Style-Content Disentanglement in Language-Image Pretraining Representations for Zero-Shot Sketch-to-Image Synthesis

Jan Zuiderveld
jan.zuiderveld@student.uva.nl
University of Amsterdam
Royal Conservatoire The Hague

Figure 1: Several results of our proposed framework, with given input sketches and style representations.

ABSTRACT

In this work, we propose and validate a framework to leverage language-image pretraining representations for training-free zero-shot sketch-to-image synthesis.

We show that disentangled content and style representations can be utilized to guide image generators to employ them as sketch-to-image generators without (re-)training any parameters. Our approach for disentangling style and content entails a simple method consisting of basic arithmetic assuming compositionality of information in representations of input sketches. Our results demonstrate that this approach is competitive with several state-of-the-art object-level domain-specific sketch-to-image models, while only depending on pretrained off-the-shelf models.

We limit our reported experiments to sketch-to-image synthesis, but preliminary results show that our method is applicable to more image-to-image translation domains.

ACM SIGGRAPH '22, August 08–11, 2022, Vancouver, Canada

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

CCS CONCEPTS

• **Computing methodologies** → **Image representations**; *Image-based rendering*.

KEYWORDS

style-content disentanglement, sketch-to-image synthesis, language-image pretraining

ACM Reference Format:
Jan Zuiderveld. 2018. Style-Content Disentanglement in Language-Image Pretraining Representations for Zero-Shot Sketch-to-Image Synthesis. In *Proceedings of Association for Computing Machinery's Special Interest Group on Computer Graphics and Interactive Techniques (ACM SIGGRAPH '22).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The power and promise of deep generative models such as GANs lie in their ability to synthesize endless realistic, diverse, and novel content with minimal user effort. The potential utility of these models continues to grow thanks to the increased quality and resolution of large-scale generative models in recent years. Conditional image synthesis allows users to use inputs to control the output of image synthesis methods. Over the years, a variety of input modalities have been studied, mostly based on conditional GANs. Several methods exist that allow generative models to be guided by conditioning variables, e.g. text, sketches or segmentation maps. Sketch conditioning finds a practical middle ground within the characteristics of these methods. It allows significantly more control over output structure than text conditioning, and does not require labels as segmentation conditioning does.

However, current sketch conditioned models are trained on a relatively narrow set of visual concepts. This restricts the space of possible inputs that result in desired output images, impeding the general usability of these models. An important cause of this is the fact that paired image datasets are labor intensive and costly to create. Approaches that circumvent this requirement exist, these include CycleGAN based architectures (1909.08313, 2104.05703.pdf) and the usage of image-to-sketch models for synthesizing paired data (2012.09290.pdf and gaugan2). But as of yet, no work has been published applying sketch-to-image on large, highly diverse datasets.

Recently, the the idea of learning visual concepts from supervision contained in natural language, Language-Image Pre-training, has gained a lot of attention. These models learn unified visual- and natural language representations from the supervision contained in the vast amount of text and associated images on the internet, resulting in very rich visual representations.

In this paper, we propose a framework to leverage classifier representations for sketch-to-image generation. A simple method consisting of basic arithmetic assuming compositionality of information in the representations is used to disentangle spatial and conceptual information from low-level, stylistic information in representations of input sketches. These disentangled representations can then be utilized to guide image generators to employ them as sketch-to-image generators without (re-)training any parameters.

The results in this paper and the supplementary material demonstrate that our approach, when used with sufficiently flexible image generators and informative representations, several state-of-the-art object-level domain-specific sketch-to-image models, while only depending on pretrained off-the-shelf models and being evaluated in a zero-shot fashion. Figure **??** shows several examples using our approach. The effectiveness of our method indicates that CLIP representations exhibit strong compositionality, something that has not been harnessed for disentanglement in earlier research.

We limit our reported experiments to sketch-to-image synthesis, but preliminary results show that our method is applicable to more image-to-image translation domains with diverse content. See Appendix **??** for a few experiments including photo-to-sketch and ...

2 RELATED WORK

2.1 Sketch to Image Synthesis

The goal of sketch-based image synthesis is to output a target image from a given sketch. Early works [8, 18, 9] regard freehand sketches as queries or constraints to retrieve each composition and stitch them into a picture, requiring availability of close image matches in the queried database for reasonable results. In recent years, an increasing number of works adopt GAN-based models [21] to learn sketch-to-image synthesis directly.

[80, 39, 7] train a Conditional GAN (Mirza and Osindero 2014) with photos and corresponding edge maps to make up for the lack of real sketch data, but only report results trained on single class datasets. However, sketches differ from edge maps in several ways, resulting in inferior results when applying these models to sketches. The authors of SketchyGAN [10] gathered the largest currently available paired dataset of sketches and pictures (75k sketches across 125 categories), and trained a conditional GAN on these. ContextualGAN [48] turns the image generation problem into an image completion problem: the network learns the joint distribution of sketch and image pairs and acquires the result by iteratively traversing the manifold. PoE-GAN is a multimodal conditional GAN that is trained on images with paired text, sketch and segmentation maps, achieving state-of-the-art sketch-to-image synthesis.

All of these proposed architectures are trained on datasets with a limited amount of classes. To the best of our knowledge, there is no published research reporting training any kind of image-to-image translation architectures on large-scale, diverse datasets for category-free, zero-shot image-to-image generation. Our work is the first to try zero-shot image-to-image generation .

2.2 Language-Image Pre-training

Multiple recent works learn cross-modal vision and language representations [12, 47, 52, 35, 30, 51, 29, 7, 32] for a variety of tasks, such as language-based image retrieval, image captioning, and visual question answering. Following the success of BERT [13] in various language tasks, recent vision and language methods typically use Transformers [55] to learn the joint representations. A recent model, based on Contrastive Language-Image Pre-training (CLIP) [42], consisting of a transformer- language encoder f_{text} and image encoder f_{image} is trained to learn a multi-modal representation space. which can be used to estimate the semantic similarity between a given text $\mathcal{T}$ and an image $\mathcal{I}$ by evaluating their cosine simalarity

CLIP was trained on 400 million text-image pairs, collected from a variety of publicly available sources on the Internet. The representations learned by CLIP have been shown to be extremely powerful, enabling state-of-the-art zero-shot image classification on a variety of datasets.

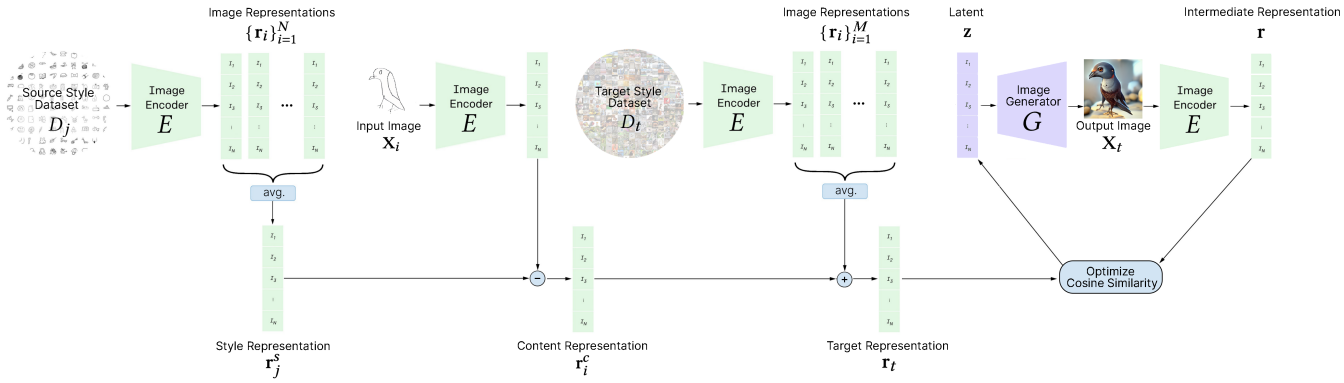

Figure 2: Overview of our proposed framework.

2.3 Classifier Guidance Based Image Synthesis

GANs for conditional image synthesis [39, 5] make heavy use of class labels. This often takes the form of class-conditional normalization statistics [16, 11] as well as discriminator heads that are explicitly designed to behave like classifiers [40], indicating that class information is crucial to the success of these models. The general idea of optimizing latent representations or parameters of an image generator using a separately trained classifier has been widely used as a powerful framework for generating, editing and recovering images; see, e.g., [1, 14, 16, 39, 2105.05233.pdf], to name only a few. Santurkar et al. X show that adversarial robustness of the classifier is a crucial aspect for optimal execution of such tasks.

Several concurrent projects use CLIP as a classifier to guide text-to-image generation through optimization. Deep Daze [38] optimizes the weights of an implicit neural representation network, while [37, 41, 10, 16] optimize the latent space of BigGAN [5], StyleGAN [25] or VQGAN [15]. GLIDE et al. propose a noised CLIP guided diffusion architecture.

3 METHODOLOGY

Much current research in unsupervised image-to-image translation relies on learning content and style representations. In these approaches, the content $\mathbf{c}_i$ of input images $\mathbf{X}_i$ and the style of target images $\mathbf{s}_K$ together condition a generator G that synthesizes translated images $\mathbf{X}_t = G(\mathbf{c}_i, \mathbf{s}_t)$.

Instead of learning content and style representations from scratch, we propose a simple method for disentangling style and content in representations of a pretrained image encoder E under the assumptions of in-distribution input and compositionality of representations. We use obtained representations to guide G. We apply this framework to do sketch-to-image synthesis. See Figure **??** for a general overview.

3.1 Problem Formulation

Let us model the formation of an image $\mathbf{X}$ as a function of style $\mathbf{s} \in \mathcal{S}$ and content $\mathbf{c} \in C$, $\mathbf{X} = G(s, c)$. Let $\mathcal{P}$ denote the set of possible parts and $\mathcal{X}$ the input space. For each $\mathbf{X} \in \mathcal{X}$, we assume the existence of a function D mapping $\mathbf{X}$ to $\mathcal{P}' \subseteq \mathcal{P}$, the set of its parts. These parts can be any feature of the input, e.g. the presence and location of an object, features of this object such as its size and color, or global style of an image. Style $\mathbf{s}$ describes all parts $\mathbf{p}^s \in D(\mathbf{X})$ that are invariant within the stylistic domain of $\mathbf{X}$, while content $\mathbf{c}$ describes all residual parts $\mathbf{p}^c \in D(\mathbf{X}_i)$ that are not described by style $\mathbf{s}$.

Given a dataset D_j of N images $\mathbf{X}_i$, all sampled from stylistic domain $\mathbf{s}_j$ with normally distributed content $\mathbf{c}_i$,

$$D_j = \{X_i\}_{i=1}^N, \quad \mathbf{X}_i = G\left(\mathbf{s}_j, \mathbf{c}_i\right), \quad \mathbf{c}_i \sim \mathcal{N}\left(0, \sigma^2 \mathbf{I}\right)$$

We use image encoder $E\colon \mathcal{X} \to \mathcal{R}$ to obtain representations $\mathbf{r}_i = E(\mathbf{X}_i)$, where $\mathcal{R}$ denotes the representation space. We assume $\mathbf{r}_i$ to be compositional, it can be expressed as a weighted sum of simpler parts. Let $h : \mathcal{P} \to \mathcal{R}$ denote a function that maps parts to representations. Formally, we define a function $f(\mathbf{X}) \in \mathcal{R}$ as compositional if it can be expressed as a weighted sum of the elements of $\{h(p) \mid p \in D(\mathbf{X})\}$ (Brendel Bethge, 2019b).

We decompose $\mathbf{r}_i$ as a sum of content representation $\mathbf{r}_i^c$ and style representation $\mathbf{r}_j^s$, $\mathbf{r}_i = \mathbf{r}_i^c + \mathbf{r}_j^s$. We assume $\mathbf{X}_i$ to be in-distribution for $E(\mathbf{X})$, thus $\mathbf{r}_i^c$ follows the distribution of $\mathbf{c}_i$. We can then obtain $\mathbf{r}_j^s$ by taking the arithmetic mean over all $\mathbf{r}_i$,

$$\mathbf{r}_j^s = \frac{1}{N}\sum_{i=1}^{N} \mathbf{r}_i = \frac{1}{N}\sum_{i=1}^{N} r_i^{\mathbf{c}} + \frac{1}{N}\sum_{i=1}^{N} \mathbf{r}_j^s = 0 + \frac{N}{N}\mathbf{r}_j^s.$$

Finally, to obtain $\mathbf{r}_i^c$ we subtract $\mathbf{r}_j^s$ from $\mathbf{r}_i$,

$$\mathbf{r}_i^c = \mathbf{r}_i - \mathbf{r}_j^s = \mathbf{r}_i^c + \mathbf{r}_j^s - \mathbf{r}_j^s.$$

Using these operations we obtain $\mathbf{r}_i^c$ describing the content of input images $\mathbf{X}_i$ from stylistic domain $\mathbf{s}_j$, and $\mathbf{r}_t^s$ describing target stylistic domain $\mathbf{s}_t$ of dataset D_t. We sum these to create target representations for guiding image synthesis: $\mathbf{r}_t = \mathbf{r}_i^c + \mathbf{r}_t^s$.

Specifically, given image generator $G(\mathbf{z})$, and image encoder E, we solve the following optimization problem to synthesize $\mathbf{X}_t$, a translation of $\mathbf{X}_i$ in style $\mathbf{s}_t$:

$$\arg\min_{\mathbf{z}\in\mathcal{Z}} \frac{\langle E(G(\mathbf{z})), \mathbf{r}_t \rangle}{\|E(G(\mathbf{z}))\| \cdot \|\mathbf{r}_t\|},$$

where $\langle \cdot, \cdot \rangle$ computes the cosine similarity between its arguments.

3.2 Improving Adversarial Robustness

Most off-the-shelf classifiers are not trained for exhibiting adversarial robustness. Since we are directly optimizing images using such classifiers, this process is prone to induce non-semantic perturbations in $\mathbf{X}_t$ that increase classification scores [cite fusedream]. Inspired by self-supervised learning methods [cite], we use augmentation pipeline a when feeding input $\mathbf{X}_t$ to image encoder e:

$$a(\mathbf{X}_t) = \mathbb{E}_{\mathbf{X}'_t \sim \pi(\cdot|\mathbf{X}_t)}$$

where $\mathbf{X}'_t$ is a random perturbation of the input image $\mathbf{X}_t$ drawn from distribution $\pi(\cdot \mid \mathbf{X}_t)$ of candidate data augmentations, including random colorization, translation, and cutout. This generates new samples $\mathbf{X}'_t$ that average out adversarial gradients while preserving most of the content and style information in $\mathbf{X}_t$.

4 EXPERIMENTAL SETUP

4.1 Image Encoder E

To evaluate the adopted framework we report experiments with CLIP [cite] as the image encoder E. CLIP is well-suited for our framework as it's large-scale dataset it has been trained on ensures that a wide range of styles $\mathbf{s}$ are in-distribution. And, more importantly, the excellent zero-shot performance of CLIP indicates its representations exhibit a significant amount of compositionality, making it well-suited for our style-content disentanglement method. As demonstrated by X et al. [1912.12179.pdf], zero-shot classification performance strongly correlates with compositionality in learned representations. This is very intuitive: we expect compositionality to be an advantage in zero-shot learning: if a model has a good understanding of how parts map to representations, it can learn to combine known concepts to describe new classes.

Since CLIP also includes a natural language encoder which maps text to the same embedding space as its image encoder, we also experiment with target styles $\mathbf{s}_t$ obtained through text encodings. See Figure **??**.

4.2 Image Generator G

The space of images our approach is limited by the representation power of the GAN we use. This makes the method difficult to generate out-of-distribution images and prune to inherit the data biases e.g., center, spatial and color bias [2, 16], from the original training set of the GAN.

Since we aim to achieve zero-shot sketch-to-image synthesis, the image generator needs to be flexible. However, very flexible image generators such as differentiable image parameterizations, e.g. implicit neural representations [cite] or direct pixel optimization, facilitate adversarial attacks.

In section **??** we report results of using VQGAN [cite] in the adopted framework. this architecture exhibits spatially flexible image synthesis due to its latent embedding structure. We initialize $\mathbf{z}$ by using VQGAN's image encoder to encode input sketch $\mathbf{X}_i$. Additionaly, we report results of using GLIDE (guided by noised CLIP) [cite] in section **??**.

REFERENCES