Cross-domain Correspondence Learning for Exemplar-based Image Translation

Pan Zhang\textsuperscript{1} * Bo Zhang\textsuperscript{2}, Dong Chen\textsuperscript{2}, Lu Yuan\textsuperscript{3}, Fang Wen\textsuperscript{2}
\textsuperscript{1}University of Science and Technology of China \textsuperscript{2}Microsoft Research Asia \textsuperscript{3}Microsoft Cloud+AI

Abstract

We present a general framework for exemplar-based image translation, which synthesizes a photo-realistic image from the input in a distinct domain (e.g., semantic segmentation mask, or edge map, or pose keypoints), given an exemplar image. The output has the style (e.g., color, texture) in consistency with the semantically corresponding objects in the exemplar. We propose to jointly learn the cross-domain correspondence and the image translation, where both tasks facilitate each other and thus can be learned with weak supervision. The images from distinct domains are first aligned to an intermediate domain where dense correspondence is established. Then, the network synthesizes images based on the appearance of semantically corresponding patches in the exemplar. We demonstrate the effectiveness of our approach in several image translation tasks. Our method is superior to state-of-the-art methods in terms of image quality significantly, with the image style faithful to the exemplar with semantic consistency. Moreover, we show the utility of our method for several applications.

1. Introduction

Conditional image synthesis aims to generate photo-realistic images based on certain input data [18, 45, 53, 6]. We are interested in a specific form of conditional image synthesis, which converts a semantic segmentation mask, an edge map, and pose keypoints to a photo-realistic image, given an exemplar image, as shown in Figure 1. We refer to this form as exemplar-based image translation. It allows more flexible control for multi-modal generation according to a user-given exemplar.

Recent methods directly learn the mapping from a semantic segmentation mask to an exemplar image using neural networks [17, 38, 34, 44]. Most of these methods encode the style of the exemplar into a latent style vector, from which the network synthesizes images with the desired style similar to the exemplar. However, the style code only characterizes the global style of the exemplar, regardless of spatial relevant information. Thus, it causes some local style “wash away” in the ultimate image.

To address this issue, the cross-domain correspondence between the input and the exemplar has to be established before image translation. As an extension of Image Analogies [14], Deep Analogy [27] attempts to find a dense semantically-meaningful correspondence between the image pair. It leverages deep features of VGG pretrained on

\* Author did this work during the internship at Microsoft Research Asia.
real image classification tasks for matching. We argue such representation may fail to handle a more challenging mapping from mask (or edge, keypoints) to photo since the pre-trained network does not recognize such images. In order to consider the mask (or edge) in the training, some methods [10, 47, 5] explicitly separate the exemplar image into semantic regions and learns to synthesize different parts individually. In this way, it successfully generates high-quality results. However, these approaches are task specific, and are unsuitable for general translation.

How to find a more general solution for exemplar-based image translation is non-trivial. We aim to learn the dense semantic correspondence for cross-domain images (e.g., mask-to-image, edge-to-image, keypoints-to-image, etc.), and then use it to guide the image translation. It is weakly supervise learning, since we have neither the correspondence annotations nor the synthesis ground truth given a random exemplar.

In this paper, we propose a Cross-domain Correspondence network (CoCosNet) that learns cross-domain correspondence and image translation simultaneously. The network architecture comprises two sub-networks: 1) Cross-domain Correspondence Network transforms the inputs from distinct domains to an intermediate feature domain where reliable dense correspondence can be established; 2) Translation network, employs a set of spatially-variant de-normalization blocks [38] to progressively synthesizes the output, using the style details from a warped exemplar which is semantically aligned to the mask (or edge, keypoints map) according to the estimated correspondence. Two sub-networks facilitate each other and are learned end-to-end with novel loss functions. Our method outperforms previous methods in terms of image quality by a large margin, with instance-level appearance being faithful to the exemplar. Moreover, the cross-domain correspondence implicitly learned enables some intriguing applications, such as image editing and makeup transfer. Our contribution can be summarized as follows:

- We address the problem of learning dense cross-domain correspondence with weak supervision—joint learning with image translation.
- With the cross-domain correspondence, we present a general solution to exemplar-based image translation, that for the first time, outputs images resembling the fine structures of the exemplar at instance level.
- Our method outperforms state-of-the-art methods in terms of image quality by a large margin in various application tasks.

2. Related Work

**Image-to-image translation** The goal of image translation is to learn the mapping between different image domains. Most prominent contemporary approaches solve this problem through conditional generative adversarial network [36] that leverages either paired data [18, 45, 38] or unpaired data [53, 48, 22, 29, 42]. Since the mapping from one image domain to another is inherently multi-modal, following works promote the synthesis diversity by performing stochastic sampling from the latent space [54, 17, 24]. However, none of these methods allow delicate control of the output since the latent representation is rather complex and does not have an explicit correspondence to image style. In contrast, our method supports customization of the result according to a user-given exemplar, which allows more flexible control for multi-modal generation.

**Exemplar-based image synthesis** Very recently, a few works [39, 44, 34, 40, 2] propose to synthesize photorealistic images from semantic layout under the guidance of exemplars. Non-parametric or semi-parametric approaches [39, 2] synthesize images by compositing the image fragments retrieved from a large database. Mainstream works, however, formulate the problem as image-to-image translation. Huang et al. [17] and Ma et al. [34] propose to employ Adaptive Instance Normalization (AdaIN) [16] to transfer the style code from the exemplar to the source image. Park et al. [38] learn an encoder to map the exemplar image into a vector from which the images are further synthesized. The style consistency discriminator is proposed in [44] to examine whether the image pairs exhibit a similar style. However, this method requires to constitute style consistency image pairs from video clips, which makes it unsuitable for general image translation. Unlike all of the above methods that only transfer the global style, our method transfers the fine style from a semantically corresponding region of the exemplar. Our work is inspired by the recent exemplar-based image colorization [49, 13], but we solve a more general problem: translating images between distinct domains.

**Semantic correspondence** Early studies [33, 8, 43] on semantic correspondence focus on matching hand-crafted features. With the advent of the convolutional neural network, deep features are proven powerful to represent the high-level semantics. Long et al. [32] first propose to establish semantic correspondence by matching deep features extracted from a pretrained classification model. Following works further improve the correspondence quality by incorporating additional annotations [52, 7, 11, 12, 21, 25], adopting coarse-to-fine strategy [27] or retaining reliable sparse matchings [1]. However, all these methods can only handle the correspondence between natural images instead of cross-domain images, e.g., edge and photorealistic images. We explore this new scenario and implicitly learns the task with weak supervision.
3. Approach

We aim to learn the translation from the source domain \( A \) to the target domain \( B \) given an input image \( x_A \in A \) and an exemplar image \( y_B \in B \). The generated output is desired to conform to the content as \( x_A \) while resembling the style from semantically similar parts in \( y_B \). For this purpose, the correspondence between \( x_A \) and \( y_B \), which lie in different domains, is first established, and the exemplar image is warped accordingly so that its semantics is aligned with \( x_A \) (Section 3.1). Thereafter, an image is synthesized according to the warped exemplar (Section 3.2). The whole network architecture is illustrated in Figure 2, by the example of mask to image synthesis.

3.1. Cross-domain correspondence network

Usually the semantic correspondence is found by matching patches [27, 25] in the feature domain with a pre-trained classification model. However, pre-trained models are typically trained on a specific type of images, e.g., natural images, so the extracted features cannot generalize to depict the semantics for another domain. Hence, prior works cannot establish the correspondence between heterogeneous images, e.g., edge and photo-realistic images. To tackle this, we propose a novel cross-domain correspondence network, mapping the input domains to a shared domain \( S \) in which the representation is capable to represent the semantics for both input domains. As a result, reliable semantic correspondence can be found within domain \( S \).

**Domain alignment** As shown in Figure 2, we first adapt the input image and the exemplar to a shared domain \( S \). To be specific, \( x_A \) and \( y_B \) are fed into the feature pyramid network that extracts multi-scale deep features by leveraging both local and global image context [41, 28]. The extracted feature maps are further transformed to the representations in \( S \), denoted by \( x_S \in \mathbb{R}^{HW \times C} \) and \( y_S \in \mathbb{R}^{HW \times C} \) respectively (\( H,W \) are feature spatial size; \( C \) is the channel-wise dimension). Let \( F_{A \rightarrow S} \) and \( F_{B \rightarrow S} \) be the domain transformation from the two input domains respectively, so the adapted representation can be formulated as,

\[
\begin{align*}
x_S &= F_{A \rightarrow S}(x_A; \theta_{F,A \rightarrow S}), \\
y_S &= F_{B \rightarrow S}(y_B; \theta_{F,B \rightarrow S}).
\end{align*}
\]

where \( \theta \) denotes the learnable parameter. The representation \( x_S \) and \( y_S \) comprise discriminative features that characterize the semantics of inputs. Domain alignment is, in practice, essential for correspondence in that only when \( x_S \) and \( y_S \) reside in the same domain can they be further matched with some similarity measure.

**Correspondence within shared domain** We propose to match the features of \( x_S \) and \( y_S \) with the correspondence layer proposed in [49]. Concretely, we compute a correlation matrix \( \mathcal{M} \in \mathbb{R}^{HW \times HW} \) of which each element is a pairwise feature correlation,

\[
\mathcal{M}(u,v) = \frac{\hat{x}_S(u)^T \hat{y}_S(v)}{\|\hat{x}_S(u)\| \|\hat{y}_S(v)\|},
\]

where \( \hat{x}_S(u) \) and \( \hat{y}_S(v) \in \mathbb{R}^C \) represent the channel-wise centralized feature of \( x_S \) and \( y_S \) in position \( u \) and \( v \), i.e.,

\[
\hat{x}_S(u) = x_S(u) - \text{mean}(x_S(u)) \quad \text{and} \quad \hat{y}_S(v) = y_S(v) - \text{mean}(y_S(v))
\]

\( \mathcal{M}(u,v) \) indicates a higher semantic similarity between \( x_S(u) \) and \( y_S(v) \).

Now the challenge is how to learn the correspondence without direct supervision. Our idea is to jointly train with image translation. The translation network may find it easier to generate high-quality outputs only by referring to the
correct corresponding regions in the exemplar, which implicitly pushes the network to learn the accurate correspondence. In light of this, we warp \( y_B \) according to \( M \) and obtain the warped exemplar \( r_{y \rightarrow x} \in \mathbb{R}^{HW} \). Specifically, we obtain \( r_{y \rightarrow x} \) by selecting the most correlated pixels in \( y_B \) and calculating their weighted average,

\[
r_{y \rightarrow x}(u) = \sum_v \text{softmax}(\alpha M(u, v)) \cdot y_B(v). \tag{4}
\]

Here, \( \alpha \) is the coefficient that controls the sharpness of the softmax and we set its default value as 100. In the following, images will be synthesized conditioned on \( r_{y \rightarrow x} \) and the correspondence network, in this way, learns its assignment with indirect supervision.

### 3.2. Translation network

Under the guidance of \( r_{y \rightarrow x} \), the translation network \( G \) transforms the constant code \( z \) to the desired output \( \hat{x}_B \in B \). In order to preserve the structural information of \( r_{y \rightarrow x} \), we employ the spatially-adaptive denormalization (SPADE) block [38] to project the spatially variant exemplar style to different activation locations. As opposed to [38] which computes layer-wise statistics for batch normalization (BN), we empirically find the normalization that computes the statistics at each spatial position, the positional normalization (PN) [26], better preserves the structure information synthesized in prior layers. Hence, we propose to marry positional normalization and spatially-variant denormalization for high-fidelity texture transfer from the exemplar.

Formally, given the activation \( F^i \in \mathbb{R}^{C_i \times H_i \times W_i} \) before the \( i \)th normalization layer, we inject the exemplar style through

\[
\alpha^i_{h,w}(r_{y \rightarrow x}) \times \frac{F^i_{c,h,w} - \mu^i_{c,h,w}}{\sigma^i_{c,h,w}} + \beta^i_{c,h,w}(r_{y \rightarrow x}), \tag{5}
\]

where the statistic value \( \mu^i_{c,h,w} \) and \( \sigma^i_{c,h,w} \) are calculated exclusively across channel direction compared to BN. The denormalization parameter \( \alpha^i \) and \( \beta^i \) characterize the style of the exemplar, which is mapped from \( r_{y \rightarrow x} \) with the projection \( T \) parameterized by \( \theta_T \), i.e.,

\[
\alpha^i, \beta^i = T_i(r_{y \rightarrow x}; \theta_T). \tag{6}
\]

We use two plain convolutional layers to implement \( T \) so \( \alpha \) and \( \beta \) have the same spatial size as \( r_{y \rightarrow x} \). With the style modulation for each normalization layer, the overall image translation can be formulated as

\[
\hat{x}_B = \mathcal{G}(z, T_i(r_{y \rightarrow x}; \theta_T); \theta_G), \tag{7}
\]

where \( \theta_G \) denotes the learnable parameter.

### 3.3. Losses for exemplar-based translation

We jointly train the cross-domain correspondence along with image synthesis with following loss functions, hoping the two tasks benefit each other.

**Losses for pseudo exemplar pairs** We construct exemplar training pairs by utilizing paired data \( \{ x_A, x_B \} \) that are semantically aligned but differ in domains. Specifically, we apply random geometric distortion to \( x_B \) and get the distorted image \( x_B' = h(x_B) \), where \( h \) denotes the augmentation operation like image warping or random flip. When \( x_B' \) is regarded as the exemplar, the translation of \( x_A \) is expected to be its counterpart \( x_B \). In this way, we obtain pseudo exemplar pairs. We propose to penalize the difference between the translation output and the ground truth \( x_B \) by minimizing the feature matching loss [19, 18, 6]

\[
\mathcal{L}_{\text{feat}} = \sum_l \lambda_l \| \phi_l(G(x_A, x_B')) - \phi_l(x_B) \|_1, \tag{8}
\]

where \( \phi_l \) represents the activation of layer \( l \) in the pre-trained VGG-19 model and \( \lambda_l \) balance the terms.

**Domain alignment loss** We need to make sure the transformed embedding \( x_S \) and \( y_S \) lie in the same domain. To achieve this, we once again make use of the image pair \( \{ x_A, x_B \} \), whose feature embedding should be aligned exactly after domain transformation:

\[
\mathcal{L}_{\text{domain}}^l = \| \mathcal{F}_{A \rightarrow S}(x_A) - \mathcal{F}_{B \rightarrow S}(x_B) \|_1. \tag{9}
\]

Note that we perform channel-wise normalization as the last layer of \( \mathcal{F}_{A \rightarrow S} \) and \( \mathcal{F}_{B \rightarrow S} \) so minimizing this domain discrepancy will not lead to a trivial solution (i.e., small magnitude of activations).

**Exemplar translation losses** The learning with pair or pseudo exemplar pair is hard to generalize to general cases where the semantic layout of exemplar differs significantly from the source image. To tackle this, we propose the following losses.

First, the ultimate output should be consistent with the semantics of the input \( x_A \), or its counterpart \( x_B \). We thereby penalize the perceptual loss to minimize the semantic discrepancy:

\[
\mathcal{L}_{\text{perc}} = \| \phi_l(\hat{x}_B) - \phi_l(x_B) \|_1. \tag{10}
\]

Here we choose \( \phi_l \) to be the activation after \text{relu4.2} layer in the VGG-19 network since this layer mainly contains high-level semantics.

On the other hand, we need a loss function that encourages \( \hat{x}_B \) to adopt the appearance from the semantically corresponding patches from \( y_B \). To this end, we employ the contextual loss proposed in [35] to match the statistics be-
where \( L \) remains in domain \( x \) controls the relative importance of different layers. Still, we rely on pretrained VGG features. As opposed to \( L_{perc} \) which mainly utilizes high-level features, the contextual loss uses \( \text{relu}_2.2 \) up to \( \text{relu}_5.2 \) layers since low-level features capture richer style information (e.g., color or textures) useful for transferring the exemplar appearance.

**Correspondence regularization** Besides, the learned correspondence should be cycle consistent, i.e., the image should match itself after forward-backward warping, which is

\[
L_{reg} = \| r_{y \rightarrow x \rightarrow y} - y_B \|_1,
\]

where \( r_{y \rightarrow x \rightarrow y}(v) = \sum u \text{softmax}_u(\alpha M(u, v)) \cdot r_{y \rightarrow x}(u) \) is the forward-backward warping image. Indeed, this objective function is crucial because the rest loss functions, imposed at the end of the network, are weak supervision and cannot guarantee that the network learns a meaningful correspondence. Figure 9 shows that without \( L_{reg} \) the network fails to learn the cross-domain correspondence correctly although it is still capable to generate plausible translation result. The regularization \( L_{reg} \) enforces the warped image \( r_{y \rightarrow x} \) remain in domain \( B \) by constraining its backward warping, implicitly encouraging the correspondence to be meaningful as desired.

**Adversarial loss** We train a discriminator [9] that discriminates the translation outputs and the real samples of domain \( B \). Both the discriminator \( D \) and the translation network \( G \) are trained alternatively until synthesized images look indistinguishable to real ones. The adversarial objectives of \( D \) and \( G \) are respectively defined as:

\[
\begin{align*}
L_{adv}^D &= -E[h(D(y_B))] - E[h(-D(x_A, y_B))], \\
L_{adv}^G &= -E[D(G(x_A, y_B))],
\end{align*}
\]

where \( h(t) = \min(0, -1 + t) \) is a hinge function used to regularize the discriminator [50, 3].

**Total loss** In all, we optimize the following objective,

\[
L_\theta = \min \max_{D, G} \psi_1 L_{feat} + \psi_2 L_{perc} + \psi_3 L_{context} + \psi_4 L_{adv} + \psi_5 L_{domain} + \psi_6 L_{reg},
\]

where weights \( \psi \) are used to balance the objectives.

### 4. Experiments

**Implementation** We use Adam [23] solver with \( \beta_1 = 0, \beta_2 = 0.999 \). Following the TTUR [15], we set imbalanced learning rates, \( 1e-4 \) and \( 4e-4 \) respectively, for the generator and discriminator. Spectral normalization [37] is applied to all the layers for both networks to stabilize the adversarial training. Readers can refer to the supplementary material for detailed network architecture. We conduct experiments using 8 32GB Tesla V100 GPUs, and it takes roughly 4 days to train 100 epochs on the ADE20k dataset [51].

**Datasets** We conduct experiments on multiple datasets with different sorts of image representation. All the images are resized to \( 256 \times 256 \) during training.

- **ADE20k** [51] consists of \( \sim 20k \) training images, each image associated with a 150-class segmentation mask. This is a challenging dataset for most existing methods due to its large diversity.
- **ADE20k-outdoor** contains the outdoor images extracted from ADE20k, as the same protocol in SIMS [39].
- **CelebA-HQ** [30] contains high quality face images. We connect the face landmarks for face region, and use Canny edge detector to detect edges in the background. We perform an edge-to-face translation on this dataset.
- **Deepfashion** [31] consists of 52,712 person images in fashion clothes. We extract the pose keypoints using the OpenPose [4], and learn the translation to human body.

| Dataset       | ADE20k | ADE20k-outdoor | CelebA-HQ | DeepFashion |
|---------------|--------|----------------|-----------|-------------|
| **FID**       |        |                |           |             |
| Pix2pixHD     | 81.8   | 35.7           | 97.8      | 34.5        | 62.7        | 43.3         | 25.2         | 16.4         |
| SPADE         | 33.9   | 19.7           | 63.3      | 21.9        | 31.5        | 26.9         | 36.2         | 27.8         |
| MUNIT         | 129.3  | 97.8           | 168.2     | 126.3       | 56.8        | 40.8         | 74.0         | 46.2         |
| SIMS          | N/A    | N/A            | 67.7      | 27.2        | N/A         | N/A          | N/A          | N/A          |
| EGSC-IT       | 168.3  | 94.4           | 210.0     | 104.9       | 29.5        | 23.8         | 29.0         | 39.1         |
| **Ours**      | 26.4   | 10.5           | 42.4      | 11.5        | 14.3        | 15.2         | 14.4         | 17.2         |

| Dataset       | ADE20k | ADE20k-outdoor | CelebA-HQ | DeepFashion |
|---------------|--------|----------------|-----------|-------------|
| **SWD**       |        |                |           |             |
| Pix2pixHD     | 0.833  | 0.848          | 0.914     | 0.943       |
| SPADE         | 0.856  | 0.867          | 0.922     | 0.936       |
| MUNIT         | 0.723  | 0.704          | 0.848     | 0.910       |
| SIMS          | N/A    | 0.822          | N/A       | N/A         |
| EGSC-IT       | 0.734  | 0.723          | 0.915     | 0.942       |
| **Ours**      | 0.862  | 0.873          | 0.949     | 0.968       |

| Dataset       | ADE20k | ADE20k-outdoor | CelebA-HQ | DeepFashion |
|---------------|--------|----------------|-----------|-------------|
| **Comparison of semantic consistency** The best scores are highlighted. |
| Pix2pixHD     |        |                |           |             |
| SPADE         |        |                |           |             |
| MUNIT         |        |                |           |             |
| SIMS          |        |                |           |             |
| EGSC-IT       |        |                |           |             |
| **Ours**      |        |                |           |             |
Table 3: **Comparison of style relevance.** A higher score indicates a higher appearance similarity relative to the exemplar. The best scores are highlighted.

| Dataset     | ADE20k | CelebA-HQ | DeepFashion |
|-------------|--------|-----------|-------------|
| Color       | Texture| Color     | Texture     |
| SPADE       | 0.874  | 0.892     | 0.955       |
| MUNIT       | 0.745  | 0.782     | 0.939       |
| EGSC-IT     | 0.781  | 0.839     | 0.965       |
| Ours        | **0.962** | **0.941** | **0.977** |

Baselines We compare our method with state-of-the-art image translation methods: 1) Pix2pixHD [45], a leading supervised approach; 2) SPADE [38], a recently proposed supervised translation method which also supports the style injection from an exemplar image; 3) MUNIT [17], an unsupervised method that produces multi-modal results; 4) SIMS [39], which synthesizes images by compositing image segments from a memory bank; 5) EGSC-IT [34], an exemplar-based method that also considers the semantic consistency but can only mimic the global style. These methods except Pix2pixHD can generate exemplar-based results, and we use their released codes in this mode to train on several datasets. Since it is computationally prohibitive to prepare a database using SIMS, we directly use their reported figures. As we aim to propose a general translation framework, we do not include other task-specific methods. To provide the exemplar for our method, we first train a plain translation network to generate natural images and use them to retrieve the exemplars from the dataset.

Quantitative evaluation We evaluate different methods from three aspects.

- We use two metrics to measure image quality. First, we use the Fréchet Inception Score (FID) [15] to measure the distance between the distributions of synthesized images and real images. While FID measures the semantic realism, we also adopt sliced Wasserstein distance (SWD) [20] to measure their statistical distance of low-level patch distributions. Measured by these two metrics, Table 1 shows that our method significantly outperforms prior methods in almost all the comparisons. Our method improves the FID score by 7.5 compared to previous leading methods on the challenging ADE20k dataset.
Figure 5: **Our results of edge to face synthesis** (CelebA-HQ dataset). First row: exemplars. Second row: our results.

Figure 6: **Our results of pose to body synthesis** (DeepFashion). First row: exemplars. Second row: our results.

Figure 7: **User study results.**

- The ultimate output should not alter the input semantics. To evaluate the semantic consistency, we adopt an ImageNet pretrained VGG model [3], and use its high-level features maps, relu3_2, relu4_2 and relu5_2, to represent high-level semantics. We calculate the cosine similarity for these layers and take the average to yield the final score. Table 2 shows that our method best maintains the semantics during translation.

- Style relevance. We use low level features relu1_2 and relu2_2 respectively to measure the color and texture distance between the semantically corresponding patches in the output and the exemplar. We do not include Pix2pixHD as it does not produce an exemplar-based translation. Still, our method achieves considerably better instance-level style relevance as shown in Table 3.

**Qualitative comparison** Figure 3 provides a qualitative comparison of different methods. It shows that our Co-cosNet demonstrates the most visually appealing quality with much fewer artifacts. Meanwhile, compared to prior exemplar-based methods, our method demonstrates the best style fidelity, with the fine structures matching the semantically corresponding regions of the exemplar. This also correlates with the quantitative results, showing the obvious advantage of our approach. We show diverse results by changing the exemplar image in Figure 4-6. Please refer to the supplementary material for more results.

**Subjective evaluation** We also conduct user study to compare the subjective quality. We randomly select 10 images for each task, yielding 30 images in total for comparison. We design two tasks, and let users sort all the methods in terms of the image quality and the style relevance. Figure 7 shows the results, where our method demonstrates a clear advantage. Our method ranks the first in 84.2% cases in

|                | FID ↓ | Semantic consistency ↑ | Style (color/texture) ↑ |
|----------------|-------|------------------------|-------------------------|
| w/o $L_{feat}$ | 14.4  | 0.948                  | 0.975 / 0.955           |
| w/o $L_{domain}$ | 21.1  | 0.933                  | **0.983** / 0.957       |
| w/o $L_{perc}$ | 59.3  | 0.852                  | 0.971 / 0.852           |
| w/o $L_{context}$ | 28.4  | 0.931                  | 0.954 / 0.948           |
| w/o $L_{reg}$  | 19.3  | 0.929                  | 0.981 / 0.951           |
| Full           | **14.3** | **0.949**             | **0.977** / **0.958**   |
evaluating the image quality, with 93.8% chance to be the best in the style relevance comparison.

Cross-domain correspondence Figure 8 shows the cross-domain correspondence. For better visualization, we just annotate the sparse points. As the first approach in doing this, our CoCosNet successfully establishes meaningful semantic correspondence which is even difficult for manual labeling. The network is still capable to find the correspondence for sparse representation such as edge map, which captures little explicit semantic information.

Ablation study In order to validate the effectiveness of each component, we conduct comprehensive ablation studies. Here we want to emphasize two key elements (Figure 9). First, the domain alignment loss $L_{\text{domain}}$ with data pairs $x_A$ and $x_B$ is crucial. Without it, the correspondence will fail in unaligned domains, leading to oversmooth dense warping. We also ablate the correspondence regularization loss $L_{\text{reg}}$, which leads to incorrect dense correspondence, e.g., face to hair in Figure 9, though the network still yields plausible final output. With $L_{\text{reg}}$, the correspondence becomes meaningful, which facilitates the image synthesis as well. We also quantitatively measures the role of different losses in Table 4 where the full model demonstrates the best performance in terms of all the metrics.

5. Applications

Our method can enable a few intriguing applications. Here we give two examples.

Image editing Given a natural image, we can manipulate its content by modifying the segmentation layout and synthesizing the image by using the original image as the self-exemplar. Since this is similar to the pseudo exemplar pairs we constitute for training, our CoCosNet could perfectly handle it and produce the output with high quality. Figure 10 illustrates the image editing, where one can move, add and delete instances.

Makeup transfer Artists usually manually adds digital makeup on portraits. Because of the dense semantic correspondence we find, we can transfer the artistic stokes to other portraits. In this way, one can manually add makeup edits on one portrait, and use our network to process a large batch of portraits automatically based on the semantic correspondence, which illustrated in Figure 11.

6. Conclusion

In this paper, we present the CocosNet, which translates the image by relying on the cross-domain correspondence. Our method achieves preferable performance than leading approaches both quantitatively and qualitatively. Besides, our method learns the dense correspondence for cross-domain images, paving a way for several intriguing applications. Our method is computationally intensive and we leave high-resolution synthesis to future work.
References

[1] K. Aberman, J. Liao, M. Shi, D. Lischinski, B. Chen, and D. Cohen-Or, “Neural best-buddies: Sparse cross-domain correspondence,” ACM Transactions on Graphics (TOG), vol. 37, no. 4, p. 69, 2018.

[2] A. Bansal, Y. Sheikh, and D. Ramanan, “Shapes and context: In-the-wild image synthesis & manipulation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2317–2326.

[3] A. Brock, J. Donahue, and K. Simonyan, “Large scale GAN training for high fidelity natural image synthesis,” arXiv preprint arXiv:1809.11096, 2018.

[4] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, “Openpose: realtime multi-person 2d pose estimation using part affinity fields,” arXiv preprint arXiv:1812.08008, 2018.

[5] H. Chang, J. Lu, F. Yu, and A. Finkelstein, “Pairedcycle-gan: Asymmetric style transfer for applying and removing makeup,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 40–48.

[6] Q. Chen and V. Koltun, “Photographic image synthesis with cascaded refinement networks,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1511–1520.

[7] C. B. Choy, J. Gwak, S. Savarese, and M. Chandraker, “Universal correspondence network,” in Advances in Neural Information Processing Systems, 2016, pp. 2414–2422.

[8] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” 2005.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in neural information processing systems, 2014, pp. 2672–2680.

[10] S. Gu, J. Bao, H. Yang, D. Chen, F. Wen, and L. Yuan, “Mask-guided portrait editing with conditional gans,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3436–3445.

[11] B. Ham, M. Cho, C. Schmid, and J. Ponce, “Proposal flow: Semantic correspondences from object proposals,” IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 7, pp. 1711–1725, 2017.

[12] K. Han, R. S. Rezende, B. Ham, K.-Y. K. Wong, M. Cho, C. Schmid, and J. Ponce, “Sfnet: Learning semantic correspondence,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1831–1840.

[13] M. He, D. Chen, J. Liao, P. V. Sander, and L. Yuan, “Deep exemplar-based colorization,” ACM Transactions on Graphics (TOG), vol. 37, no. 4, p. 47, 2018.

[14] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, “Image analogies,” in Proceedings of the 28th annual conference on Computer graphics and interactive techniques. ACM, 2001, pp. 327–340.

[15] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in Advances in Neural Information Processing Systems, 2017, pp. 6626–6637.

[16] X. Huang and S. Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1501–1510.

[17] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, “Multimodal unsupervised image-to-image translation,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 172–189.

[18] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1125–1134.

[19] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in European conference on computer vision. Springer, 2016, pp. 694–711.

[20] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of GANs for improved quality, stability, and variation,” arXiv preprint arXiv:1710.10196, 2017.

[21] S. Kim, D. Min, B. Ham, S. Jeon, S. Lin, and K. Sohn, “Fcss: Fully convolutional self-similarity for dense semantic correspondence,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6560–6569.

[22] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, “Learning to discover cross-domain relations with generative adversarial networks,” in Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017, pp. 1857–1865.

[23] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[24] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, “Diverse image-to-image translation via disentangled representations,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 35–51.

[25] J. Lee, D. Kim, J. Ponce, and B. Ham, “Sfnet: Learning object-aware semantic correspondence,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2278–2287.

[26] B. Li, F. Wu, K. Q. Weinberger, and S. Belongie, “Positional Normalization,” arXiv e-prints, p. arXiv:1907.04312, Jul. 2019.
[53] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2223–2232.

[54] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, “Toward multimodal image-to-image translation,” in Advances in Neural Information Processing Systems, 2017, pp. 465–476.
Appendix A. Additional Qualitative Results

**Mask-to-image** We perform mask-to-image synthesis on three datasets — ADE20k, CelebA-HQ and Flickr dataset, and we show their results in Figure 12-14 respectively.

![Exemplars](image)

Figure 12: **Our results of mask-to-image synthesis (ADE20k dataset)**. In each group, the first row shows exemplars, and the second row shows the segmentation masks along with our results.
Figure 13: Our results of mask-to-image synthesis (CelebAHQ dataset). In each group, the first row shows exemplars, and the second row shows the segmentation masks along with our results.
Figure 14: Our results of mask-to-image synthesis (Flickr dataset). In each group, the first row shows exemplars, and the second row shows the segmentation masks along with our results.
Figure 15 shows additional results of edge-to-face synthesis on CelebA-HQ dataset.

Figure 15: Our results of edge-to-face synthesis (CelebA-HQ dataset). In each group, the first row shows exemplars, and the second row shows the edge maps along with our results.
**Pose-to-body**  Figure 16 shows more pose synthesis results on DeepFashion dataset.

![Pose-to-body](image)

Figure 16: **Our results of pose to image synthesis (DeepFashion dataset).** In each group, the first row shows exemplars, and the second row shows the pose images along with our results.
Appendix B. Additional Results of Dense Correspondence

The proposed CoCosNet is able to establish the dense correspondence between different domains. Figure 17 shows the dense warping results from domain B to domain A according to the correspondence ($r_{y \rightarrow x}$ in Equation 4).

Figure 17: **Warping according to the dense correspondence.** The warped image $r_{y \rightarrow x}$ is semantically aligned as the image in domain A.
Appendix C. Additional Ablation Studies

**Positional Normalization** In the translation sub-network, we empirically find the normalization that computes the statistics at each spatial position better preserves the structure information synthesized in prior layers. Such positional normalization significantly improves the lower bound of our approach. We show the worst case result of ADE20k dataset in Figure 18, where the normalization helps produce vibrant image even when the correspondence is hard to be established in the complex scene.

![Figure 18: Positional Normalization vs. Batch Normalization.](image)

The positional normalization significantly improves the lower bound of the translation image quality.

**Feature normalization for correspondence** Note that we normalize the features before computing the correlation matrix. Likewise, we propose to calculate the statistics along the channel dimension while keeping the spatial size as the feature maps. This helps transfer the fine structures in the exemplar. As shown in Figure 19, the channel-wise normalization better maintains the window structures in the ultimate output.

![Figure 19: Channel-wise normalization during correspondence.](image)

The channel-wise normalization helps transfer the window structures from the exemplar image.
Appendix D. Additional Application Results

**Image editing**  We show another example of the semantic image editing in Figure 20, where we manipulate the instances in the image by modifying their segmentation masks.

![Image editing](image.png)

Figure 20: **Image editing**. Giving the original input image along with segmentation mask (1st column), we manipulate the image by changing its semantic layout (2nd-5th columns).

**Makeup transfer**  Thanks to the dense semantic correspondence, we can transfer the makeup brushes to a batch of portraits. Figure 21 gives more supplementary results.

![Makeup transfer](image.png)

Figure 21: **Makeup transfer**. Given a portrait along with makeup edits (1st column), we can transfer the makeup to other portraits by matching the semantic correspondence.
Appendix E. Implementation Details

The detailed architecture of CoCosNet is shown in Table 5, with the naming convention as the CycleGAN.

**Cross-domain correspondence network** Two domain adaptors without weight sharing are used to adapt the input image and the exemplar to a shared domain $S$. The domain adaptors comprise several Conv-InstanceNorm-LeakReLU blocks and the spatial size of features in $S$ is $64 \times 64$. Once the intermediate domain $S$ is found, a shared adaptive feature block further transforms the features from two branches to the representation suitable for correspondence. The correlation layer computes pairwise affinity values between $4096 \times 1$ normalized features vectors. We downscale the exemplar image to $64 \times 64$ to fit the size of correlation matrix, and thus obtain the warped image on this scale. We use synchronous batch normalization within this sub-network.

**Translation network** The translation network generates the final output based on the style of the warped exemplar. We encode the exemplar style through two convolutional layers, which outputs $\alpha_i$ and $\beta_i$ to modulate the normalization layer in the generator network. We have seven such style encoder, each responsible for modulating an individual normalization layer. The generator consists of seven normalization layer, which progressively utilizes the style code to synthesize the final output. The generator also employs a nonlocal block so that a larger receptive field can be utilized to enhance the global structural consistency. We use positional normalization within this sub-network.

**Warm-up strategy** For the most challenging ADE20k dataset, a mask warm strategy is used. At the beginning of the training, we explicitly provide the segmentation mask for the domain adaptors, and employ cross-entropy loss to encourage that the masks are correctly aligned after dense warping. Such warm-up helps speed up the convergence of the correspondence network and improve the correspondence accuracy. After training 80 epochs, we replace the segmentation masks with Gaussian noise. We just use the segmentation mask for warm up and there is no need to provide the masks during inference.

Table 5: The architecture of CoCosNet. k3s1 indicates the convolutional layer with kernel size 3 and stride 1. The $i$th style encoder outputs features with dimensions matching the $i$th Resblock in the generator.

| Sub-network       | Module            | Layers in the module | Output shape (H×W×C) |
|-------------------|-------------------|----------------------|----------------------|
| Correspondence    | Domain adaptor×2  | Conv2d / k3s1        | 256×256×64           |
|                   |                   | Conv2d / k4s2        | 128×128×128          |
|                   |                   | Conv2d / k3s1        | 128×128×256          |
|                   |                   | Conv2d / k4s2        | 64×64×512            |
|                   |                   | Conv2d / k3s1        | 64×64×512            |
|                   | Resblock×3 / k3s1 | 64×64×256            |
| Adaptive feature  | Resblock×4        | Bilinear interpolation| $h' \times w' \times 3$ |
| block             | Conv2d / k1s1     | Conv2d / k3s1        | $h' \times w' \times 128$ |
|                   |                   | Conv2d / k3s1        | $h' \times w' \times C'$ |
| Correspondence    | Correlation&warping | 64×64×3             |
| Translation       | Style encoder×7   | Conv2d / k3s1        | 8×8×1024             |
| Network           |                   | Resblock×5           | 128×128×256          |
|                   |                   | Nonlocal             | 128×128×256          |
|                   |                   | Resblock×2           | 256×256×64           |
|                   |                   | Conv2d / k3s1        | 256×256×3            |
Appendix F. Detailed User Study Results

Figure 22 shows the detailed results of user study. In ADE20k, there are 67.3% and 91.9% users respectively that prefer the image quality and style relevance for our method. Regarding edge-to-face translation on CelebA-HQ, 91.3% users prefer our image quality while 90.6% users believes our method most resembles the exemplar. For pose synthesis on DeepFashion dataset, 90.6% and 98.8% users prefer our results according to the image quality and the style resemblance respectively.

Figure 22: Detailed user study results for ADE20k, CelebA-HQ and DeepFashion dataset.
Appendix G. Multimodal results for Flickr dataset

Similar to the practice in [38], we collect 56,568 landscape images from Flickr. The semantic segmentation masks are computed using a pre-trained UPerNet101 [46] network. By feeding different exemplar, our method supports multimodal landscape synthesis. Figure 23 shows highly realistic landscape results using the images in Flicker dataset.

Figure 23: Multimodal results of Flickr dataset. We only present the final synthesis results here.
Appendix H. Limitation

As an exemplar-based approach, our method may not produce satisfactory results due to one-to-many and many-to-one mappings as shown in Figure 24. We leave further research tackling these issues as future work.

Figure 24: **Limitation.** Our method may produce mixed color artifact due the one-to-many mapping (1st row). Besides, the multiple instances (pillows in the figure) may use the same style in the cases of many-to-one mapping (2nd row).

Another limitation is that the computation of the correlation matrix takes tremendous GPU memory, which makes our method hardly scale for high resolution images. We leave the solve of this issue in future work.