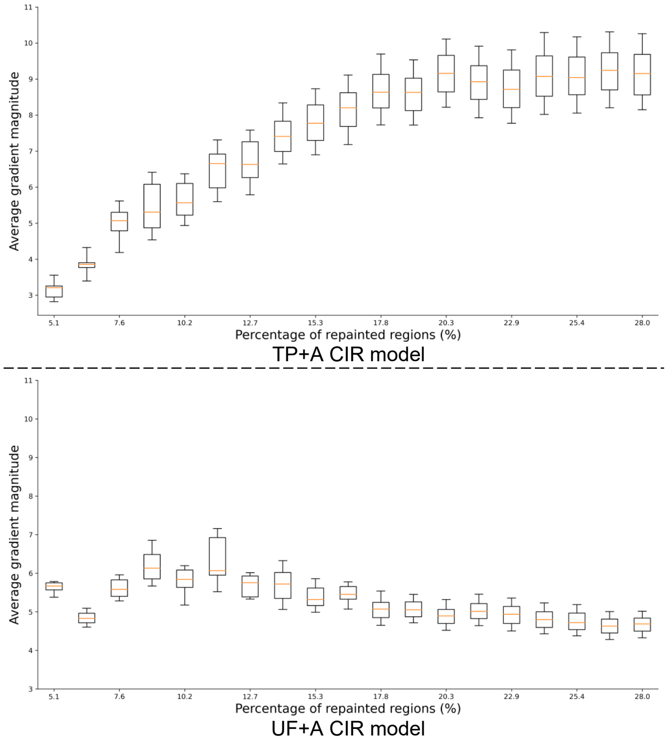
We show the box plots of gradient magnitude in Fig. 9. The lateral axis denotes the percentage of the repainted area to the whole image area. The vertical axis denotes the gradient magnitude distribution of the last convolutional layer that generates visual content. The two-phased CIR model (TP+A [6]) discards the generated meaningless background, which masks the gradient of background regions, making the gradient magnitude distribution related to the area of repainted regions, as shown in Fig. 9 (top row). This enlarges the variance of the stochastic gradient among samples and causes low convergence. The gradient magnitude distribution of the unified CIR model (UF+A [4]) is shown in Fig. 9 (bottom row), demonstrating that the gradient of unified CIR models is irrelevant to the area of repainted regions. As such, the variance of the stochastic gradient among samples is smaller, which is one of the reasons why unified CIR models perform better than two-phased CIR models.

## 11 Modified TP+L CIR model

To make this paper well-organized, we build the TP+L CIR model as a variant of the original model [5]. Specifically, we replace the original piecewise value function with the compositing loss proposed in the TP+A CIR model [6], as shown in Eq. (14) of the main paper. In this way, two-phased CIR models have the same architecture and training losses, which makes this paper easy to follow. This modification has a minimal impact on performance, *e.g.*, the FID score only increases from 12.08 to 12.16 in the Bird-CIR dataset.

## 12 Additional qualitative results

In Fig. 10, we show more qualitative comparison results with related methods across all 4 CIR datasets, including CSC [3], TDANet (TDANet-R and TDANet-F) [7], SGENet (SGENet-R and SGENet-F) [2], and Imagic [1]. We further show more qualitative comparison between different architectures and more ablation study of the UF+L CIR model on the effect of LA attention and CSS in Fig. 11. We provide detailed descriptions in Sec. 7.2, Sec. 7.3, and Sec. 7.4 of the main paper.



TP+A CIR model

UF+A CIR model

Fig. 9. The box plots of gradient magnitude with the area of repainted regions in the two-phased CIR model (TP+A [6], top row) and in the unified CIR model (UF+A [4], bottom row). This verifies that the variance of the stochastic gradient among samples is reduced in the proposed unified architecture.

## 13 Application

In Fig. 12, we show more results of color interpolation and the process of iteratively editing using unified CIR models. In Fig. 13, we in detail demonstrate the practical applications of CIR models in object insertion and fashion editing. In Fig. 14, We present an example of synthesizing virtual video using unified CIR models. In Fig. 15, we modify unified CIR models to restore damaged images and colorize grayish images, which demonstrates that CIR models have significant potential to be extended to other related tasks in future research. To demonstrate the robustness and controllability of unified CIR models, we create the bird matrix by controlling input conditions in Fig. 16. Furthermore, we repaint diverse results with different texture conditions in Fig. 17.

- * *Corresponding author (shiboxin@pku.edu.cn).*
- *S. Weng and B. Shi are with the National Key Laboratory for Multimedia Information Processing and National Engineering Research Center of Visual Technology, School of Computer Science, Peking University, Beijing 100871, China.*

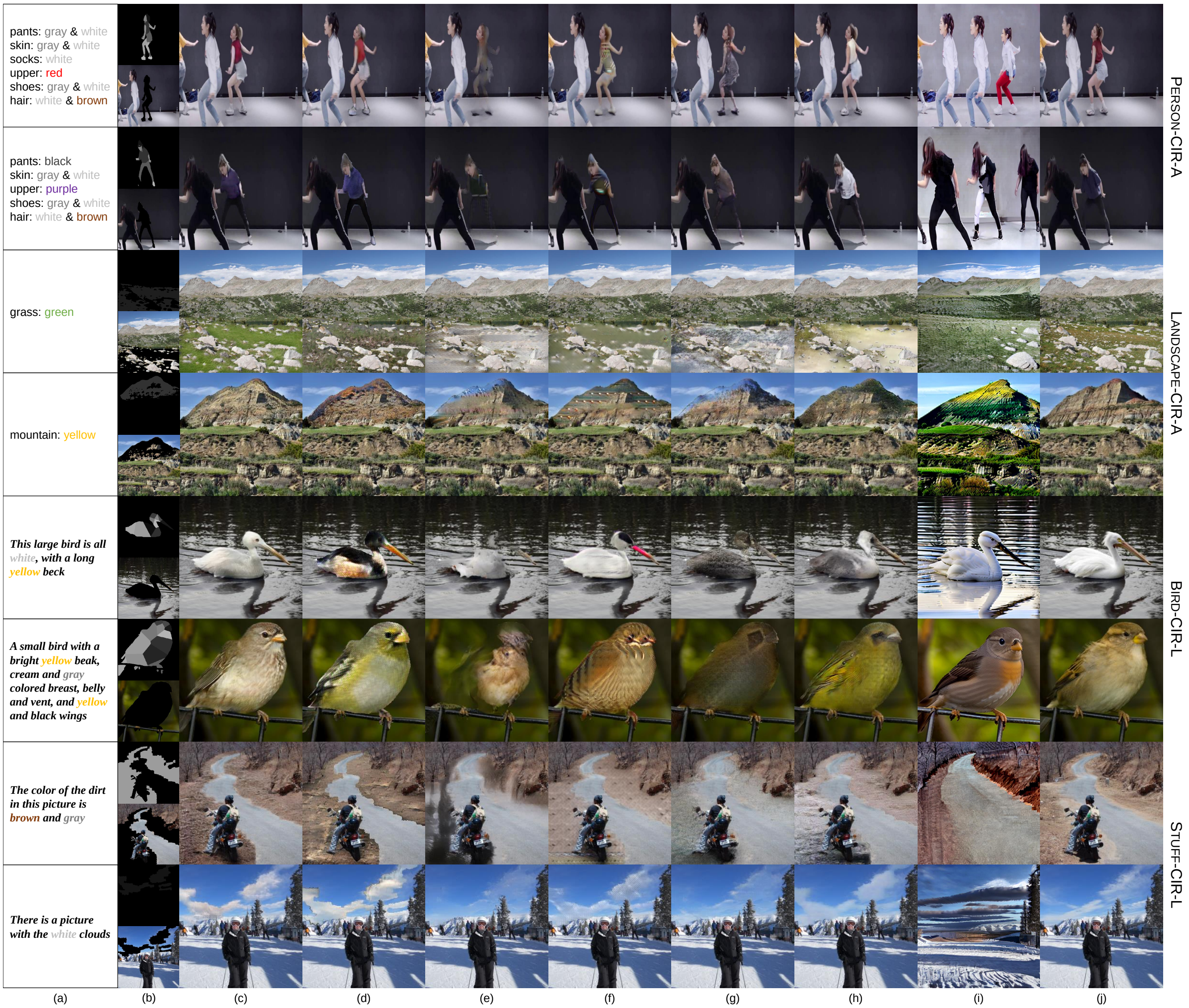


Fig. 10. Qualitative comparison results with related methods on 4 CIR datasets. (a) The color conditions. (b) The geometry conditions and the background conditions. (c) The UF+A CIR model (top four rows) or the UF+L CIR model (bottom four rows). (d) CSC [3]. (e) TDANet-R [7]. (f) TDANet-F [7]. (g) SGENet-R [2]. (h) SGENet-F [2]. (i) Imagic [1]. (j) Original images.

## 14 Acquisition of geometry condition

There are three strategies to acquire our geometry condition while preparing CIR datasets, including: *(i)* For manual creation, we rely on the handcrafted annotation of pixel-wise parsing masks, *e.g.*, PERSON-CIR-A dataset [6] and STUFF-CIR-L dataset. *(ii)* For automatic creation, we utilize state-of-the-art segmentation models for scene parsing, *e.g.*, LANDSCAPE-CIR-A dataset [4]. *(iii)* For semi-automatic creation, we provide semantic points and estimate the parsing mask, *e.g.*, BIRD-CIR-L dataset [5].

## References

[1] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *CVPR*, 2023.

[2] Liang Liao, Jing Xiao, Zheng Wang, Chia-Wen Lin, and Shin'ichi Satoh. Guidance and evaluation: Semantic-aware image inpainting for mixed scenes. In *ECCV*, 2020.

[3] Dario Pavllo, Aurelien Lucchi, and Thomas Hofmann. Controlling style and semantics in weakly-supervised image generation. In *ECCV*, 2020.

[4] Jimeng Sun, Shuchen Weng, Zheng Chang, Si Li, and Boxin Shi. UniCoRN: A unified conditional image repainting network. In *CVPR*, 2022.

[5] Shuchen Weng, Wenbo Li, Dawei Li, Hongxia Jin, and Boxin Shi. Conditional image repainting via semantic bridge and piecewise value function. In *ECCV*, 2020.

[6] Shuchen Weng, Wenbo Li, Dawei Li, Hongxia Jin, and Boxin Shi. MISC: Multi-condition injection and spatially-adaptive compositing for conditional person image synthesis. In *CVPR*, 2020.

[7] Lisai Zhang, Qingcai Chen, Baotian Hu, and Shuoran Jiang. Text-guided neural image inpainting. In *ACM MM*, 2020.

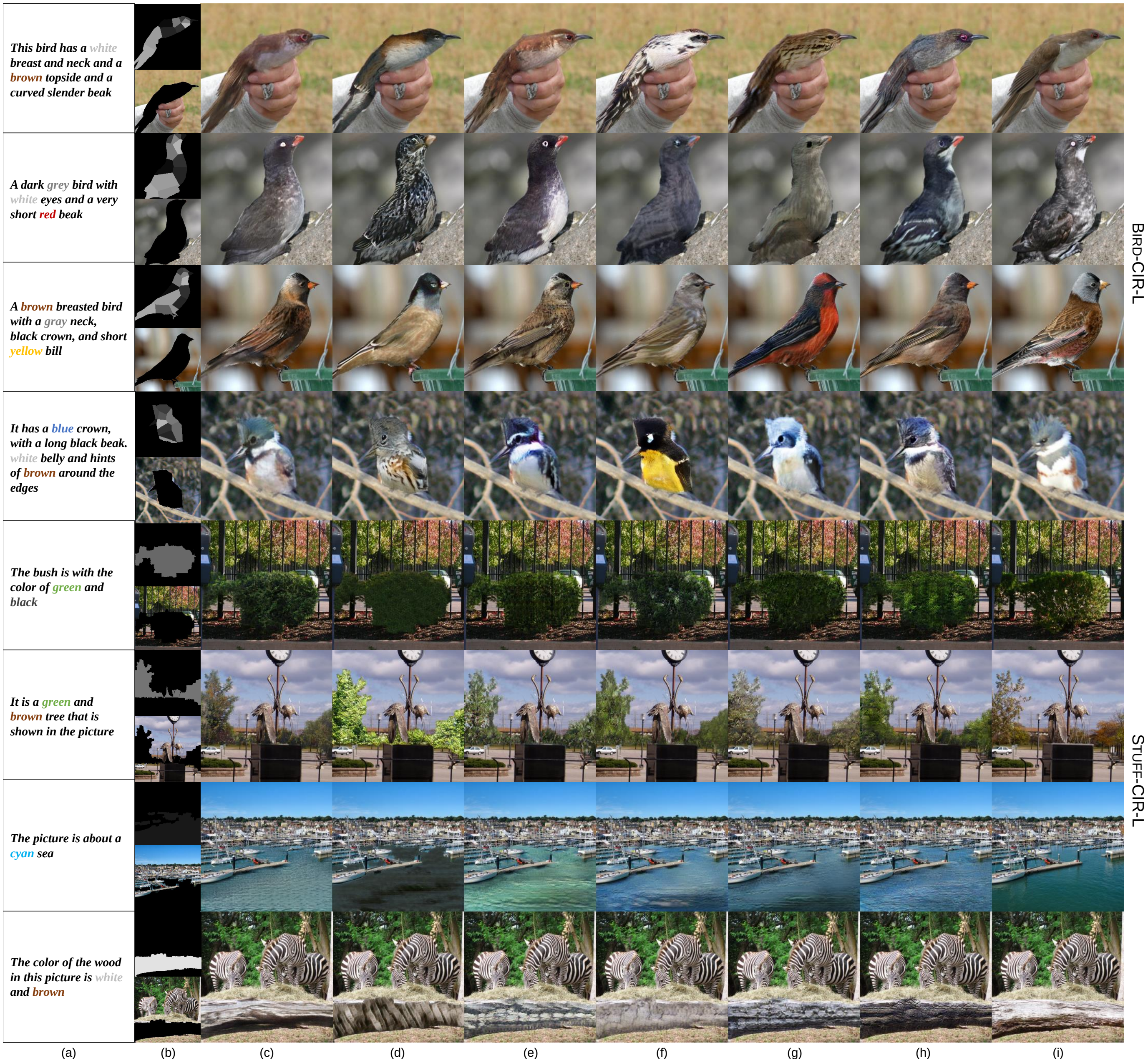
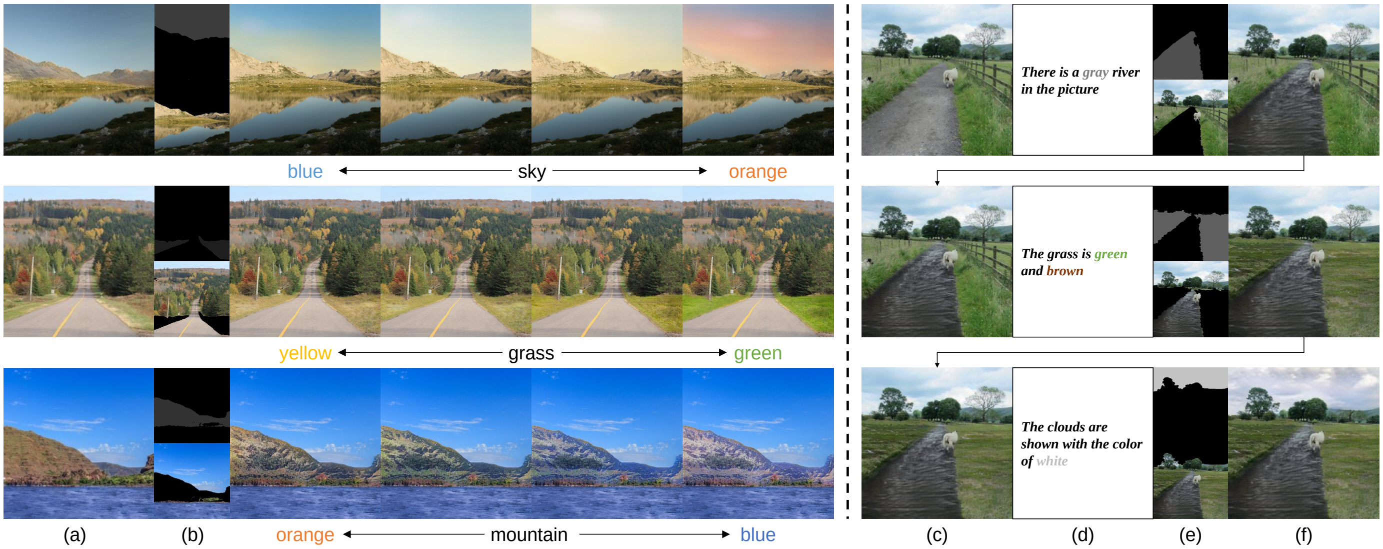

This bird has a white breast and neck and a brown topside and a curved slender beak
A dark grey bird with white eyes and a very short red beak
A brown breasted bird with a gray neck, black crown, and short yellow bill
It has a blue crown, with a long black beak. white belly and hints of brown around the edges
The bush is with the color of green and black
It is a green and brown tree that is shown in the picture
The picture is about a cyan sea
The color of the wood in this picture is white and brown
BIRD-CIR-L
STUFF-CIR-L
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)

Fig. 11. Comparison between different architectures and the UF+L CIR model ablation study on CIR datasets with the language color condition. (a) The color conditions. (b) The geometry conditions and the background conditions. (c) The UF+L CIR model. (d) The TP+L CIR model [5]. (e) W/o EB. (f) W/o RB. (g) SEBE. (h) W/o CSS. (i) Original images.

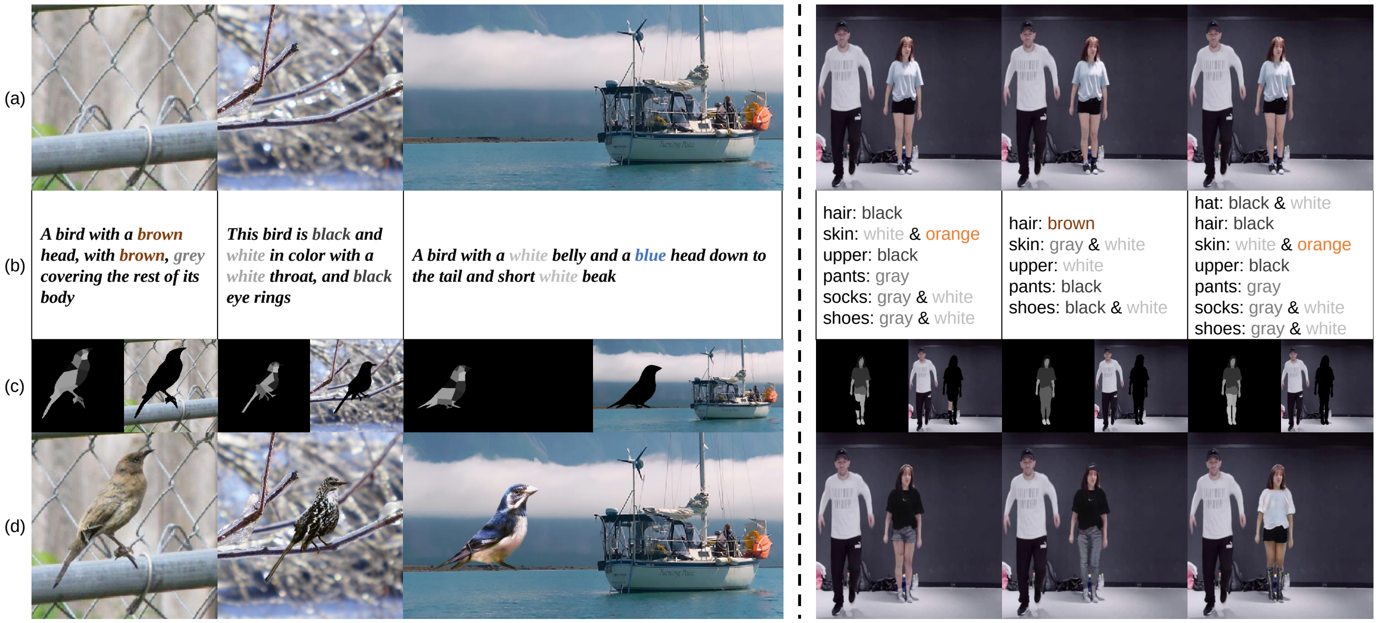


*There is a gray river in the picture*

*The grass is green and brown*

*The clouds are shown with the color of white*

Fig. 12. **Left**: More samples of color interpolation. (a) Original images. (b) The geometry conditions and the background conditions. **Right**: Illustration of Iterative editing. (c) Images to be repainted. (d) The color conditions. (e) The geometry conditions and the background conditions. (f) Repainted images.



***A bird with a brown head, with brown, grey covering the rest of its body***

***This bird is black and white in color with a white throat, and black eye rings***

***A bird with a white belly and a blue head down to the tail and short white beak***

hair: black
skin: white & orange
upper: black
pants: gray
socks: gray & white
shoes: gray & white

hair: brown
skin: gray & white
upper: white
pants: black
shoes: black & white

hat: black & white
hair: black
skin: white & orange
upper: black
pants: gray
socks: gray & white
shoes: gray & white

Fig. 13. **Left**: Illustration of object insertion. **Right**: Illustration of fashion editing. (a) Original images. (b) The color conditions. (c) The geometry conditions and the background conditions. (d) Repainted images.

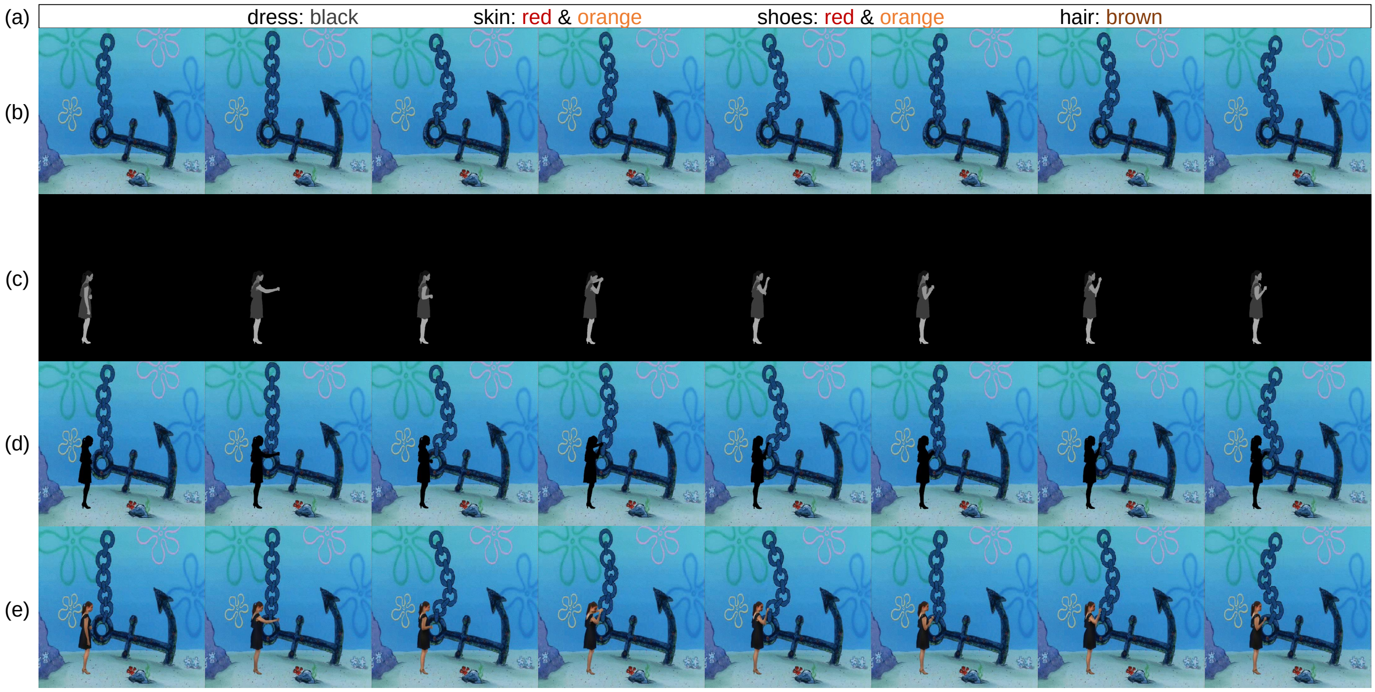


Fig. 14. Illustration of animation synthesis. (a) The fixed color condition. (b) Original images. (c) The geometry conditions. (d) The background conditions. (e) Repainted images. After all images are repainted, we use the optical flow to enhance the appearance consistency. Please check the video in the supplementary material..

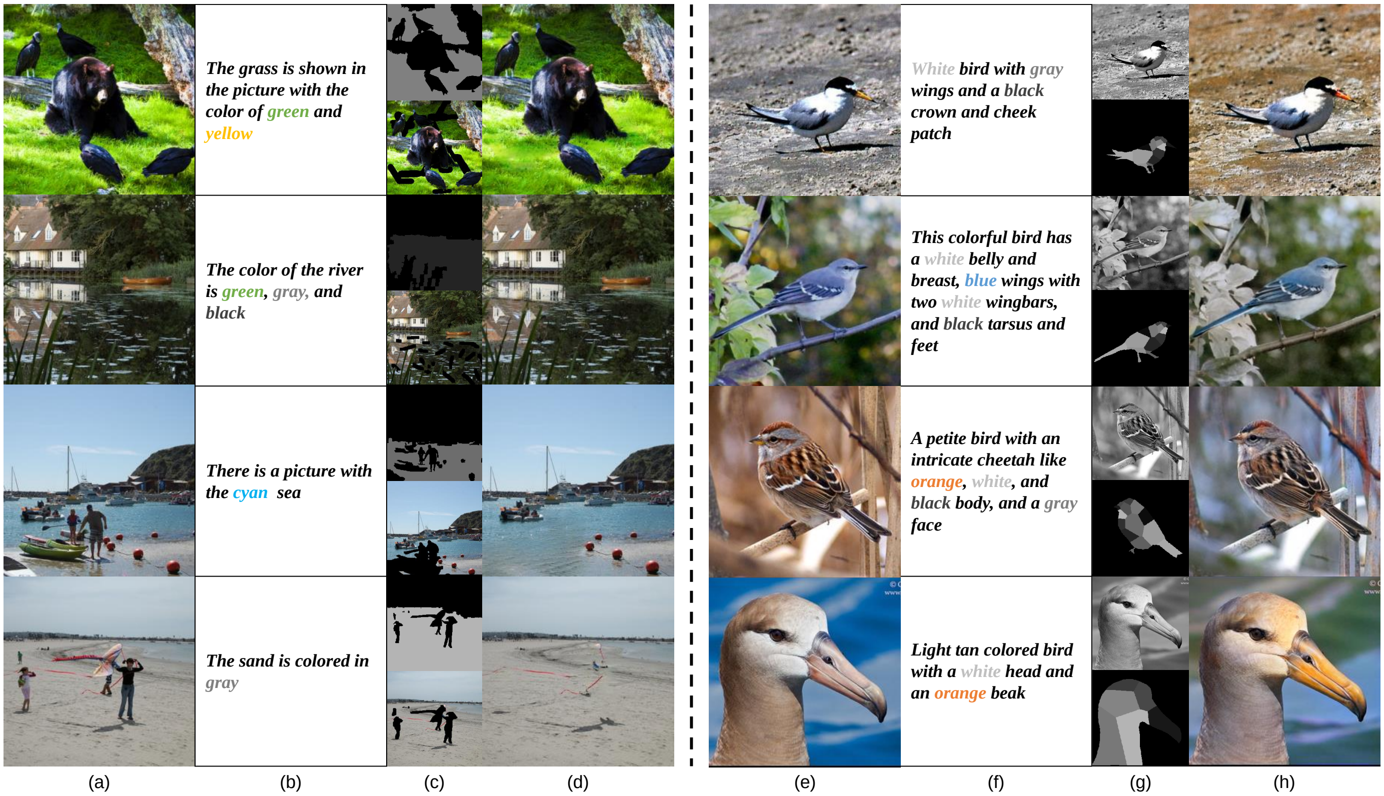

The grass is shown in the picture with the color of green and yellow
The color of the river is green, gray, and black
There is a picture with the cyan sea
The sand is colored in gray
White bird with gray wings and a black crown and cheek patch
This colorful bird has a white belly and breast, blue wings with two white wingbars, and black tarsus and feet
A petite bird with an intricate cheetah like orange, white, and black body, and a gray face
Light tan colored bird with a white head and an orange beak
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)

Fig. 15. **Left**: Illustration of damaged image restoration. (a) Ground truth. (b) The color conditions. (c) The geometry conditions and the background conditions. (d) Restoration results. **Right**: Illustration of image colorization. (e) Ground truth. (f) The color conditions. (g) The geometry conditions and the background conditions. (h) Colorization results.

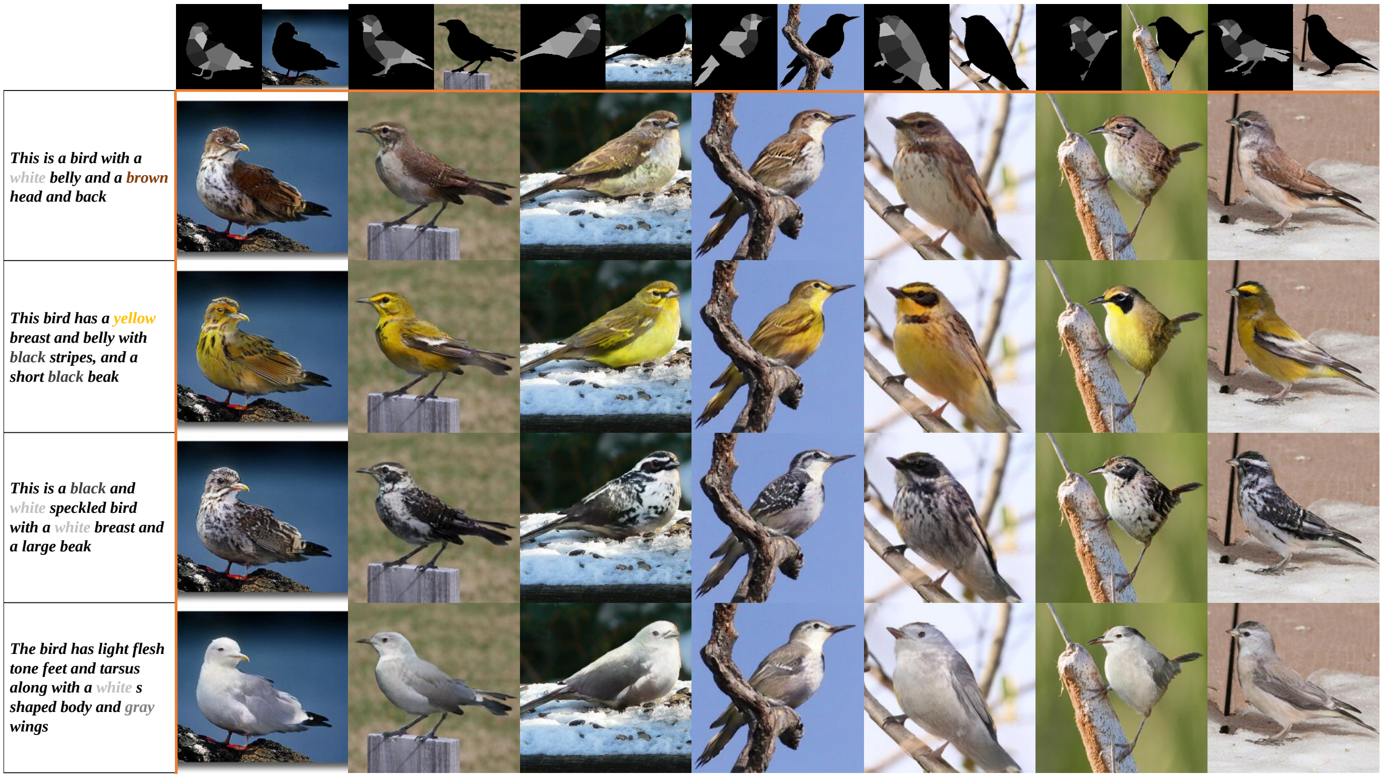

This is a bird with a white belly and a brown head and back
This bird has a yellow breast and belly with black stripes, and a short black beak
This is a black and white speckled bird with a white breast and a large beak
The bird has light flesh tone feet and tarsus along with a white s shaped body and gray wings

Fig. 16. Creating the bird matrix by controlling input conditions. Birds in each row share the same texture and color condition, which results in a similar appearance. Birds in each column share the same geometry condition and background condition, so that they have the similar posture.

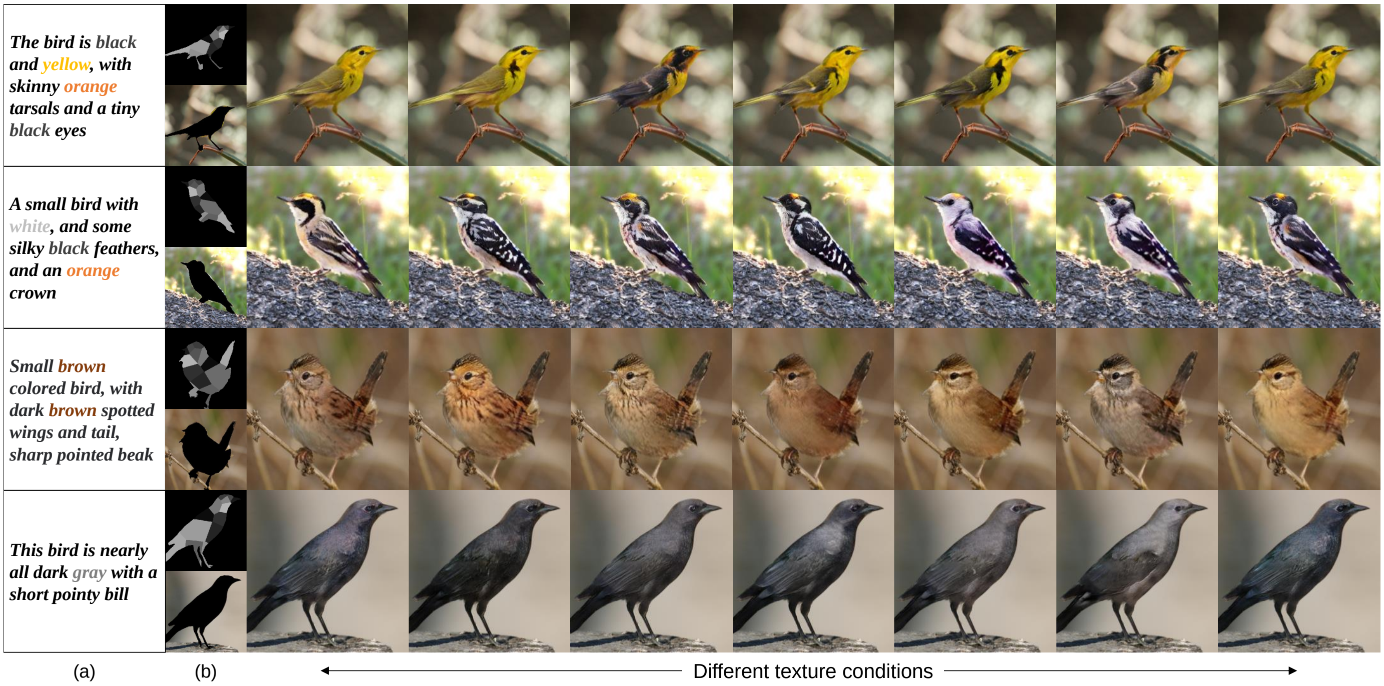

The bird is black and yellow, with skinny orange tarsals and a tiny black eyes
A small bird with white, and some silky black feathers, and an orange crown
Small brown colored bird, with dark brown spotted wings and tail, sharp pointed beak
This bird is nearly all dark gray with a short pointy bill
(a)
(b)
Different texture conditions

Fig. 17. Repainting diverse results with different texture conditions. (a) The color conditions. (b) The geometry conditions and the background conditions.