StEP: Style-based Encoder Pre-training for Multi-modal Image Synthesis

Moustafa Meshry  Yixuan Ren  Larry S. Davis  Abhinav Shrivastava
University of Maryland, College Park

Abstract

We propose a novel approach for multi-modal Image-to-image (I2I) translation. To tackle the one-to-many relationship between input and output domains, previous works use complex training objectives to learn a latent embedding, jointly with the generator, that models the variability of the output domain. In contrast, we directly model the style variability of images, independent of the image synthesis task. Specifically, we pre-train a generic style encoder using a novel proxy task to learn an embedding of images, from arbitrary domains, into a low-dimensional style latent space. The learned latent space introduces several advantages over previous traditional approaches to multi-modal I2I translation. First, it is not dependent on the target dataset, and generalizes well across multiple domains. Second, it learns a more powerful and expressive latent space, which improves the fidelity of style capture and transfer. The proposed style pre-training also simplifies the training objective and speeds up the training significantly. Furthermore, we provide a detailed study of the contribution of different loss terms to the task of multi-modal I2I translation, and propose a simple alternative to VAEs to enable sampling from unconstrained latent spaces. Finally, we achieve state-of-the-art results on six challenging benchmarks with a simple training objective that includes only a GAN loss and a reconstruction loss.

1. Introduction

Image-to-Image (I2I) translation is the task of transforming images from one domain to another (e.g., semantic maps → scenes, sketches → photo-realistic images, etc.). Many problems in computer vision and graphics can be cast as I2I translation, such as photo-realistic image synthesis [1–3], super-resolution [4], colorization [5, 6], and inpainting [7]. Therefore, I2I translation has recently received significant attention in the literature. One main challenge in I2I translation is the multi-modal nature for many such tasks – the relation between an input domain $A$ and an output domain $B$ is often times one-to-many, where a single input image $I^A_i \in A$ can be mapped to different output images from domain $B$. For example, a sketch of a shoe or a handbag can be mapped to corresponding objects with different colors or styles, or a semantic map of a scene can be mapped to many scenes with different appearance, lighting and/or weather conditions. Since I2I translation networks typically learn one-to-one mappings due to their deterministic nature, an extra input is required to specify an output mode to which an input image will be translated. Simply injecting extra random noise as input proved to be ineffective as shown in [2, 8], where the generator network just learns to ignore the extra noise and collapses to a single or few modes (which is one form of the mode collapse problem). To overcome this problem, Zhu et al. [8] proposed BicycleGAN, which trains an encoder network $E$, jointly with the I2I translation network, to encode the distribution of different possible outputs into a latent vector $z$, and then learns a deterministic mapping $G : (A, z) \rightarrow B$. So, depending on the latent vector $z$, a single input $I^A_i \in A$ can be mapped to multiple outputs in $B$. While BicycleGAN requires paired training data, several works , like MUNIT [9] and DRIT [10], extended it to the unsupervised case, where images in domains $A$ and $B$ are not in correspondence (‘unpaired’). One main component of unpaired I2I is a cross-cycle consistency constraint, where the network generates an intermediate output by swapping the styles of a pair of images, then swaps the style between the intermediate output again to reconstruct the original images. This enforces that the latent vector $z$ preserves the encoded style information when translated from an image $i$ to another image $j$ and back to image $i$ again. This constraint can also be applied to paired training data, where it encourages style/attribute transfer between images. However, training BicycleGAN [8] or its unsupervised counterparts [9, 10] is not trivial. For example, BicycleGAN combines the objectives of both conditional Variational Auto-Encoders (cVAEs) [11] and a conditional version of Latent Regressor GANs (cLR-GANs) [12, 13] to train their network. The training objective of [9, 10] is even more involved to handle the unsupervised setup.

In this work, we propose a novel weakly-supervised pre-training strategy to learn an expressive latent space for the task of multi-modal I2I translation. While end-to-end training of the encoder network $E$ with the I2I translation network poses a convenience, we show that it can be advantageous to break down the training into proxy tasks. In specific, we
We note that our proposed style pre-training is weakly-supervised and doesn’t require any manual labeling. Instead, it relies on a pre-trained VGG network [14] to provide training supervision. Our approach is inspired by and extends the work of Meshry et al. [15] which utilizes a staged training strategy to re-render scenes under different lighting, time of day, and weather conditions. Our work is also inspired by the standard training paradigm in visual recognition of first pre-training on a proxy task, either large supervised datasets (e.g., ImageNet) [16–18] or unsupervised tasks (e.g., [19, 20]), and then finetuning (transfer learning) on the desired task. Similarly, we propose to pre-train the encoder using a proxy task that encourages capturing style into a latent space. Our goal is to highlight the benefits of encoder pre-training and demonstrate its effectiveness for multi-modal image synthesis. In particular, we make the following contributions:

• We propose to pre-train an encoder to learn a low-dimensional projection of Gram matrices (from Neural Style Transfer) and show that the pre-trained embedding is effective for multi-modal I2I translation, and that it simplifies and stabilizes the training.

• We show that the pre-trained latent embedding is not dependent on the target domain and generalizes well to other domains (transfer learning).

• We provide a study of the importance of different loss terms for multi-modal I2I translation network.

• We propose an alternative to enable sampling from a latent space instead of enforcing a prior as done in VAE training.

• We achieve state-of-the-art results on six benchmarks in terms of style capture and transfer, and diversity of results.

2. Related work
Deep generative models. There has been incredible progress in the field of image synthesis using deep neural networks. In its unconditional setting, a decoder network learns to map random values drawn from a prior distribution (typically Gaussian) to output images. Variational Auto-Encoders (VAEs) [21] assume a bijection mapping between output images and some latent distribution and learn to map the latent distribution to a unit Gaussian using the reparameterization trick. Alternatively, Generative Adversarial Networks (GANs) [22] directly map random values sampled from a unit Gaussian to images, while using a discriminator network to enforce that the distribution of generated images resembles that of real images. Recent works proposed improvements to stabilize the training [23–26] and improve the quality and diversity of the output [27, 28]. Other works combine both VAEs and GANs into a hybrid VAE-GAN model [29, 30].

Conditional image synthesis. Instead of generating images from input noise, the generator can be augmented with side information in the form of extra conditional inputs. For example, Sohn et al. [11] extended VAEs to their conditional setup (cVAEs). Also, GANs can be conditioned on different information, like class labels [31–33], language description [34, 35], or an image from another domain [1, 2]. The latter is called Image-to-Image translation.

Image-to-Image (I2I) translation. I2I translation is the task of transforming an image from one domain, such as a sketch, into another domain, such as photo-realistic images. While there are regression-based approaches to this problem [1, 36], significant successes in this field are based on GANs and the influential work of pix2pix [2]. Following the success of pix2pix [2], I2I translation has since been utilized in a large number of tasks, like inpainting [7], colorization [5, 6], super-resolution [4], rendering [15, 37, 38], and many more [39–41]. There has also been works to extend this task to the unsupervised setting [36, 42–46], to multiple domains [47, 48], and to videos [49, 50].

Multi-modal I2I translation. Image translation networks are typically deterministic function approximators that learn a one-to-one mapping between inputs and outputs. To extend I2I translation to the case of diverse multi-modal outputs, Zhu et al. [8] proposed the BicycleGAN framework that learns a latent distribution that encodes the variability of the output domain and conditions the generator on this extra latent vector for multi-modal image synthesis. Wang et al. [3, 50] learn instance-wise latent features for different objects in a target image, which are clustered after training to find $f$ fixed modes for different semantic classes. At test time, they sample one of the feature clusters for each object to achieve multi-modal synthesis. Other works extended the multi-modal I2I framework to the unpaired setup, where images from the input and output domains are not in correspondence [9, 10, 51], by augmenting BicycleGAN with different forms of a cross-cycle consistency constraint between two unpaired image pairs. In our work, we focus on...
the supervised setting of multi-modal I2I translation. We propose a pre-training strategy to learn a latent distribution that encodes the variability of the output domain. The learned distribution can be easily adapted to new unseen datasets with simple finetuning, instead of training from scratch.

3. Approach

Current multi-modal image translation networks require an extra input $z$ that allows for modelling the one-to-many relation between an input domain $A$ and an output domain $B$ as a one-to-one relation from a pair of inputs $(A, z) \rightarrow B$. In previous approaches, there has been a trade-off between simplicity and effectiveness for providing the input $z$. On one hand, providing random noise as the extra input $z$ maintains a simple training objective (same as in pix2pix [2]). However, [2, 8] showed that the generator has little incentive to utilize the input vector $z$ since it only encodes random information, and therefore the generator ends up ignoring $z$ and collapsing to one or few modes. On the other hand, BicycleGAN [8] combines the objectives of both conditional Variational Auto-Encoder GANs (cVAE-GAN) and conditional Latent Regressor GANs (cLR-GAN) to learn a latent embedding $z$ simultaneously with the generator $G$. Their training enforces two cycle consistencies: $B \rightarrow z \rightarrow \hat{B}$ and $z \rightarrow \hat{B} \rightarrow \hat{z}$. This proved to be very effective, but the training objective is more involved, which makes the training slower. Also, since the latent embedding is being trained simultaneously with the the generator, hyper-parameter tuning becomes more critical and sensitive. The training objective of more recent works (e.g., [9, 10]) is even more complicated. We aim to combine the best of both worlds: an effective training of a latent embedding that models the distribution of possible outputs, while retaining a simple training objective. This would allow for faster and more efficient training, as well as less sensitivity to hyper-parameters. We observe that the variability in many target domains can be represented by the style diversity of images in the target domain $B$, where the style is defined in terms of the Gram matrices used in the Neural Style Transfer literature [52]. However, using Gram matrices directly to represent styles is not feasible due to its very high dimensionality. So, instead we learn an embedding by separately training an encoder network $E$ on an auxiliary task to optimize for $z = E(I^B)$ capturing the style of an image $I^B$. Visualizing the pre-trained latent space shows that our pre-trained encoder models different modes of the output distribution (e.g., different colors, lighting and weather conditions, ... etc.) as clusters of images with similar styles as shown in §4.7. Then, to synthesize an image $\hat{I}^B = G(I^A, z)$, the input latent can be used to clearly distinguish the style cluster to which the output belongs. This makes for an effective and more stable training of the generator $G$, since $G$ is just required to discover the correlation between output images and their corresponding style embedding $z$. Moreover, experimental evaluation shows that the proposed style-based pre-training yields better results in terms of more faithful style capture and transfer, as well as better output quality and diversity.

To incorporate this into BicycleGAN [8], we replace the simultaneous training of the encoder $E$ and the generator $G$ with a staged training (Figure 1) as follows:

- **Stage 1**: Pre-train $E$ on a proxy task that optimizes an embedding of images in the output domain $B$ into a low-dimensional style latent space, such that images with similar styles lie closely in that space (i.e., clustered).
- **Stage 2**: Train the generator network $G$ while fixing the encoder $E$, so that $G$ learns to associate the style of output images to their deterministic style embedding $z = E(I^B)$.
- **Stage 3**: Finetune both the $E$ and $G$ networks together, allowing for the style embedding to be further adapted to best suit the image synthesis task for the target domain.

Next, we explain how to pre-train the style encoder network $E$ in §3.1, and how to train the generator $G$ using the pre-learned embeddings (§3.2). Finally, we demonstrate the generalization of pre-training the style encoder $E$ in §3.3.

3.1. Weakly-supervised encoder pre-training

The goal of pre-training the encoder network $E$ is to learn a deterministic mapping from the style of a target image $I^B \in B$ to a latent style code $z_i = E(I^B)$. Ideally, images with similar styles should be close in the style embedding space, while images with different styles should be far apart. To supervise training such an embedding, we utilize the style loss [52] as a distance metric to measure the style similarity between any two given images. The style encoder network $E$ is then trained using a triplet loss [53], where the input is a triplet of images $(I_a, I_p, I_n)$, where $(I_a, I_p)$ have similar style, while $(I_a, I_n)$ have different style, as measured by the style loss metric. The training objective for $E$ is given by:

$$
\mathcal{L}^{tri}(I_a, I_p, I_n) = \max \left( \left[ \| z_a - z_p \|^2 - \| z_a - z_n \|^2 + \alpha \right], 0 \right) + \lambda \mathcal{L}^{reg}(z_a, z_p, z_n)
$$

(1)
where $\alpha$ is a separation margin, $\lambda$ is a relative weighting parameter between the main triplet objective and an optional regularization term $\mathcal{L}_{reg}(\cdot)$ which is an $L2$ regularization to encourage learning a compact latent space.

**Triplet selection.** To generate triplets for pre-training the encoder $E$, we compute the set of $k_c$ closest and $k_f$ furthest neighbors for each anchor image $I_a$ as measured by the style loss. Then, for each anchor image $I_a$, we randomly sample a positive image $I_p$ and a negative image $I_n$ from the set of closest and furthest neighbors respectively. We found that, for large datasets, it is sufficient to generate triplets for a subset of the training images. One challenge is the set of images with an outlier style. Such images will be furthest neighbors to most images, and can mislead the training by just projecting outlier images to separate clusters. To deal with this, we sample the negative style image $I_n$ from a larger set of furthest neighbors; while the positive image $I_p$ is sampled from a small set of closest neighbors so that it would have reasonable style similarity to the anchor image.

### 3.2. Generator training

After pre-training the style encoder $E$ (stage 1), we have established a mapping from images in the output domain, $I^B \in B$, to their style-embedding $z = E(I^B)$. Feeding the style embedding as input to the generator during training, the generator has good incentive to associate the style of output images to their corresponding style embedding instead of learning to hallucinate the style. It’s important to retain the deterministic correspondence between images and their style codes to facilitate the job of the generator to discover this correlation. This is why, during stage 2, we keep the weights of the style encoder, $E$, fixed. The forward pass reconstructs a training image $I^B_a$ as $\hat{I}^B_a = G(I^A_a, z_i)$, where $z_i = E(I^B_a)$.

The training objective is similar to that of pix2pix [2]:

$$
\mathcal{L}_{\text{img}}(I^B_a, \hat{I}^B_a) = \mathcal{L}_{\text{cGAN}}(I^B_a, \hat{I}^B_a) + \lambda_{\text{rec}} \mathcal{L}_{\text{rec}}(I^B_a, \hat{I}^B_a) \quad (2)
$$

where we use the Least Square GAN loss (LSGAN) [25] for the $\mathcal{L}_{\text{cGAN}}$ term, and a VGG-based perceptual loss [14] for the reconstruction term $\mathcal{L}_{\text{rec}}$. Once the generator has learned to associate the output style with the input style embedding, stage 3 finetunes both the generator, $G$, and the style encoder, $E$, together using the same objective (2).

**Style sampling.** To perform multimodal synthesis on a given input at test time, we can capture the latent vector $z$ from any existing image and transfer the style to the generated image. However, if we wish to sample styles directly from the latent distribution, one option is to enforce a prior on the latent distribution. For example, we found it effective to add an $L2$ regularization on the latent vectors to enforce zero-mean embeddings and limit the variance of the latent space. We then compute an empirical standard deviation for sampling. Another alternative to enable sampling is to train a mapper network $M$ to map the unit Gaussian to the latent distribution. This can be done as a post-processing step after the style encoder has been trained and finetuned. Specifically, we propose to train a mapper network $M$ using the nearest-neighbor based Implicit Maximum Likelihood Estimation (IMLE) training [54, 55]. The training objective...
is given by:

$$M = \arg\min_M \sum_i \|z_i - \tilde{M}(e_i)\|_2^2, \quad e_i = \arg\min_{r_j} \|z_i - \tilde{M}(r_j)\|_2^2$$

where \(\{r_j\}\) is a set of random samples from the unit Gaussian prior, and for each latent code \(z_i\), we select \(e_i\) that generates the nearest neighbor \(M(e_i)\) to \(z_i\).

### 3.3. Generalizing the pre-training stage

The use of Gram matrices for Neural Style Transfer proved to be very effective and it reliably captures the style of arbitrary input images. This implies that Gram matrices can reliably encode styles from a wide range of domains, and they are not specific to a certain domain. Therefore, we hypothesize that encoder pre-training using a style-based triplet loss would learn a generic style embedding that can generalize across multiple domains and be effective for multi-modal I2I translation. This would allow for performing the pre-training stage only once using auxiliary training data. The finetuning stage eventually tweaks the embedding to better suit the specific target domain \(B\). We validate our hypothesis experimentally in §4, and show that pre-training the style encoder on datasets other than the target domain \(B\) doesn’t degrade the performance. It can even improve the performance if the target dataset is small, in which case pre-training on an auxiliary dataset helps with the generalization of the overall model.

### 4. Experimental evaluation

#### Datasets

We evaluate our approach on five standard I2I translation benchmarks used in [2, 8]: Architectural labels → photo, aerial → map, edges → shoes/handbags and night → day. In addition, we use the Space Needle timelapse dataset [56], which consists of 2068 paired images with a 8280 × 1080 resolution, where the input domain includes images with temporally smoothed appearance, and the output domain contains real images spanning different lighting and weather conditions.

#### Baselines

While we report numbers for retrained models using the official code released with BicycleGAN (BicycleGAN v0) for completeness, we mainly compare to two stronger baselines:

- **BicycleGAN v1**: we implement BicycleGAN using the same network architecture as used in our approach to have a fair comparison (see supp. material for implementation details).

- **MUNIT-p**: We train MUNIT [9] in a paired setup by applying its cross-cycle consistency constraint as follows: the input is a pair of training examples \((I_1^A, I_1^B), (I_2^A, I_2^B)\) for which we obtain their respective style embeddings \(z_1 = E(I_1^A), z_2 = E(I_1^B)\). We then apply a 2-step cyclic reconstruction of \(I_1^B, I_2^B\), in the first step, we generate both images with a swapped style \(u = G(I_1^A, z_2), v = G(I_2^A, z_1)\). In the second step, we re-capture the latent style vectors \(z_2 = E(u), z_1 = E(v)\) and generate the original images \(I_1^B, I_2^B\) by swapping the style again: \(I_1^B = G(I_1^A, z_1), I_2^B = G(I_2^A, z_2)\). We add a cyclic reconstruction loss for \(I_1^B, I_2^B\).

#### 4.1. Image reconstruction

We report the reconstruction quality of validation set images, using both PSNR and AlexNet-based LPIPS [57] metrics, in Table 1. Note that our results with the pre-trained embeddings without finetuning (stage 2) are on-par-with the baselines. This verifies the validity of our approach and that style-based encoder pre-training successfully learns to distinguish different modes in the output domain, which proves effective for training multi-modal I2I networks. Finetuning (stage 3) further improves our results compared to the baselines. See supp. material for more quantitative comparison including Inception Scores (IS). Figure 2 shows qualitatively how our approach reconstructs the target style more faithfully. Our approach not only matches the ground truth colors better, but also texture (e.g., left column, first and third rows). While the baselines rely on VAEs to provide the style sampling property, we observe that, for a low dimensional latent space, the noise robustness added by VAEs reduces the expressiveness of the latent space, since slight changes to one style code would still be mapped to the same point in the latent space. This explains why our approach, which doesn’t use VAEs, achieves more faithful style capture and reconstruction. We verified this by studying the effect of removing the VAE component from the baselines, which improved their performance as shown in §4.5.

#### 4.2. Style transfer and sampling

Figure 3 shows style transfer to images from the validation set of different datasets. Note how the style trans-
Figure 5: Style sampling for different datasets using our approach after full training (e.g., phase 3). We sample either from $N(\mu, \sigma)$, where $\mu, \sigma$ are computed from the train set (middle), or using the mapper network $M$ (right).

Table 1: Validation set reconstruction quality, as measured by PSNR (higher is better) and LPIPS [57] (lower is better), for various datasets. We compare between retraining BicycleGAN [8] authors’ released code (Bicycle v0), the baselines (BicycleGAN v1 and MUNIT-p) described in §4, and our approach both before finetuning (ours - stage 2), and after finetuning (ours - stage 3).

<table>
<thead>
<tr>
<th></th>
<th>edges2handbags</th>
<th>edges2shoes</th>
<th>labels2facades</th>
<th>night2day</th>
<th>maps</th>
<th>space needle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicycle v0</td>
<td>17.08</td>
<td>0.255</td>
<td>20.24</td>
<td>0.177</td>
<td>12.64</td>
<td>0.431</td>
</tr>
<tr>
<td>Bicycle v1</td>
<td>18.52</td>
<td>0.198</td>
<td>21.84</td>
<td>0.124</td>
<td>13.08</td>
<td>0.378</td>
</tr>
<tr>
<td>MUNIT-p</td>
<td>19.23</td>
<td>0.192</td>
<td>22.51</td>
<td>0.132</td>
<td>13.36</td>
<td>0.375</td>
</tr>
<tr>
<td>Ours - stage 2</td>
<td>18.01</td>
<td>0.209</td>
<td>21.40</td>
<td>0.140</td>
<td>13.44</td>
<td>0.383</td>
</tr>
<tr>
<td>Ours - stage 3</td>
<td>18.91</td>
<td>0.177</td>
<td>22.68</td>
<td>0.117</td>
<td>13.44</td>
<td>0.370</td>
</tr>
<tr>
<td></td>
<td>PSNR ↑</td>
<td>LPIPS ↓</td>
<td>PSNR ↑</td>
<td>LPIPS ↓</td>
<td>PSNR ↑</td>
<td>LPIPS ↓</td>
</tr>
<tr>
<td>Bicycle v0</td>
<td></td>
<td></td>
<td>13.25</td>
<td>0.491</td>
<td>14.67</td>
<td>0.359</td>
</tr>
<tr>
<td>Bicycle v1</td>
<td></td>
<td></td>
<td>14.48</td>
<td>0.480</td>
<td>16.17</td>
<td>0.407</td>
</tr>
<tr>
<td>MUNIT-p</td>
<td></td>
<td></td>
<td>14.48</td>
<td>0.480</td>
<td>16.17</td>
<td>0.407</td>
</tr>
<tr>
<td>Ours - stage 2</td>
<td></td>
<td></td>
<td>14.48</td>
<td>0.480</td>
<td>16.17</td>
<td>0.407</td>
</tr>
<tr>
<td>Ours - stage 3</td>
<td></td>
<td></td>
<td>15.05</td>
<td>0.452</td>
<td>15.15</td>
<td>0.349</td>
</tr>
</tbody>
</table>

4.3. Style interpolation

Figure 4 shows style interpolation by linearly interpolating between two latent vectors. For example, note the smooth change in lighting and cloud patterns when going from cloudy to sunny in the Space Needle dataset. More interpolation results on other datasets, and comparison with interpolation results of the baselines can be found in the supplementary material.

4.4. Pre-training generalization

Since the notion of style, as defined in the Neural Style Transfer literature, is universal and not specific to a certain domain, we hypothesized that style-based encoder pretraining would learn a generic style embedding that can generalize across multiple domains and be effective for

fer copies the weather conditions in the Space Needle and Night2day datasets. For example, in the Space Needle dataset, we show sunset, sunny, foggy and cloudy weather. Also, the Night2day examples exhibit variation in lighting conditions including transferring whether the surface is sunlit or not, as well as different cloud patterns and clear skies. Comparison to the baselines’ style transfer results in the supplementary material further highlights the improvements of our approach. We can also sample random styles directly from the latent distribution as described in §3.2. Figure 5 shows results for both adhoc sampling from the assumed $N(\mu, \sigma)$ empirical distribution, as well as formally sampling from a unit Gaussian using the mapper network $M$. Note that the diversity of sampled styles doesn’t stem from simple color changes; for example, sampled styles for the Space Needle dataset show clear weather changes, such as cloudy/sunny, different cloud patterns and even sampling foggy weather which was present in some images in the training set. In the Maps dataset, the existence and/or density of bushes clearly varies between different sampled styles. Also, in the Edges2Handbags dataset, the texture of the bag varies between smooth and rough leather (better seen in zoom). While the results of both sampling methods look good, we note that the assumption for adhoc sampling is not explicitly enforced, and thus could lead to sampling bad style codes outside the distribution (see supp. material for examples).
we train the generator $G$

Table 2: Ablation study of the effect of different components and loss terms using the edges2handbags dataset. We study direct and cyclic reconstructions on ground truth images ($\text{dir}_\text{recon}$, $\text{cyc}_\text{recon}$), discriminator loss on direct reconstructions and on generated images with a randomly sampled style ($D_{\text{dir}}$, $D_{\text{rand}}$, $z_{\text{recon}}$), latent reconstruction ($z_{\text{recon}}$), $L2$ and $\text{KL}$ regularization on the latent vector $z$ ($z_{\text{L2}}$, $z_{\text{KL}}$), and finally the use of VAE vs. just an auto-encoder.

<table>
<thead>
<tr>
<th>Approach</th>
<th>$\text{dir}_\text{recon}$</th>
<th>$\text{cyc}_\text{recon}$</th>
<th>$D_{\text{dir}}$</th>
<th>$D_{\text{rand}}$, $z_{\text{recon}}$</th>
<th>$z_{\text{L2}}$, $z_{\text{KL}}$</th>
<th>VAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicycle v1</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>MUNIT-p</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>Bicycle v2</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>MUNIT-p v2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>Bicycle v3</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>MUNIT-p v3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ours v1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ours v2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ours v3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ours v4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 3: Generalization of a pretrained style encoder $E$. We report validation set reconstruction for the edges2handbags and night2day datasets when pretraining with different datasets. Stages 2, 3 show results before/after finetuning $E$ respectively.

<table>
<thead>
<tr>
<th>Loss setup</th>
<th>IS↑</th>
<th>PSNR↑</th>
<th>LPIPS ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>$z_{\text{L2}}, z_{\text{KL}}$</td>
<td>2.31 ± 0.05</td>
<td>18.28 ± 0.30</td>
<td>0.201 ± 0.003</td>
</tr>
<tr>
<td>$z_{\text{L2}}, z_{\text{KL}}$</td>
<td>2.45 ± 0.07</td>
<td>18.96 ± 0.30</td>
<td>0.192 ± 0.002</td>
</tr>
<tr>
<td>$z_{\text{L2}}, z_{\text{KL}}$</td>
<td>2.36 ± 0.12</td>
<td>19.02 ± 0.10</td>
<td>0.175 ± 0.001</td>
</tr>
<tr>
<td>$z_{\text{L2}}, z_{\text{KL}}$</td>
<td>2.44 ± 0.06</td>
<td>19.34 ± 0.07</td>
<td>0.176 ± 0.002</td>
</tr>
</tbody>
</table>

Table 4: Diversity score is the average LPIPS distance [57]. User preference score is the percentage a method is preferred over ‘Ours v4’, on the edges2shoes dataset.

<table>
<thead>
<tr>
<th>Approach</th>
<th>LPIPS ↑ transfer</th>
<th>LPIPS ↑ sampling</th>
<th>User preference ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicycle v1</td>
<td>0.102</td>
<td>0.119</td>
<td>30.6%</td>
</tr>
<tr>
<td>MUNIT-p</td>
<td>0.138</td>
<td>0.132</td>
<td>37.7%</td>
</tr>
<tr>
<td>Ours v1</td>
<td>0.153</td>
<td>0.148</td>
<td>46.5%</td>
</tr>
<tr>
<td>Ours v2</td>
<td>0.171</td>
<td>0.140</td>
<td>41.1%</td>
</tr>
<tr>
<td>Ours v3</td>
<td>0.149</td>
<td>0.165</td>
<td>50.4%</td>
</tr>
<tr>
<td>Ours v4</td>
<td>0.154</td>
<td>0.132</td>
<td>50%</td>
</tr>
</tbody>
</table>

multi-modal image ΩI translation. Here, we experimentally verify our hypothesis in Table 3. For a target dataset, we train the generator $G$ three times, each with different pre-training of the style encoder $E$: (1) same dataset pre-training: pre-train $E$ using the output domain $B$ of the target dataset. (2) similar-domain pre-training: pre-train on a different dataset, but whose output domain bears resemblance to the output domain of the target dataset (e.g., edges2shoes and edges2handbags, or day images from night2day and the Space Needle timelapse dataset). (3) different-domain pre-training: pre-train on a different dataset whose output domain has different styles from that of the target dataset (e.g., edges2handbags and the Space Needle timelapse datasets, or night2day and edges2handbags datasets). Table 3 shows that without finetuning (i.e., stage 2), the edges2handbags dataset shows a slight performance degradation when going from pre-training on the same dataset, to pre-training on a similar-domain dataset, and finally pre-training on a different-domain dataset. On the other hand, the night2day dataset has only ~100 unique scenes for training. So, pre-training on another dataset such as Space Needle generalizes better to new scenes in the validation set, since it helps avoid overfitting the small number of unique scenes in the training set. After finetuning, performance differences further reduce to be insignificant. We also investigate the generalization of the proposed encoder pre-training to the case of using non-style distance metrics in the supplementary material.

4.5 Ablative study

We investigate the role of different loss terms as we transition from the loss setup of the baselines to that of our training approach. We first remove the variational part in both BicycleGAN v1 and MUNIT-p baselines resulting in Bicycle v2, MUNIT-p v2. We further remove the Gaussian prior and replace the KL loss with an $L2$ regularization in Bicycle v3, MUNIT-p v3. To maintain random latent vector sampling during training without a prior, we sample a random training image, and use its style code. We define different versions of our approach (v1, v2, v3, and v4) based on different loss setup during training as follows: we start with ‘Ours v1’, which has the same setup as MUNIT-p v3, except that it uses pre-trained embeddings as described in §3.1. We then remove cyclic reconstruction, random $z$ sampling, and $L2$ regularization terms resulting in ‘Ours v2’, ‘v3’, and ‘v4’ re-
respectively. We run each setup on the edges2handbags dataset. In order to draw more reliable conclusions, we repeat each experiment 3 times and report the mean and standard deviation in Table 2. We notice that removing the variational part in VAEs is enough to improve the reconstruction results. While VAEs in general are robust to noise in the input latent, we observe that this comes at the expense of the expressiveness of the latent space (e.g., less faithful style capture and transfer), especially for low dimensional latents. We also observe that our approach generally performs better with less constraints (loss terms). For example, “Ours v1, v2” have lower results than their “Bicycle v3”, “MUNIT-p v3” counterparts. This shows that the main benefit of pre-trained embeddings is when the network is less constrained.

4.6. Diversity and user study

We evaluate diversity by computing the average LPIPS distance over 1600 outputs. We measure diversity on two setups: we sample 100 validation images, and (1) apply style transfer from 16 randomly sampled images, or (2) we sample 16 random codes using the mapper network $M$ to obtain 1600 outputs. We also measure the realism and faithfulness of style transfer through a user study, where 30 participants are shown an input shoe sketch, an input style image and two style transfer outputs. They are asked to choose which output looks more realistic, and if both are realistic, then which transfers the style more faithfully. We fix ‘Ours v4’ approach as anchor and compare other methods to it. Table 4 shows that the baselines achieve lower diversity and user preference compared to our approach, specially in the style transfer setup. Different variations of our method, except for ‘Ours v2’ yield similar diversity and user preference scores. We observe that ‘Ours v2’ shows artifacts in some outputs, leading to higher diversity but lower user preference. Our diversity results for the style sampling setup have some variation and are sensitive to the mapper network training, but are still either on-par or better than the baselines.

4.7. Visualizing pre-trained embeddings

Figure 6 visualizes the pre-trained latent space learned by the style encoder $E$. The visualization shows meaningful clusters of similar styles (weather conditions for the Space Needle timelapse dataset). Refer to supp. material for visualization of the latent space after finetuning, as well as the latent space learned by the baselines.

More analysis and discussion: Please refer to the supplementary material for convergence analysis, training time comparison, and more quantitative and qualitative results.

5. Conclusion

We investigated the effectiveness of Style-based Encoder Pre-training (StEP) for the task of multi-modal I2I translation. The proposed pre-training can be done once on auxiliary data and generalizes well to several domains. This allows for a faster training of I2I translation networks with fewer losses and achieves more faithful style capture and transfer. Furthermore, we studied the contribution of different loss terms to the task of multi-modal I2I translation, where we discovered that noise added by a variational auto-encoder can limit the expressiveness of low-dimensional latent spaces. We proposed two simple alternatives to VAEs to provide latent code sampling. Finally, we achieved state-of-the-art results on several benchmarks.

Acknowledgements. We would like to thank Ricardo Martin-Brualla for helping with initial concepts and reviewing drafts. We also thank Kamal Gupta and Alex Hanson for suggesting the acronym for our approach, and the members of the Perception and Intelligence (PI) Lab for their helpful feedback. The project was partially funded by DARPA MediFor (FA87501620191) and DARPA SAIL-ON (W911NF2020009) programs.


[41] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *CVPR*, 2017. 2


