LANGUAGE-DRIVEN IMAGE STYLE TRANSFER

Anonymous authors
Paper under double-blind review

ABSTRACT

Despite having promising results, style transfer, which requires preparing style images in advance, may result in lack of creativity and accessibility. Following human instruction, on the other hand, is the most natural way to perform artistic style transfer that can significantly improve controllability for visual effect applications. We introduce a new task—language-driven image style transfer (`LDIST`)—to manipulate the style of a content image, guided by a text. We propose contrastive language visual artist (CLVA) that learns to extract visual semantics from style instructions and accomplish `LDIST` by the patch-wise style discriminator. The discriminator considers the correlation between language and patches of style images or transferred results to jointly embed style instructions. CLVA further compares contrastive pairs of content image and style instruction to improve the mutual relativeness between transfer results. The transferred results from the same content image can preserve consistent content structures. Besides, they should present analogous style patterns from style instructions that contain similar visual semantics. The experiments show that our CLVA is effective and achieves superb transferred results on `LDIST`.

1 INTRODUCTION

Style transfer (Gatys et al., 2015b; Li et al., 2017c; Huang & Belongie, 2017; Jing et al., 2017) adopts appearances and visual patterns from another reference style images to manipulate a content image. Artistic style transfer has a considerable application value for creative visual design, such as image stylization and video effect (Somavarapu et al., 2020; Zhang et al., 2019b; Wang et al., 2019a; Gao et al., 2018; Huang et al., 2017b). However, it requires preparing collections of style image in advance. It even needs to redraw new references first if there is no expected style images, which is impractical due to an additional overhead. Language is the most natural way for humans to communicate. If a system can follow textual descriptions and automatically perform style transfer, we can significantly improve accessibility and controllability.

In this paper, we introduce language-driven image style transfer (`LDIST`). As shown in Fig. 1, `LDIST` treats a content image and an instruction as the input, and the style transferred result is manipulated from a style description. It should preserve the structure of *car scene* from the content image and simultaneously modifies the style pattern that corresponds to *"horizontally lined texture, blues."* Our `LDIST` task is different from the general language-based image-editing (LBIE) (Nam et al., 2018; Li et al., 2020; Liu et al., 2020; El-Nouby et al., 2019), which usually alters objects or properties of object in an image. The main challenge of `LDIST` is to express visual semantics from language as style patterns. For example, not only color distributions (*"blue"* or *"green"*) but also texture patterns (*"horizontally lined"* or *"veined, bumpy"*) should be presented in transferred results. More importantly, it requires linking concrete objects with their visual concepts, such as *"zebra"* with *"black-and-white stripe"* or *"pineapple"* with *"bumpy yellow."*

We present contrastive language visual artist (CLVA), including language visual artist (LVA) and contrastive reasoning (CR), to perform style transfer conditioning on guided texts. LVA preserves content structures from content images ($\mathcal{C}$) and extracts visual semantics from style instructions ($\mathcal{X}$) to carry out `LDIST`. With the patch-wise style discriminator, it considers patches of style image and transferred result with language to jointly embed style instructions. Furthermore, CR compares contrastive pairs to obtain an additional improvement, where relative content images or style instructions should present similar content structures or style patterns.

Project website: `https://ai-sub.github.io/ldist/`

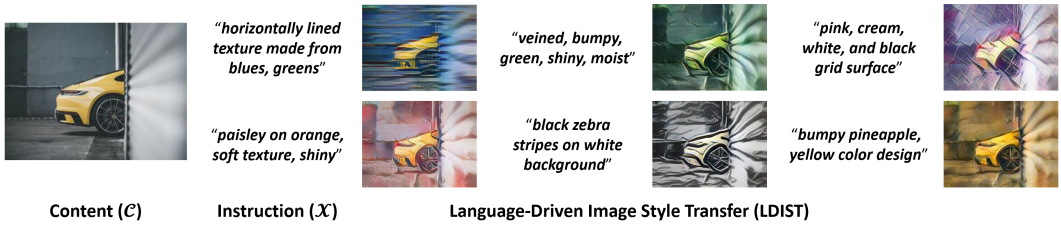

Figure 1: The introduced language-driven image style transfer (LDIST) task. LDIST allows manipulating colors and textures of a content image ($\mathcal{C}$), guided by a style instruction ($\mathcal{X}$).

To evaluate LDIST, we conduct a new dataset built upon DTD[2](Wu et al., 2020), which provides texture images with textual descriptions ($\mathcal{X}$) as reference styles. We also collect numerous wallpapers that present diverse scenes as content images ($\mathcal{C}$). The experiments show that our CLVA is effective for LDIST and achieves superb transferred results on both automatic metrics and human evaluation. Besides, using natural instructions improves the controllability of style transfer and can support partial semantic editing of reference style. In summary, our contributions are four-fold:

- We introduce LDIST that follows natural language to accomplish artistic style transfer;
- We present CLVA, which learns to align visual semantics from style instructions with style images, to provide sufficient style patterns for LDIST;
- For evaluation, we prepare a new dataset containing diverse scenes as content images and descriptions of texture as style instructions;
- Extensive experiments and qualitative examples demonstrate that our CLVA outperforms baselines on both automatic metrics and human evaluation.

2 Related Work

Artistic Style Transfer Style transfer (Gatys et al., 2015b; Li et al., 2017b; Risser et al., 2017; Li et al., 2017a; Ruder et al., 2016; Li & Wand, 2016b; Champandard, 2016; Chen & Hsu, 2016; Mechrez et al., 2018; Liao et al., 2017; Gatys et al., 2017; Atarsaikhan et al., 2017; Castillo et al., 2017; Selim et al., 2016; Jing et al., 2017) is to redraw an image with a specific style. In general, style transfer can be divided into two categories: *photorealistic* and *artistic*. Photorealistic style transfer (Zhang et al., 2017b; Luan et al., 2017; Mechrez et al., 2017; Li et al., 2018; Yoo et al., 2019; Park et al., 2020b) aims at applying reference styles on scenes without hurting details and satisfying contradictory objectives. By contrast, artistic style transfer (Huang et al., 2017a; Gupta et al., 2017; Chen et al., 2017a; Johnson et al., 2016; Ulyanov et al., 2016; 2017; Li & Wand, 2016a; Chen et al., 2018a; Jiang & Fu, 2017; Lu et al., 2017; Azadi et al., 2018; Liu et al., 2017; Wang et al., 2017; Jing et al., 2018; Dumoulin et al., 2017; Chen et al., 2017b;c; Zhang & Dana, 2017; Chen & Schmidt, 2016; Ghiasi et al., 2017; Li et al., 2017c; Huang & Belongie, 2017; Kotovenko et al., 2019; Sanakoyeu et al., 2018; Kolkin et al., 2019; Li et al., 2019) captures style concepts from reference and modifies color distributions and texture patterns of content images. However, it requires preparing numerous style images in advance, which limits practicality of style transfer. To tackle this issue, LDIST allows following textual descriptions to perform *artistic* style transfer.

Language-based Image Editing The general task of LDIST is language-based image editing (LBIE), which also uses language to edit input images. With rule-based instructions and predefined semantic labels, they (Cheng et al., 2013; Laput et al., 2013) first carry out LBIE but under limited practicality. Inspired by text-to-image generation (Reed et al., 2016; Zhang et al., 2017a; Xu et al., 2018), previous works (Chen et al., 2018b; Shinagawa et al., 2017; Dong et al., 2017; Nam et al., 2018; Li et al., 2020; Xia et al., 2021; Liu et al., 2020; El-Nouby et al., 2019; Fu et al., 2020) perform LBIE by conditional GAN. LBIE usually modifies objects and properties of objects in the image. In contrast, LDIST aims at preserving the scene structure from the content image and manipulating the color distribution and the texture pattern from the style instruction.

Contrastive Representation Learning Contrastive learning has been widely used in self-supervised maximizing mutual information (He et al., 2020; Hjelm et al., 2019; Löwe et al., 2019; Misra & van der Maaten, 2020; Wu et al., 2018b). For computer vision, they involve patch-wise comparison (Hénaff et al., 2019; Park et al., 2020a) to provide contrastive loss. For language, both

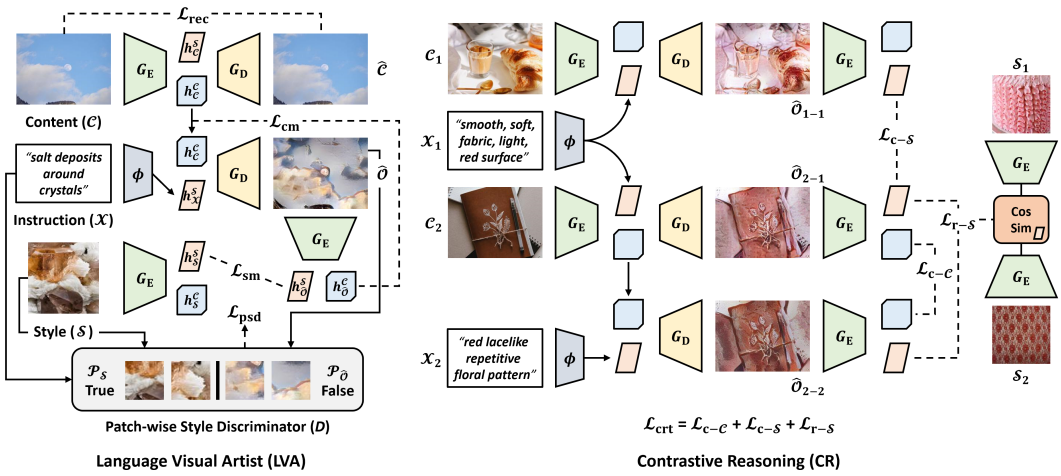

Figure 2: Contrastive language visual artist (CLVA), including language visual artist (LVA) and contrastive reasoning (CR). LVA learns to jointly embed style images ($\mathcal{S}$) and style instructions ($\mathcal{X}$) by the patch-wise style discriminator (D) and perform LDIST for content images ($\mathcal{C}$). CR compares contrasitve pairs ($\{\mathcal{C}_1, \mathcal{X}_1\}$, $\{\mathcal{C}_2, \mathcal{X}_2\}$) to improve the relativeness between transferred results ($\hat{\mathcal{O}}$).

positive and negative contexts are sampled to learn sentence representation (Logeswaran & Lee, 2018; Srinivas et al., 2020). Regarding vision-and-language research, we are the first to apply contrastive learning into image editing by considering contrastive visual-text pairs to improve LDIST.

3 Language-Driven Image Style Transfer (LDIST)

3.1 Overview of CLVA

We introduce the language-driven image style transfer (LDIST) task to manipulate colors and textures of a content image ($\mathcal{C}$), guided by a style instruction ($\mathcal{X}$). As illustrated in Fig. 1, a system is required to extract not only color distribution but also texture patterns from style instructions ($\mathcal{X}$). For training, we have pairs of style image ($\mathcal{S}$) with style instruction ($\mathcal{X}$) to learn the correlation between visual patterns and language semantics. During testing, only style instructions ($\mathcal{X}$) are provided for LDIST to carry out style transfer purely relied on language.

We present contrastive language visual artist (CLVA) in Fig. 2. Language visual artist (LVA) extracts content structures from content images ($\mathcal{C}$) and visual patterns from style instructions ($\mathcal{X}$) to carry out LDIST. LVA adopts the patch-wise style discriminator (D) to connect extracted visual semantics to patches of paired style image ($\mathcal{P}_{\mathcal{S}}$ in Fig. 2). Contrastive reasoning (CR) allows comparing contrastive pairs of content image and style instruction ($\mathcal{C}_1 - \mathcal{X}_1$, $\mathcal{C}_2 - \mathcal{X}_1$, and $\mathcal{C}_2 - \mathcal{X}_2$). In this way, it should present consistent content structures from the same content images ($\mathcal{C}_2$) or analogous style patterns from related style images ($\mathcal{S}_1$ and $\mathcal{S}_2$), despite using different style instructions.

3.2 Language Visual Artist (LVA)

To tackle LDIST, language visual artist (LVA) first adopts visual encoder (G_{E}) to extract the content feature ($h^{\mathcal{C}}$) and the style feature ($h^{\mathcal{S}}$) for an image and text encoder (ϕ) to extract the style instruction feature ($h^{\mathcal{S}}_{\mathcal{X}}$) from an instruction. $h^{\mathcal{C}} \in \mathcal{R}^{h' \times w' \times c}$ is a spatial tensor containing the content structure feature, and $h^{\mathcal{S}} \in \mathcal{R}^{s}$ is a vector representing the global style pattern but without spatial information. $\mathcal{S}^{\mathcal{S}}_{\mathcal{X}} \in \mathcal{R}^{s}$ embeds into the same space of $h^{\mathcal{S}}$ to reflect the extracted visual semantic. Then, visual decoder (G_{D}) produces transferred result ($\hat{\mathcal{O}}$) from $h^{\mathcal{C}}_{\mathcal{C}}$ and $h^{\mathcal{S}}_{\mathcal{X}}$, which performs style transfer for content images by style instructions:

$$
\begin{aligned}
h^{\mathcal{C}}_{\mathcal{C}}, h^{\mathcal{S}}_{\mathcal{C}} &= G_{\text{E}}(\mathcal{C}), \\
h^{\mathcal{S}}_{\mathcal{X}} &= \phi(\mathcal{X}), \\
\hat{\mathcal{O}} &= G_{\text{D}}(h^{\mathcal{C}}_{\mathcal{C}}, h^{\mathcal{S}}_{\mathcal{X}}).
\end{aligned} \tag{1}
$$

In particular, visual decoder (G_{D}) applies self-attention (Zhang et al., 2019a; El-Nouby et al., 2019; Fu et al., 2020) along the channel side to fuse content features ($h^{\mathcal{C}}$) and style features ($h^{\mathcal{S}}$), where $h^{\mathcal{S}}$ is concatenated with $h^{\mathcal{C}}$ along each spatial dimension. There are two goals to train LVA for LDIST: preserving *content structures* from content images and presenting *style patterns* correlated with visual semantics of style instructions.

Structure Reconstruction For content structures, we conduct an auto-encoder (Hinton & Salakhutdinov, 2006) process, where visual decoder (G_{D}) should be able to reconstruct input content images using extracted content features and style features from visual encoder (G_{E}):

$$\begin{aligned} \hat{\mathcal{C}} &= G_{\text{D}}(h^{\mathcal{C}}_{\mathcal{C}}, h^{\mathcal{S}}_{\mathcal{C}}), \\ \mathcal{L}_{\text{rec}} &= ||\hat{\mathcal{C}} - \mathcal{C}||_2. \end{aligned} \tag{2}$$

The reconstruction loss ($\mathcal{L}_{\text{rec}}$) is computed as the mean L2 difference between reconstructed content images ($\hat{\mathcal{C}}$) and input content images ($\mathcal{C}$), where visual encoder learns to preserve content structures.

Patch-wise Style Discriminator (D) Regarding style patterns, results ($\hat{\mathcal{O}}$) guided by style instructions ($\mathcal{X}$) are expected to present analogously to reference style images ($\mathcal{S}$). To address the connection between language semantics from $\mathcal{X}$ and visual semantics from $\mathcal{S}$, we introduce the patch-wise style discriminator (D). Inspired by texture synthesis (Xian et al., 2018; Gatys et al., 2015a), images with analogous patch patterns should appear perceptually similar texture patterns. D tries to recognize the correspondence between an image patch ($\mathcal{P}$) and a style instruction ($\mathcal{X}$):

$$\begin{aligned} \mathcal{P}_{\hat{\mathcal{O}}}, \mathcal{P}_{\mathcal{S}} &= \text{Crop}(\hat{\mathcal{O}}), \text{Crop}(\mathcal{S}), \\ \mathcal{L}_{\text{psd}} &= \log(1 - D(\mathcal{P}_{\hat{\mathcal{O}}}, \mathcal{X})), \\ \mathcal{L}_{D} &= \log(1 - D(\mathcal{P}_{\hat{\mathcal{O}}}, \mathcal{X})) + \log(D(\mathcal{P}_{\mathcal{S}}, \mathcal{X})), \end{aligned} \tag{3}$$

where Crop is to randomly crop an image ($1/8$ on each side) into patches. With the patch-wise style loss ($\mathcal{L}_{\text{psd}}$), we aim at generating transferred results that are correlated with style instructions. On the other hand, by the discriminator loss ($\mathcal{L}_D$), D learns to distinguish that patches from style images ($\mathcal{P}_{\mathcal{S}}$) are true cases, and patches from transferred results ($\mathcal{P}_{\hat{\mathcal{O}}}$) should be false cases. This adversarial loss (Goodfellow et al., 2014; Salehi et al., 2020) enforces that style patterns from style instructions are presented similarly with style images in transferred results, which jointly embeds the extracted visual semantic between each other.

Content Matching and Style Matching To further enhance the alignment with inputs, inspired by cycle consistency (Wang et al., 2019b; Zhu et al., 2017; Wu et al., 2018a; Dwibedi et al., 2019; Qiao et al., 2019; Yi et al., 2017; Xia et al., 2016), we consider the content matching loss ($\mathcal{L}_{\text{cm}}$) and the style matching loss ($\mathcal{L}_{\text{sm}}$) of transferred results ($\hat{\mathcal{O}}$). We adopt visual encoder (G_{E}) again to extract content features ($h^{\mathcal{C}}_{\hat{\mathcal{O}}}$) and style features ($h^{\mathcal{S}}_{\hat{\mathcal{O}}}$) of $\hat{\mathcal{O}}$, where $h^{\mathcal{C}}_{\hat{\mathcal{O}}}$ and $h^{\mathcal{S}}_{\hat{\mathcal{O}}}$ should correlate with content features ($h^{\mathcal{C}}_{\mathcal{C}}$) of content image and style features ($h^{\mathcal{S}}_{\mathcal{S}}$) of reference style image ($\mathcal{S}$):

$$\begin{aligned} (h^{\mathcal{C}}_{\hat{\mathcal{O}}}, h^{\mathcal{S}}_{\hat{\mathcal{O}}}), (h^{\mathcal{C}}_{\mathcal{S}}, h^{\mathcal{S}}_{\mathcal{S}}) &= G_{\text{E}}(\hat{\mathcal{O}}), G_{\text{E}}(S), \\ \mathcal{L}_{\text{cm}}, \mathcal{L}_{\text{sm}} &= ||h^{\mathcal{C}}_{\hat{\mathcal{O}}} - h^{\mathcal{C}}_{\mathcal{C}}||_2, ||h^{\mathcal{S}}_{\hat{\mathcal{O}}} - h^{\mathcal{S}}_{\mathcal{S}}||_2. \end{aligned} \tag{4}$$

Therefore, transferred results are required to align with content structures and style patterns from inputs, which meets the goal of LDIST.

3.3 Contrastive Reasoning (CR)

The above LVA process considers LDIST with a single pair of content image and style instruction. However, a content image should be able to transfer to various styles while preserving the same content structure. Moreover, related style instructions can apply analogous style patterns to all kinds of content images. As shown in Fig. 2, contrastive reasoning (CR) compares content structures or style patterns from transferred results of contrastive pair. A contrastive pair consists of two different content images ($\mathcal{C}_1$ and $\mathcal{C}_2$ in Fig. 2) with two reference styles ($\{\mathcal{S}_1, \mathcal{X}_1\}$ and $\{\mathcal{S}_2, \mathcal{X}_2\}$). We follow the LVA inference to acquire cross results for pairs of content image and style instruction:

$$\begin{aligned} (h^{\mathcal{C}}_{\mathcal{C}_1}, h^{\mathcal{S}}_{\mathcal{C}_1}), (h^{\mathcal{C}}_{\mathcal{C}_2}, h^{\mathcal{S}}_{\mathcal{C}_2}) &= G_{\text{E}}(\mathcal{C}_1), G_{\text{E}}(\mathcal{C}_2), \\ h^{\mathcal{S}}_{\mathcal{X}_1}, h^{\mathcal{S}}_{\mathcal{X}_2} &= \phi(\mathcal{X}_1), \phi(\mathcal{X}_2) \\ \hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}, \hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_2}, \hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}, \hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2} &= G_{\text{D}}(h^{\mathcal{C}}_{\mathcal{C}_1}, h^{\mathcal{S}}_{\mathcal{X}_1}), G_{\text{D}}(h^{\mathcal{C}}_{\mathcal{C}_1}, h^{\mathcal{S}}_{\mathcal{X}_2}), G_{\text{D}}(h^{\mathcal{C}}_{\mathcal{C}_2}, h^{\mathcal{S}}_{\mathcal{X}_1}), G_{\text{D}}(h^{\mathcal{C}}_{\mathcal{C}_2}, h^{\mathcal{S}}_{\mathcal{X}_2}). \end{aligned}$$

Algorithm 1 Learning of Language Visual Artist (LVA)

G_{E}, G_{D}: Visual Encoder, Visual Decoder
ϕ: Text Encoder
D: Patch-wise Style Discriminator
while TRAIN_VLA **do**
  $\mathcal{C}, \{\mathcal{S}, \mathcal{X}\} \leftarrow$ Sampled content/style

  $h^{\mathcal{C}}_{\mathcal{C}}, h^{\mathcal{S}}_{\mathcal{C}} \leftarrow G_{\mathrm{E}}(\mathcal{C})$ $\hat{\mathcal{C}} \leftarrow G_{\mathrm{D}}(h^{\mathcal{C}}_{\mathcal{C}}, h^{\mathcal{S}}_{\mathcal{C}})$
  $\mathcal{L}_{\text{rec}} \leftarrow$ Reconstruction loss ▷ Eq. 2
  $h^{\mathcal{S}}_{\mathcal{X}} \leftarrow \phi(\mathcal{X})$ $\hat{\mathcal{O}} \leftarrow G_{\mathrm{D}}(h^{\mathcal{C}}_{\mathcal{C}}, h^{\mathcal{S}}_{\mathcal{X}})$ $\mathcal{P}_{\mathcal{S}}, \mathcal{P}_{\hat{\mathcal{O}}} \leftarrow \text{Crop}(\mathcal{S}), \text{Crop}(\hat{\mathcal{O}})$
  $\mathcal{L}_{\text{psd}} \leftarrow$ Patch-wise style loss ▷ Eq. 3
  $(h^{\mathcal{C}}_{\hat{\mathcal{O}}}, h^{\mathcal{S}}_{\hat{\mathcal{O}}}), (h^{\mathcal{C}}_{\mathcal{S}}, h^{\mathcal{S}}_{\mathcal{S}}) \leftarrow G_{\mathrm{E}}(\hat{\mathcal{O}}), G_{\mathrm{E}}(\mathcal{S})$
  $\mathcal{L}_{\text{cm}}, \mathcal{L}_{\text{sm}} \leftarrow$ Content matching loss, Style matching loss ▷ Eq. 4

  $\mathcal{L}_G \leftarrow \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{psd}} + \mathcal{L}_{\text{cm}} + \mathcal{L}_{\text{sm}}$
  $\mathcal{L}_D \leftarrow$ Discriminator loss for D ▷ Eq. 3
  Update G_{E}, G_{D}, ϕ by minimizing $\mathcal{L}_G$
  Update D by maximizing $\mathcal{L}_D$
end while

Consistent Matching The transfer results should present similar content structures ($\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}$ and $\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}$) or analogous style patterns ($\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}$ and $\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}$) if using the same content image ($\mathcal{C}_2$) or the same style instruction ($\mathcal{X}_1$):

$$
\begin{aligned}
(h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}}, h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}}), (h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_2}}, h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_2}}) &= G_{\mathrm{E}}(\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}), G_{\mathrm{E}}(\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_2}), \\
(h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}}, h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}}), (h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}}, h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}}) &= G_{\mathrm{E}}(\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}), G_{\mathrm{E}}(\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}), \\
\mathcal{L}_{\mathrm{c}-\mathcal{C}} &= ||h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}} - h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_2}}||_2 + ||h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}} - h^{\mathcal{C}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}}||_2, \\
\mathcal{L}_{\mathrm{c}-\mathcal{S}} &= ||h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}} - h^{\mathcal{S}}_{\hat{\mathcal{S}}_{2-1}}||_2 + ||h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_2}} - h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}}||_2,
\end{aligned}
\tag{5}
$$

where *consistent matching* of content structure ($\mathcal{L}_{\mathrm{c}-\mathcal{C}}$) or style pattern ($\mathcal{L}_{\mathrm{c}-\mathcal{S}}$) is aligned by content features or style features of transferred results, extracted by the visual encoder (G_{E}).

Relative Matching Apart from consistent matching, distinct style instructions, which imply corresponding visual semantics, should still present relative style patterns in transferred results. For example, we can only discover *"red, repetitive, floral"* literally from $\mathcal{X}_2$. However, if comparing reference style images ($\mathcal{S}_1$ and $\mathcal{S}_2$), we can perceive that they share a similar texture pattern and link the visual concept of *"lacelike"* from $\mathcal{X}_2$ to *"smooth, soft, fabric"* from $\mathcal{X}_1$. We define *relative matching* ($\mathcal{L}_{\mathrm{r}-\mathcal{S}}$) with the cosine similarity (CosSim) between reference style images as the weight:

$$
\begin{aligned}
(h^{\mathcal{C}}_{\mathcal{S}_1}, h^{\mathcal{S}}_{\mathcal{S}_1}), (h^{\mathcal{C}}_{\mathcal{S}_2}, h^{\mathcal{S}}_{\mathcal{S}_2}) &= G_{\mathrm{E}}(\mathcal{S}_1), G_{\mathrm{E}}(\mathcal{S}_2), \\
\mathcal{L}_{\mathrm{r}-\mathcal{S}} &= (||h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}} - h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_2}}||_2 + ||h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}} - h^{\mathcal{S}}_{\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}}||_2) \cdot \text{CosSim}(h^{\mathcal{S}}_{\mathcal{S}_1}, h^{\mathcal{S}}_{\mathcal{S}_2}).
\end{aligned}
\tag{6}
$$

When style images are related, it has to align style features to certain extent even if paired style instructions are different. Otherwise, $\mathcal{L}_{\mathrm{r}-\mathcal{S}}$ will be close to 0 and ignore this unrelated style pair. The overall contrastive reasoning loss ($\mathcal{L}_{\text{crt}}$) considers both consistent matching and relative matching:

$$
\mathcal{L}_{\text{crt}} = \mathcal{L}_{\mathrm{c}-\mathcal{C}} + \mathcal{L}_{\mathrm{c}-\mathcal{S}} + \mathcal{L}_{\mathrm{r}-\mathcal{S}}. \tag{7}
$$

3.4 Learning of CLVA

For each epoch of training, we first train with the LVA process and then CR for our CLVA. Algo. 1 presents the learning process of VLA with the patch-wise style discriminator (D). We consider the reconstruction loss ($\mathcal{L}_{\text{rec}}$) to preserve content structure and the patch-wise style loss ($\mathcal{L}_{\text{psd}}$) between style instruction and visual pattern of transferred results. Both content matching loss ($\mathcal{L}_{\text{cm}}$) and style matching loss ($\mathcal{L}_{\text{sm}}$) are applied to enhance the matching with the inputs. We minimize $\mathcal{L}_G$, the total of above, to train VLA. At the same time, we update D by maximizing the discriminator loss

($\mathcal{L}_D$) to distinguish true (from style images ($\mathcal{P}_{\mathcal{S}}$)) or false (from transferred results ($\mathcal{P}_{\hat{\mathcal{O}}}$)) patches, with respect to style instructions. During CR, contrastive pairs of content image ($\mathcal{C}_1$ and $\mathcal{C}_2$) and style instruction ($\mathcal{X}_1$ and $\mathcal{X}_2$) are randomly sampled. The transferred results are produced across the contrastive pair. For example, $\hat{\mathcal{O}}_{\mathcal{C}_1-\mathcal{X}_1}$ and $\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}$ are from the same $\mathcal{X}_1$, and $\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_1}$ and $\hat{\mathcal{O}}_{\mathcal{C}_2-\mathcal{X}_2}$ are from the same $\mathcal{C}_2$. We further update by minimizing the contrastive reasoning loss ($\mathcal{L}_{\text{crt}}$) to allow considering content structure consistency and style relativeness from contrastive transferred results. Therefore, the overall optimization of CLVA can be summarized as:

$$\begin{aligned} \mathcal{L}_G =& \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{psd}} + \mathcal{L}_{\text{cm}} + \mathcal{L}_{\text{sm}}, \\ \min_{G,\phi} \max_{D}\ & \mathcal{L}_G + \mathcal{L}_D + \mathcal{L}_{\text{crt}}. \end{aligned} \tag{8}$$

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Dataset To evaluate our contrastive language visual artist (CLVA), we build a new dataset for language-driven image style transfer (`LDIST`). We collect 14,924 wallpapers from WallpapersCraft[1], which presents diverse scenes as content images ($\mathcal{C}$). Each content image is resized to 256x192 in our experiment. For style instructions, the DTD[2] (Wu et al., 2020) dataset supplies 5,368 pairs of texture image and natural description. We consider texture images (Cimpoi et al., 2014) as style images ($\mathcal{S}$) and related descriptions as style instructions ($\mathcal{X}$), illustrated in Fig. 3. We randomly sample 2,500 pairs of unseen content image and unseen style instruction, which are not accessible for training, to evaluate the generalizability of `LDIST` during testing. Note that both style images and style instructions appear for training, but only style instructions are provided during testing to perform style transfer guided by language.

Evaluation Metrics To support large-scale evaluation, we treat style transfer results from style images as the semi-ground truth (Semi-GT) (Al-Sarraf et al., 2014; Borkar et al., 2010; Salvo, 2013) by FastStyleTransfer (Hub, 2021). We apply the following metrics to compare the visual similarity between `LDIST` results and Semi-GT:

- **Structural Similarity Index Measure (SSIM)**: SSIM (Wang et al., 2004) compares images in the luminance, contrast, and structure aspects. A higher SSIM has a higher structural similarity;
- **Perceptual Loss (Percept)**: Percept (Johnson et al., 2016) computes from the gram matrix of visual features. A lower Percept difference shows that two results share the similar style pattern;
- **Frechet Activitation Distance (FAD)**: Inspired from FID (Heusel et al., 2017), FAD is computed by the mean L2 distance of the activations from the Inception V3 (Szegedy et al., 2016) feature. As a distance metric, a lower FAD represents that `LDIST` results and Semi-GT are more relevant.

Apart from visual similarity, we also consider the correlation between style instructions and `LDIST` results. We adopt CLIP (Shi et al., 2020; Wu et al., 2021) that provides visual-text matching:

- **Vision-and-Language Similarity (VLS)**: VLS calculates the cosine similarity between the joint embedding of the instruction ($\mathcal{X}$) and the transferred result ($\hat{\mathcal{O}}$) from CLIP;
- **Relative Similarity (RS)**: Since VLS only provides an absolute score of semantic matching, we regard the relative VLS with the Semi-GT ($\mathcal{O}$) as RS to reduce the influence from CLIP;

$$\text{VLS}(\hat{\mathcal{O}}, \mathcal{X}) = \text{CosSim}(\text{CLIP}(\hat{\mathcal{O}}), \text{CLIP}(\mathcal{X})), \text{RS}(\hat{\mathcal{O}}, \mathcal{O}) = \frac{\text{VLS}(\hat{\mathcal{O}}, \mathcal{X})}{\text{VLS}(\mathcal{O}, \mathcal{X})}. \tag{9}$$

We also conduct human evaluation in Sec. 4.2 to investigate the quality from the human aspect.

Baselines Since being a brand new task, there is no existing baseline on our `LDIST`. We consider typical arbitrary style transfer methods, where a single trained model should support any content images and style images, as the compared baselines. Style instructions are jointly embedded with style features for training and directly serves as the provided style during testing.

- **NST** (Gatys et al., 2015b): NST adopts the gram matrix as the style feature and requires numerous iterations of optimization to acquire the transferred result;

[1] WallpapersCraft: `https://wallpaperscraft.com/`

Method	Automatic Metrics (*vs*. Semi-GT)					Human Evaluation (↑)	
	SSIM (↑)	Percept (↓)	FAD (↓)	VLS (↑)	RS (↑)	*vs*. Instruction	*vs*. Style
NST	12.121	0.11728	0.17647	21.973	96.853	491	490
WCT	36.114	0.05073	0.13164	22.265	98.078	534	547
AdaIN	57.228	0.02747	0.11731	22.475	98.261	594	597
CLVA	**60.586**	**0.02076**	**0.11318**	**22.785**	**98.798**	**631**	**616**

Table 1: Testing results on automatic metrics and human evaluation (accumulated ranking points).

- **WCT** (Li et al., 2017c): WCT applies an encoder-decoder architecture for style transfer, where the whitening transform and the coloring transform is to match the covariance of the style feature;
- **AdaIn** (Huang & Belongie, 2017): AdaIn is also a feed-forward-based method that incorporates the adaptive instance normalization to fuse the content feature and the style feature.

Implementation Detail Visual encoder (G_{E}) contains 4 layers of downsampling ResBlock (He et al., 2015) to extract content feature and style feature. Particularly, style feature is acquired from a dense layer after average pooling to represent a global style pattern without spatial information. To understand style instructions, text encoder (ϕ) first adopts RoBERTa (Liu et al., 2019; Reimers & Gurevych, 2019) for a general language representation, and then a dense layer to jointly embed with style feature. During self-attention in visual decoder (G_{D}), we first shrink the size of the CNN channel down to 64 (Zhang et al., 2019a), but expand into the same input size 256 as the output. G_{D} is built upon 4 upsampling ResBlocks and a convolution layer with output feature 3 to produce RGB results. The patch-wise style discriminator (D) follows a similar architecture, where self-attention with a dense layer determines the correlation between instruction feature and patch style feature (El-Nouby et al., 2019; Fu et al., 2020) from ϕ and G_{E}. We adopt Adam (Kingma & Ba, 2015) to optimize the entire CLVA with learning rate 3e-4 for $\mathcal{L}_G$, 1e-4 for $\mathcal{L}_D$, and 3e-5 for $\mathcal{L}_{\mathrm{crt}}$.

4.2 Quantitative Results

Automatic Evaluation Table 1 shows the automatic evaluation results on LDIST. For NST (Gatys et al., 2015b), although style instructions are jointly embedded with gram matrices, it cannot provide sufficient style features and leads to poor transferred results (low 12.1 SSIM and high 0.117 Percept). Both WCT (Li et al., 2017c) and AdaIN (Huang & Belongie, 2017) relies on predefined style statistics, which makes the style transform quite limited (low 36.1 SSIM of WCT and high 0.117 FAD of AdaIn). Our CLVA, which adopts contrastive reasoning to compare pairs of content image and style instruction, achieves the best LDIST results with the highest 60.6 SSIM, the lowest 0.021 Percept, and the lowest 0.113 FAD. A similar trend can be found on the visual-text matching evaluation. NST and WCT are both subject to limited style features from style instructions and result in lower VLS and RS. On the other hand, with the patch-wise discriminator between style images and transferred results from style instructions, our CLVA can obtain LDIST results that are more correlated with style instructions (the highest 22.8 VLS and the highest 98.8 RS).

Human Evaluation Apart from automatic metrics, we also investigate the quality of LDIST results from the human aspect. Table 1 demonstrates human evaluation between baselines and our CLVA. We randomly sample 75 results from pairs of content image and style instruction. MTurkers from Amazon Mechanical Turks[2] (AMT) rank the correlation of the LDIST result from each method between the style instruction (*vs*. Instruction) or the style image (*vs*. Style), where rank 1 gets 4 points, rank 2 gets 3 points, and so on. Each example is assigned to 3 different MTurkers to avoid evaluation bias. At first, our CLVA acquires the highest points (631 points in total) regarding style instructions (*vs*. Instruction), which indicates that transferred results from CLVA are the most corresponding for humans. Concerning style images (*vs*. Style), we still obtain the highest 616 points and represent that CLVA is not only corresponding to style instructions but also correlated with style patterns. Last but not least, we discover that human evaluation results follow a similar trend to the results of automatic metrics. It shows that our conducted automatic metrics are aligned with the human aspect and can provide a reasonable large-scale evaluation for LDIST.

[2]Amazon Mechanical Turks: https://www.mturk.com/

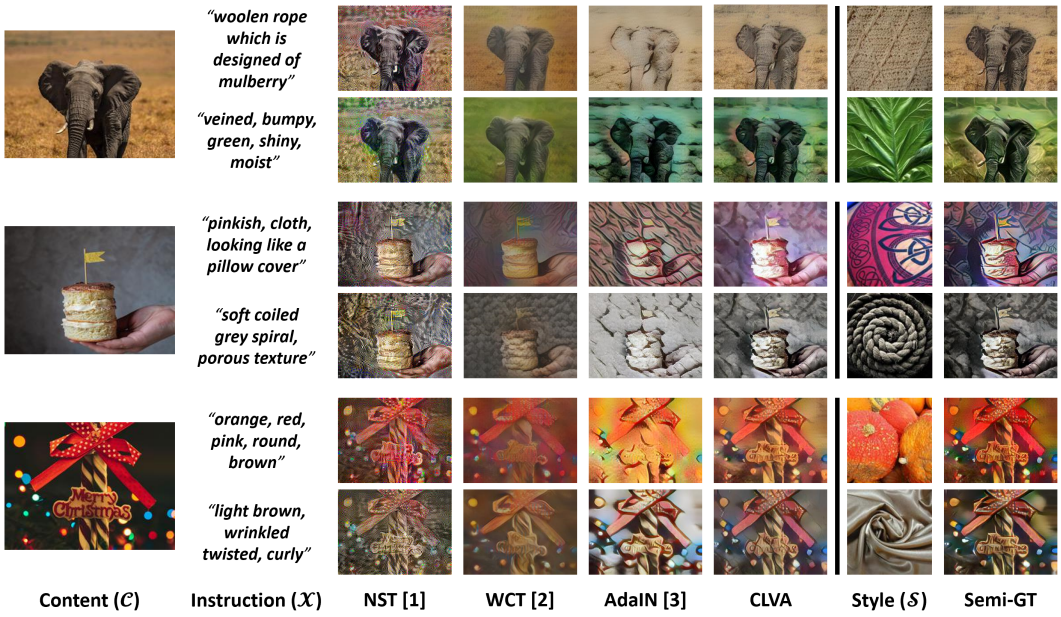

Figure 3: Visualization of baselines and our CLVA on language-driven image style transfer (LDIST).

	Ablation Settings				Automatic Metrics (*vs.* Semi-GT)				
	$\mathcal{L}_{\text{rec}}$+$\mathcal{L}_{\text{psd}}$	$\mathcal{L}_{\text{cm}}$	$\mathcal{L}_{\text{sm}}$	$\mathcal{L}_{\text{cst}}$	SSIM (↑)	Percept (↓)	FAD (↓)	VLS (↑)	RS (↑)
(a)	✗ (Warm-Up from AdaIn)				57.258	0.02798	0.11733	22.552	98.437
(b)	✓	✗	✗	✗	57.874	0.02576	0.11566	22.567	98.474
(c)	✓	✓	✗	✗	59.244	0.02927	0.11559	22.589	98.554
(d)	✓	✗	✓	✗	58.059	0.02318	0.11410	22.602	98.566
(e)	✓	✓	✓	✗	59.491	0.02244	0.11399	22.604	98.597
(f)	✓	✓	✓	✓	**60.586**	**0.02076**	**0.11318**	**22.785**	**98.798**

Table 2: Ablation study with reconstruction ($\mathcal{L}_{\text{rec}}$), patch-wise style discriminator ($\mathcal{L}_{\text{psd}}$), content matching ($\mathcal{L}_{\text{cm}}$), style matching ($\mathcal{L}_{\text{sm}}$), and contrastive reasoning ($\mathcal{L}_{\text{cst}}$) of CLVA.

4.3 Ablation Study

We conduct an ablation study to present the effect of each component in Table 2. At row (a), our CLVA is purely pretrained on AdaIn results as a warm-up. It slightly overpasses AdaIn, which shows that our self-attention fusion between content images and style instructions has stronger generalizability than the adaptive instance normalization on LDIST. With the reconstruction loss ($\mathcal{L}_{\text{rec}}$) and the patch-wise style discriminator ($\mathcal{L}_{\text{psd}}$) at row (b), CLVA can transfer more concrete structures and more correlated style patterns with respect to content images and style instructions. We then discuss the strength of content matching ($\mathcal{L}_{\text{cm}}$) and style matching ($\mathcal{L}_{\text{sm}}$) at row (c)-(e). In particular, content matching helps the scene similarity to content images (higher 59.2 SSIM at row (c)). For style matching, it aims at producing analogous visual patterns to style images, which leads to better style quality (lower 0.023 Percept and higher 22.6 VLS at row (d)). If considering altogether at row (e), it can benefit from both. In the end, contrastive reasoning ($\mathcal{L}_{\text{crt}}$) further enables CLVA to consider contrastive pairs, making a comprehensive improvement on LDIST at row (f).

Inference Efficiency Table 3 shows the inference time (in second) about running LDIST on GPU (NVIDIA TITAN X) with different batch sizes (BS). NST, which relies on numerous iterations, executes with low efficiency. Despite being an encoder-decoder architecture, the coloring transform in WCT takes an overhead when searching the correlation. In contrast, both AdaIN and our CLVA go through a feed-forward process (about 100 FPS). CLVA even achieves 180 FPS with parallelization of 50 examples (with image size 256x192).

Method	BS=1	BS=50
NST	40.40	1,035
WCT	0.159	4.796
AdaIN	0.010	0.336
CLVA	**0.009**	**0.268**

Table 3: Inference time cost.

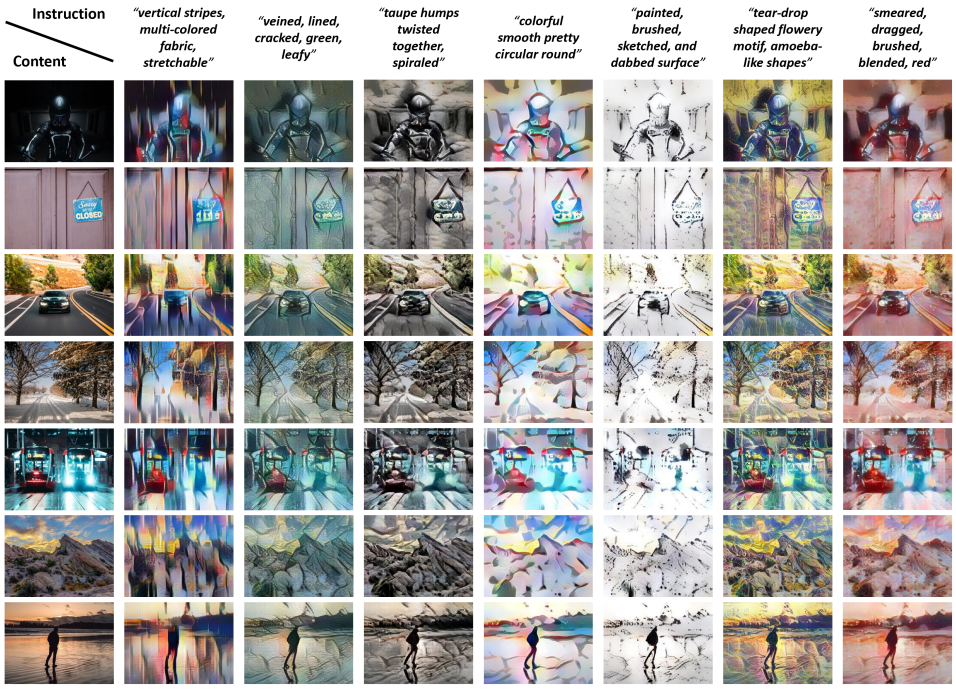

Figure 5: Visualization of CLVA results[3] on diverse pairs of content image and style instruction.

Qualitative Results Fig. 3 demonstrates the visualization results of baselines and our CLVA on LDIST, along with the semi-ground truth (Semi-GT). For NST, the pixels are broken and lead to poor style quality of transferred results. WCT merely picks color information from style instructions, where it only transfers color distribution but not style patterns. AdaIn produces better results, but the objects in content images sometimes become vague. With the patch-wise style discriminator and contrastive reasoning, our CLVA captures not only color distribution but also texture patterns that are correlated with style instructions. For example, the first row of the second content image, though our CLVA result is not that matching with the Semi-GT, it presents a corresponding pink appearance with a soft texture as the assigned *"pinkish pillow covers"* in the style instruction. Fig. 5 illustrates more visualization results[3] by our CLVA on diverse pairs of content image and style instruction.

Fine-grained Control via Partial Semantic Editing Furthermore, LDIST allows fine-grained control of transferred styles via partial semantic editing. As shown in Fig. 4, we can easily modify the language prompts to control the style semantic. For example, we can manipulate the color distribution from *"green"* to *"orange"* or the texture pattern from *"veined, bumpy"* to *"fabric, metallic"*. Then, the related semantics of LDIST results would be changed correspondingly by the edited style instructions.

Figure 4: Partial semantic editing.

5 Conclusion

We introduce language-driven image style transfer (LDIST) to manipulate colors and textures of a content image by a style instruction. We propose contrastive language visual artist (CLVA) that adopts the patch-wise style discriminator and contrastive reasoning to jointly learn between style images and style instructions. The experiments show that CLVA outperforms baselines on both automatic metrics and human evaluation, and using guided language can improve accessibility without preparing style images. LDIST also supports partial semantic editing, which makes visual applications like image/video effect more controllable for humans.

[3]Please visit project website for more visualization results: https://ai-sub.github.io/ldist/

REFERENCES

Ali Al-Sarraf, Bok-Suk Shin, Zezhong Xu, and Reinhard Klette. Ground Truth and Performance Evaluation of Lane Border Detection. In *ICCVG*, 2014.

Gantugs Atarsaikhan, Brian Kenji Iwana, Atsushi Narusawa, Keiji Yanai, and Seiichi Uchida. Neural Font Style Transfer. In *ICDAR*, 2017.

Samaneh Azadi, Matthew Fisher, Vladimir Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. Multi-Content GAN for Few-Shot Font Style Transfer. In *CVPR*, 2018.

Amol Borkar, Monson Hayes, and Mark T. Smith. An Efficient Method to Generate Ground Truth for Evaluating Lane Detection Systems. In *ICASSP*, 2010.

Carlos Castillo, Soham De, Xintong Han, Bharat Singh, Abhay Kumar Yadav, and Tom Goldstein. Son of Zorn's Lemma: Targeted Style Transfer Using Instance-aware Semantic Segmentation. In *ICASSP*, 2017.

Alex J. Champandard. Semantic Style Transfer and Turning Two-Bit Doodles into Fine Artworks. In *arXiv:1603.01768*, 2016.

Dongdong Chen, Jing Liao, Lu Yuan, Nenghai Yu, , and Gang Hua. Coherent Online Video Style Transfer. In *ICCV*, 2017a.

Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. StyleBank: An Explicit Representation for Neural Image Style Transfer. In *CVPR*, 2017b.

Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. Diversified Texture Synthesis with Feed-forward Networks. In *CVPR*, 2017c.

Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. Stereoscopic Neural Style Transfer. In *CVPR*, 2018a.

Jianbo Chen, Yelong Shen, Jianfeng Gao, Jingjing Liu, and Xiaodong Liu. Language-Based Image Editing with Recurrent Attentive Models. In *CVPR*, 2018b.

Tian Qi Chen and Mark Schmidt. Fast Patch-based Style Transfer of Arbitrary Style. In *NeurIPS WS*, 2016.

Yi-Lei Chen and Chiou-Ting Hsu. Towards Deep Style Transfer: A Content-Aware Perspective. In *BMVC*, 2016.

Ming-Ming Cheng, Shuai Zheng, Wen-Yan Lin, Jonathan Warrell, Vibhav Vineet, Paul Sturgess, Nigel Crook, Niloy Mitra, and Philip Torr. ImageSpirit: Verbal Guided Image Parsing. In *ACM Transactions on Graphics*, 2013.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing Textures in the Wild. In *CVPR*, 2014.

Hao Dong, Simiao Yu, Chao Wu, and Yike Guo. Semantic Image Synthesis via Adversarial Learning. In *ICCV*, 2017.

Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A Learned Representation For Artistic Style. In *ICLR*, 2017.

Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Cycle-Consistent Deep Generative Hashing for Cross-Modal Retrieval. In *CVPR*, 2019.

Alaaeldin El-Nouby, Shikhar Sharma, Hannes Schulz, Devon Hjelm, Layla El Asri, Samira Ebrahimi Kahou, Yoshua Bengio, and Graham W.Taylor. Tell, Draw, and Repeat: Generating and Modifying Images Based on Continual Linguistic Instruction. In *ICCV*, 2019.

Tsu-Jui Fu, Xin Eric Wang, Scott Grafton, Miguel Eckstein, and William Yang Wang. SSCR: Iterative Language-Based Image Editing via Self-Supervised Counterfactual Reasoning. In *EMNLP*, 2020.

Chang Gao, Derun Gu, Fangjun Zhang, and Yizhou Yu. ReCoNet: Real-time Coherent Video Style Transfer Network. In *arXiv:1807.01197*, 2018.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Texture Synthesis Using Convolutional Neural Networks. In *NeurIPS*, 2015a.

Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A Neural Algorithm of Artistic Style. In *arXiv:1508.06576*, 2015b.

Leon A. Gatys, Alexander S. Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Controlling Perceptual Factors in Neural Style Transfer. In *CVPR*, 2017.

Golnaz Ghiasi, Honglak Lee, Manjunath Kudlur, Vincent Dumoulin, and Jonathon Shlens. Exploring the Structure of a Real-Time, Arbitrary Neural Artistic Stylization Network. In *BMVC*, 2017.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. In *NeurIPS*, 2014.

Agrim Gupta, Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Characterizing and Improving Stability in Neural Style Transfer. In *ICCV*, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *ICCV*, 2015.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *CVPR*, 2020.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *NeurIPS*, 2017.

Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. In *Science*, 2006.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning Deep Representations by Mutual Information Estimation and Maximization. In *ICLR*, 2019.

Haozhi Huang, Hao Wang, Wenhan Luo, Lin Ma, Wenhao Jiang, Xiaolong Zhu, Zhifeng Li, and Wei Liu. Real-Time Neural Style Transfer for Videos. In *CVPR*, 2017a.

Haozhi Huang, Hao Wang, Wenhan Luo, Lin Ma, Wenhao Jiang, Xiaolong Zhu, Zhifeng Li, and Wei Liu. Real-Time Neural Style Transfer for Videos. In *CVPR*, 2017b.

Xun Huang and Serge Belongie. Arbitrary Style Transfer in Real-time with Adaptive Instance Normalization. In *ICCV*, 2017.

TensorFlow Hub. Fast Style Transfer for Arbitrary Styles. 2021.

Olivier J. Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Aaron van den Oord. Data-Efficient Image Recognition with Contrastive Predictive Coding. In *CVPR*, 2019.

Shuhui Jiang and Yun Fu. Fashion Style Generator. In *IJCAI*, 2017.

Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song. Neural Style Transfer: A Review. In *arXiv:1705.04058*, 2017.

Yongcheng Jing, Yang Liu, Yezhou Yang, Zunlei Feng, Yizhou Yu, Dacheng Tao, and Mingli Song. Stroke Controllable Fast Style Transfer with Adaptive Receptive Fields. In *ECCV*, 2018.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. In *ECCV*, 2016.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR*, 2015.

Nicholas Kolkin, Jason Salavon, and Greg Shakhnarovich. Style Transfer by Relaxed Optimal Transport and Self-Similarity. In *CVPR*, 2019.

Dmytro Kotovenko, Artsiom Sanakoyeu, Pingchuan Ma, Sabine Lang, and Björn Ommer. A Content Transformation Block For Image Style Transfer. In *CVPR*, 2019.

Gierad Laput, Mira Dontcheva, Gregg Wilensky, Walter Chang, Aseem Agarwala, Jason Linder, and Eytan Adar. PixelTone: A Multimodal Interface for Image Editing. In *CHI*, 2013.

Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip H. S. Torr. ManiGAN: Text-Guided Image Manipulation. In *CVPR*, 2020.

Chuan Li and Michael Wand. Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks. In *ECCV*, 2016a.

Chuan Li and Michael Wand. Combining Markov Random Fields and Convolutional Neural Networks for Image Synthesis. In *CVPR*, 2016b.

Shaohua Li, Xinxing Xu, Liqiang Nie, and Tat-Seng Chua. Laplacian-Steered Neural Style Transfer. In *ACMMM*, 2017a.

Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. Learning Linear Transformations for Fast Arbitrary Style Transfer. In *CVPR*, 2019.

Yanghao Li, Naiyan Wang, Jiaying Liu, and Xiaodi Hou. Demystifying Neural Style Transfer. In *IJCAI*, 2017b.

Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal Style Transfer via Feature Transforms. In *NeurIPS*, 2017c.

Yijun Li, Ming-Yu Liu, Xueting Li, Ming-Hsuan Yang, and Jan Kautz. A Closed-form Solution to Photorealistic Image Stylization. In *ECCV*, 2018.

Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. Visual Attribute Transfer through Deep Image Analogy. In *SIGGRAPH*, 2017.

Xiao-Chang Liu, Ming-Ming Cheng, Yu-Kun Lai, and Paul L. Rosin. Depth-Aware Neural Style Transfer. In *SNPAR*, 2017.

Xihui Liu, Zhe Lin, Jianming Zhang, Handong Zhao, Quan Tran, Xiaogang Wang, and Hongsheng Li. Open-Edit: Open-Domain Image Manipulation with Open-Vocabulary Instructions. In *ECCV*, 2020.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. In *arXiv:1907.11692*, 2019.

Lajanugen Logeswaran and Honglak Lee. An Efficient Framework for Learning Sentence Representations. In *ICLR*, 2018.

Ming Lu, Hao Zhao, Anbang Yao, Feng Xu, Yurong Chen, and Li Zhang. Decoder Network over Lightweight Reconstructed Feature for Fast SemanticStyle Transfer. In *ICCV*, 2017.

Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep Photo Style Transfer. In *CVPR*, 2017.

Sindy Löwe, Peter O'Connor, and Bastiaan S. Veeling. Putting An End to End-to-End: Gradient-Isolated Learning of Representations. In *NeurIPS*, 2019.

Roey Mechrez, Eli Shechtman, and Lihi Zelnik-Manor. Photorealistic Style Transfer with Screened Poisson Equation. In *BMVC*, 2017.

Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The Contextual Loss for Image Transformation with Non-Aligned Data. In *ECCV*, 2018.

Ishan Misra and Laurens van der Maaten. Self-Supervised Learning of Pretext-Invariant Representations. In *CVPR*, 2020.

Seonghyeon Nam, Yunji Kim, and Seon Joo Kim. Text-Adaptive Generative Adversarial Networks: Manipulating Images with Natural Language. In *NeurIPS*, 2018.

Taesung Park, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive Learning for Unpaired Image-to-Image Translation. In *ECCV*, 2020a.

Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei A. Efros, and Richard Zhang. Swapping Autoencoder for Deep Image Manipulation. In *NeurIPS*, 2020b.

Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. MirrorGAN: Learning Text-to-image Generation by Redescription. In *CVPR*, 2019.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative Adversarial Text to Image Synthesis. In *ICML*, 2016.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP*, 2019.

Eric Risser, Pierre Wilmot, and Connelly Barnes. Stable and Controllable Neural Texture Synthesis and Style Transfer Using Histogram Losses. In *arXiv:1701.08893*, 2017.

Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. Artistic Style Transfer for Videos. In *GCPR*, 2016.

Pegah Salehi, Abdolah Chalechale, and Maryam Taghizadeh. Generative Adversarial Networks (GANs): An Overview of Theoretical Model, Evaluation Metrics, and Recent Developments. In *arXiv:2005.13178*, 2020.

Roberto Di Salvo. Large Scale Ground Truth Generation for Performance Evaluation of Computer Vision Methods. In *VIGTA*, 2013.

Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Björn Ommer. A Style-Aware Content Loss for Real-time HD Style Transfer. In *ECCV*, 2018.

Ahmed Selim, Mohamed Elgharib, and Linda Doyle. Painting Style Transfer for Head Portraits using Convolutional Neural. In *SIGGRAPH*, 2016.

Lei Shi, Kai Shuang, Shijie Geng, Peng Su, Zhengkai Jiang, Peng Gao, Zuohui Fu, Gerard de Melo, and Sen Su. Contrastive Visual-Linguistic Pretraining. In *arXiv:2007.13135*, 2020.

Seitaro Shinagawa, Koichiro Yoshino, Sakriani Sakti, Yu Suzuki, and Satoshi Nakamura. Interactive Image Manipulation with Natural Language Instruction Commands. In *NeurIPS WS*, 2017.

Nathan Somavarapu, Chih-Yao Ma, and Zsolt Kira. Frustratingly Simple Domain Generalization via Image Stylization. In *arXiv:2006.11207*, 2020.

Aravind Srinivas, Michael Laskin, and Pieter Abbeel. CURL: Contrastive Unsupervised Representations for Reinforcement Learning. In *ICML*, 2020.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *CVPR*, 2016.

Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor Lempitsky. Texture Networks: Feed-forward Synthesis of Textures and Stylized Images. In *ICML*, 2016.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved Texture Networks: Maximizing Quality and Diversity in Feed-forward Stylization and Texture Synthesis. In *CVPR*, 2017.

Wenjing Wang, Shuai Yang, Jizheng Xu, and Jiaying Liu. Consistent Video Style Transfer via Relaxation and Regularization. In *TIP*, 2019a.

Xiaolong Wang, Allan Jabri, and Alexei A. Efros. Learning Correspondence from the Cycle-Consistency of Time. In *ICCV*, 2019b.

Xin Wang, Geoffrey Oxholm, Da Zhang, and Yuan-Fang Wang. Multimodal Transfer: A Hierarchical Deep Convolutional Neural Network for Fast Artistic Style Transfer. In *CVPR*, 2017.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncel. Image Quality Assessment: From Error Visibility to Structural Similarity. In *TIP*, 2004.

Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. GODIVA: Generating Open-DomaIn Videos from nAtural Descriptions. In *arXiv:2104.14806*, 2021.

Chenyun Wu, Mikayla Timm, and Subhransu Maji. Describing Textures using Natural Language. In *ECCV*, 2020.

Lin Wu, Yang Wang, and Ling Shao. Cycle-Consistent Deep Generative Hashing for Cross-Modal Retrieval. In *TIP*, 2018a.

Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination. In *CVPR*, 2018b.

Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. TediGAN: Text-Guided Diverse Face Image Generation and Manipulation. In *CVPR*, 2021.

Yingce Xia, Di He, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. Dual Learning for Machine Translation. In *NeurIPS*, 2016.

Wenqi Xian, Patsorn Sangkloy, Varun Agrawal, Amit Raj, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. TextureGAN: Controlling Deep Image Synthesis with Texture Patches. In *CVPR*, 2018.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong Hes. AttnGAN: FineGrained Text to Image Generation with Attentional Generative Adversarial Networks. In *CVPR*, 2018.

Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. DualGAN: Unsupervised Dual Learning for Image-to-Image Translation. In *ICCV*, 2017.

Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. Photorealistic Style Transfer via Wavelet Transforms. In *ICCV*, 2019.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. In *ICCV*, 2017a.

Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-Attention Generative Adversarial Networks. In *PMLR*, 2019a.

Hang Zhang and Kristin Dana. Multi-style Generative Network for Real-time Transfer. In *arXiv:1703.06953*, 2017.

Lvmin Zhang, Yi Ji, and Xin Lin. Style Transfer for Anime Sketches with Enhanced Residual U-net and Auxiliary Classifier GAN. In *ACPR*, 2017b.

Zhifei Zhang, Zhaowen Wang, Zhe Lin, and Hairong Qi. Image Super-Resolution by Neural Texture Transfer. In *CVPR*, 2019b.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *ICCV*, 2017.