RETHINKING THE TRULY UNSUPERVISED IMAGE-TO-IMAGE TRANSLATION

Arxiv 20.06.11

박성현

DAVIAN
Data and Visual Analytics Lab

Level of supervision in Image-to-Image Translation

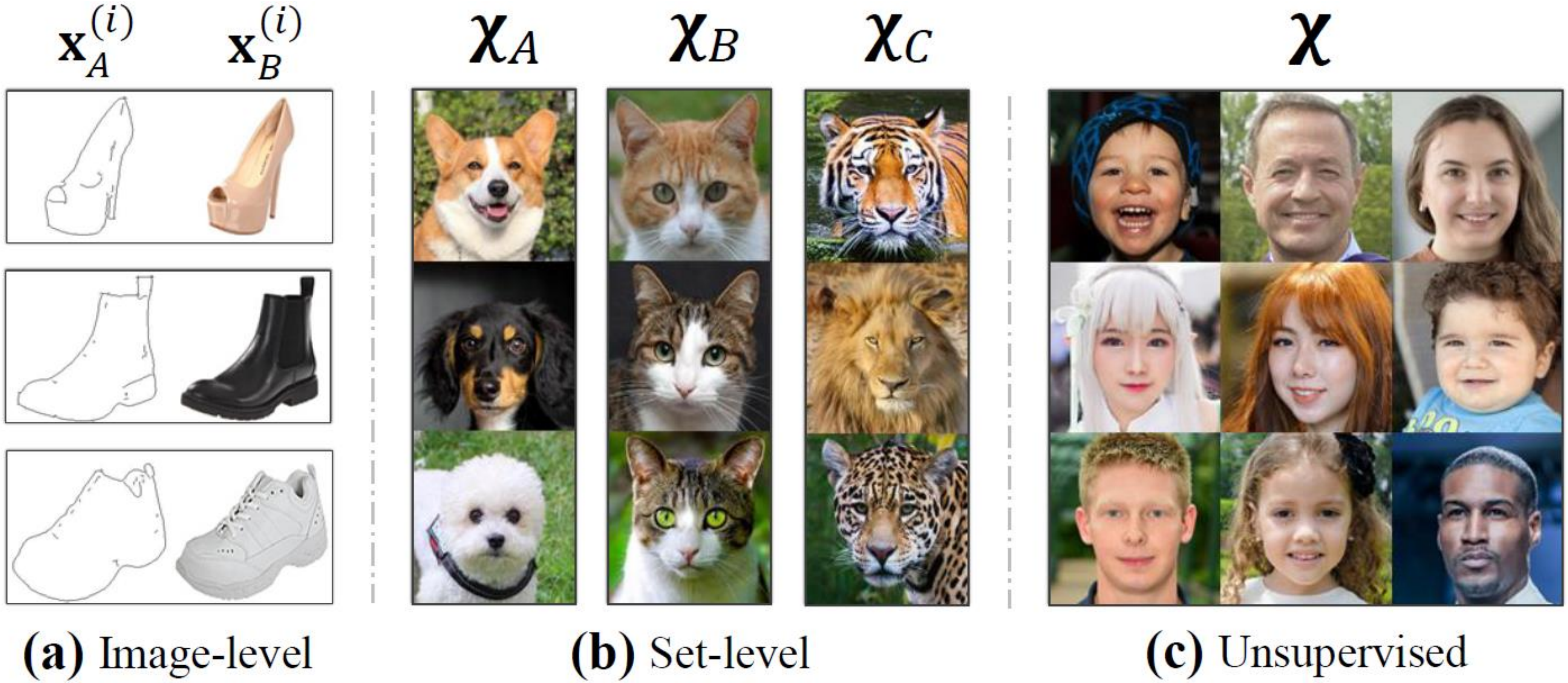


Figure 2. **Levels of supervision.** The previous methods conduct image-to-image translation relying on either **(a)** image-level or **(b)** set-level supervision. Our proposed method can perform the task using **(c)** a dataset without any supervision.

Overview of TUNIT

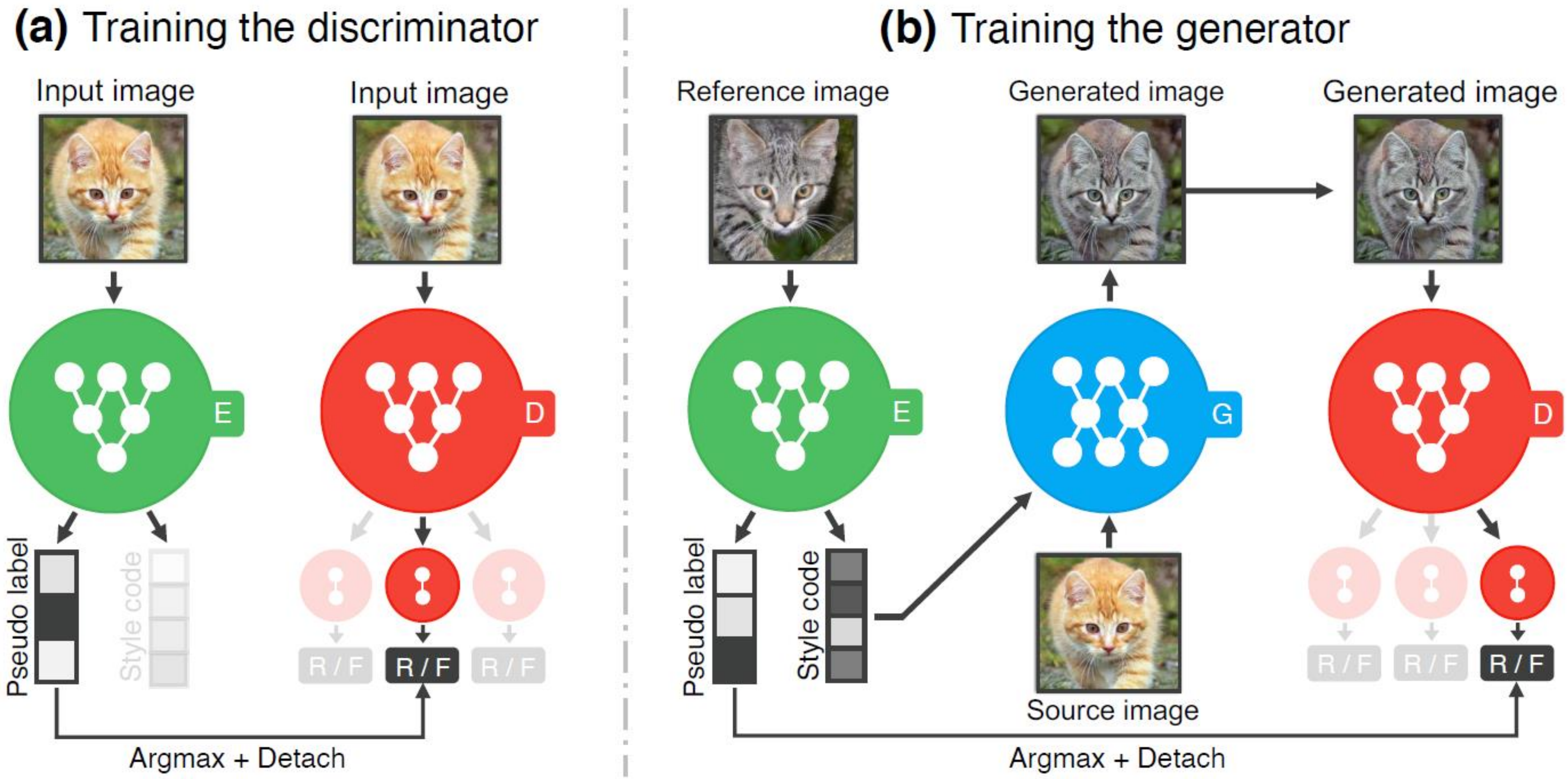


Figure 3. **Overview of our proposed method.** The figure illustrates how our model changes the breed of the cat. **(a)** An estimated domain from our guiding network E is used to train the multi-task discriminator D. **(b)** E provides the generator G with the style code of a reference image and the estimated domain is again used for GAN training.

Domain Classification: Unsupervised domain classification

- Maximize the mutual information (MI) between the domain assignments of an image x and those of its randomly augmented version x^+.

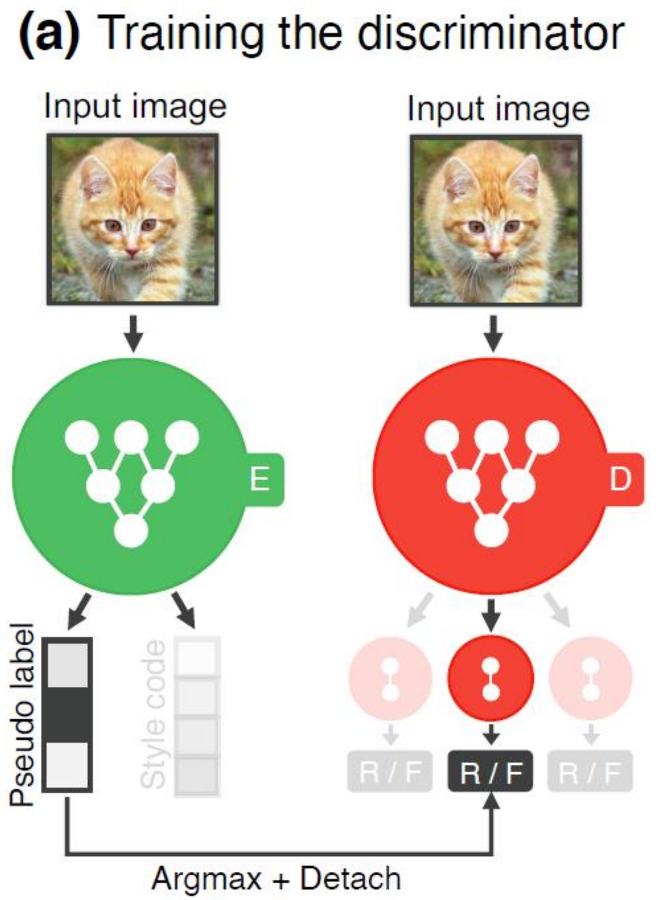
$$I(\mathbf{p}, \mathbf{p}^+) = H(\mathbf{p}) - H(\mathbf{p}|\mathbf{p}^+),$$

- By maximizing the MI, the network is encouraged to distribute all the samples as evenly as possible over K domains while confidently classifying the paired samples (x, x^+) as the same domain.
- The joint probability matrix for MI is given by $K \times K$ matrix P.

$$\mathbf{P} = \mathbb{E}_{\mathbf{x}^+ \sim f(\mathbf{x})|\mathbf{x} \sim p_{data}(\mathbf{x})}[E_{class}(\mathbf{x}) \cdot E_{class}(\mathbf{x}^+)^T],$$

- Guiding network E is trained by directly maximizing the MI.

$$\mathcal{L}_{MI} = I(\mathbf{p}, \mathbf{p}^+) = I(\mathbf{P}) = \sum_{i=1}^{K}\sum_{j=1}^{K} \mathbf{P}_{ij} \ln \frac{\mathbf{P}_{ij}}{\mathbf{P}_i \mathbf{P}_j},$$



Domain Classification: Improving domain classification

- Though maximizing L_{MI} provides a way to automatically generate the domain labels for input images, it fails to scale up when the resolution of images becomes higher that 64x64.
- Overcome this by adding an auxiliary branch E_{style} to the guiding network E and imposing the contrastive loss.

$$\mathcal{L}^{E}_{style} = -\log \frac{\exp(\mathbf{s} \cdot \mathbf{s}^{+}/\tau)}{\sum_{i=0}^{N} \exp(\mathbf{s} \cdot \mathbf{s}_i^{-}/\tau)},$$

- Utilize not only the similarity of the positive pair (s, s^{+}) but also the dissimilarity of the negative pairs (s, s_i^{-}), where the negative style codes s_i^{-} are stored into a queue using previously sampled images.

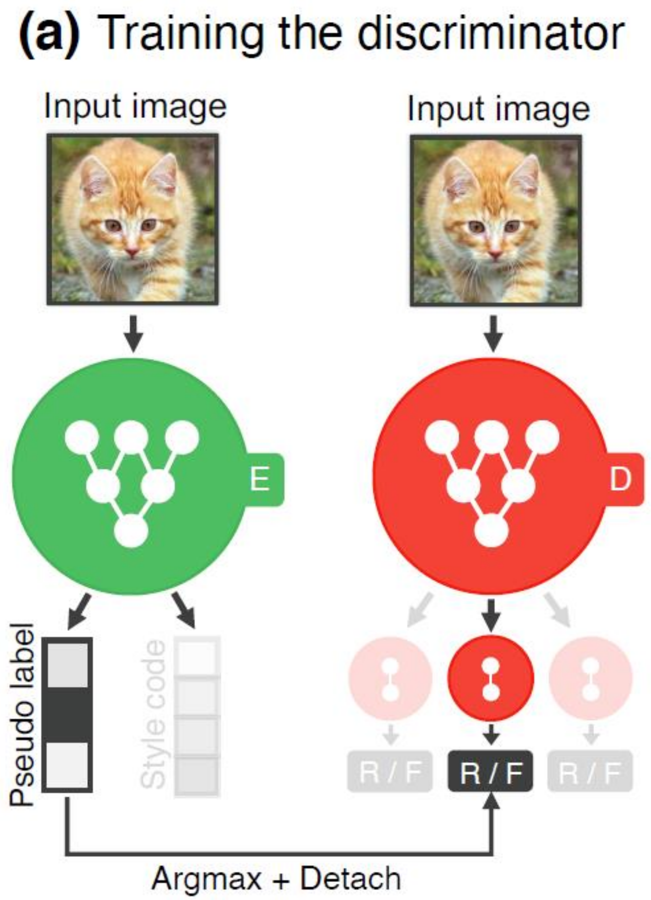


Image-to-Image Translation with the domain guidance

- Adversarial loss
  - Adopt the multi-task discriminator.
  - The discriminator outputs a binary vector whose length is the number of domains (K).
  - For the domain label of the input image, utilize the pseudo label from the guiding network.

$$\mathcal{L}_{adv} = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D_y(\mathbf{x})] + \mathbb{E}_{\mathbf{x},\tilde{\mathbf{x}}\sim p_{data}(\mathbf{x})}[\log(1 - D_{\tilde{y}}(G(\mathbf{x},\tilde{\mathbf{s}})))],$$

- Image reconstruction loss
  - To ensure that the generator G can reconstruct the source image x when given with its original style s.
  - This objective not only ensures the generator G to preserve domain-invariant characteristics(e.g., pose) of its input image x, but also helps to learn the style representation of the guiding network E by extracting the original style s of the source image x.

$$\mathcal{L}^{G}_{style} = \mathbb{E}_{\mathbf{x},\tilde{\mathbf{x}}\sim p_{data}(\mathbf{x})}\left[-\log \frac{\exp(\mathbf{s}' \cdot \tilde{\mathbf{s}})}{\sum_{i=0}^{N} \exp(\mathbf{s}' \cdot \mathbf{s}_i^- / \tau)}\right]$$

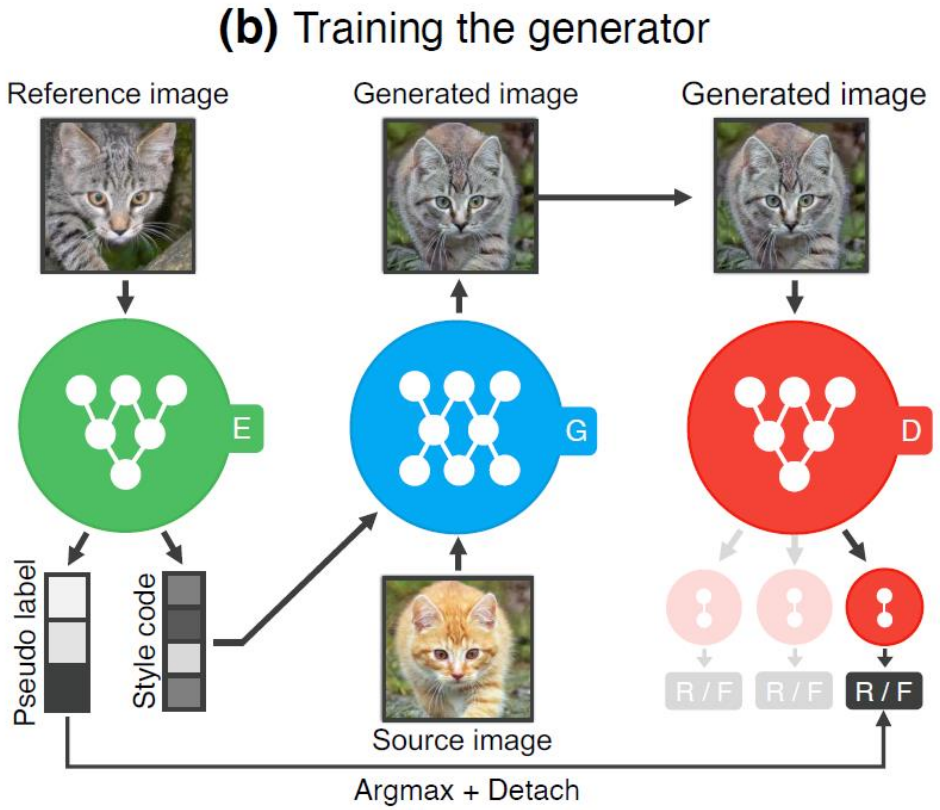


Image-to-Image Translation with the domain guidance

- Style contrastive loss
  - In order to prevent degenerate situation where the generator ignores the given style code $\tilde{s}$ and synthesize a random image of the domain $\tilde{y}$.

$$\mathcal{L}_{style}^{G} = \mathbb{E}_{\mathbf{x},\tilde{\mathbf{x}} \sim p_{data}(\mathbf{x})} \left[-\log \frac{\exp(\mathbf{s}' \cdot \tilde{\mathbf{s}})}{\sum_{i=0}^{N} \exp(\mathbf{s}' \cdot \mathbf{s}_i^- / \tau)} \right]$$

  - $s' = E_{style}(G(x, \tilde{s}))$ denotes the style code of the translated image and s_i^- denotes the negative style codes.
  - This loss guides the generated image to have a style similar to the reference image $\tilde{x}$ and dissimilar to random negative samples.
  - By doing so, avoid the degenerated solution where the encoder maps all the images to the same style code of the reconstruction loss based on L1 or L2 norm.

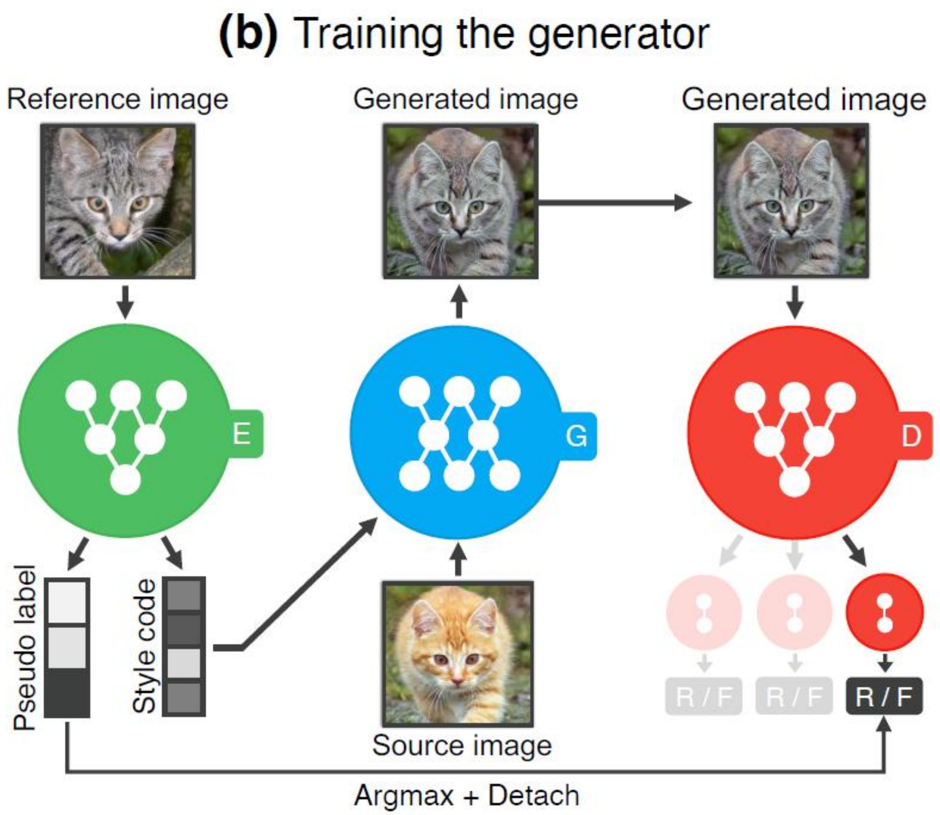
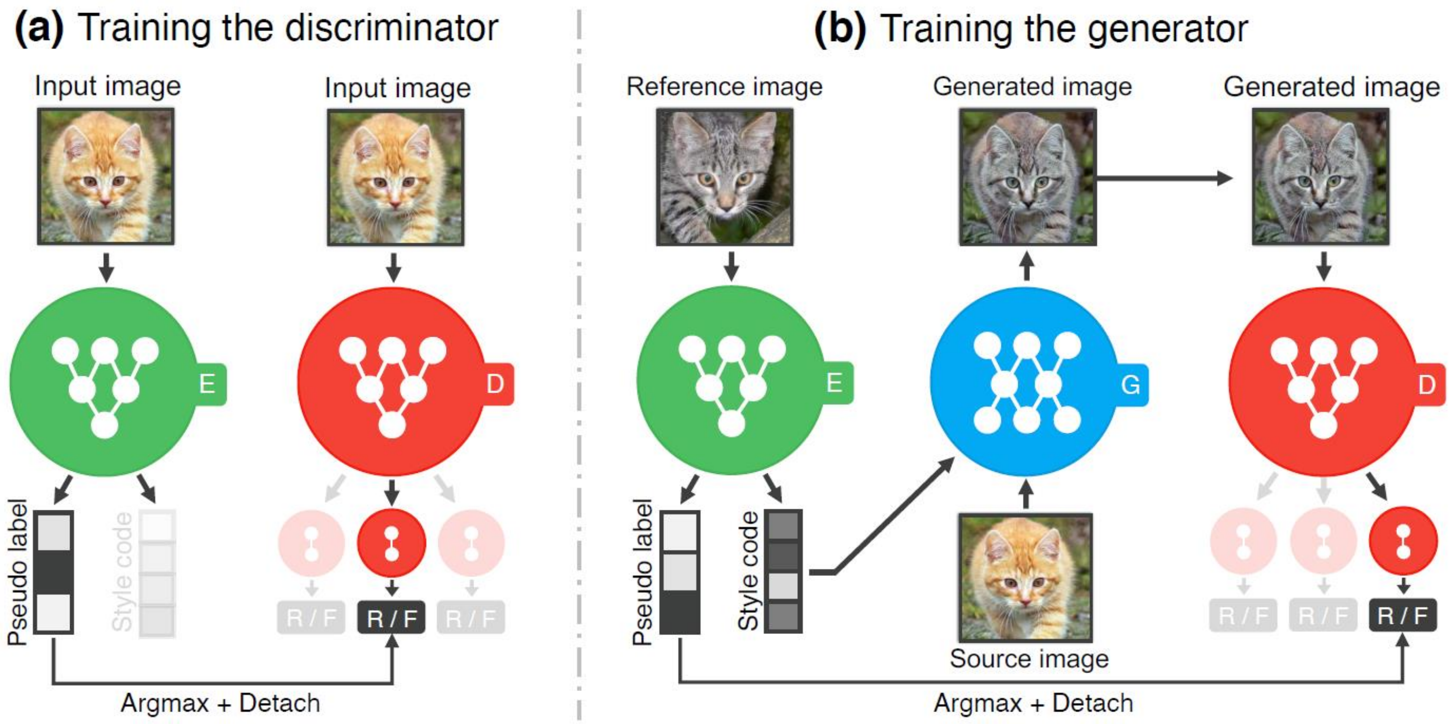


Image-to-Image Translation with the domain guidance

- Overall objective functions

$$\mathcal{L}_D = -\mathcal{L}_{adv},$$
$$\mathcal{L}_G = \mathcal{L}_{adv} + \lambda^G_{style}\mathcal{L}^G_{style} + \lambda_{rec}\mathcal{L}_{rec},$$
$$\mathcal{L}_E = \mathcal{L}_G - \lambda_{MI}\mathcal{L}_{MI} + \lambda^E_{style}\mathcal{L}^E_{style}$$



Experiments

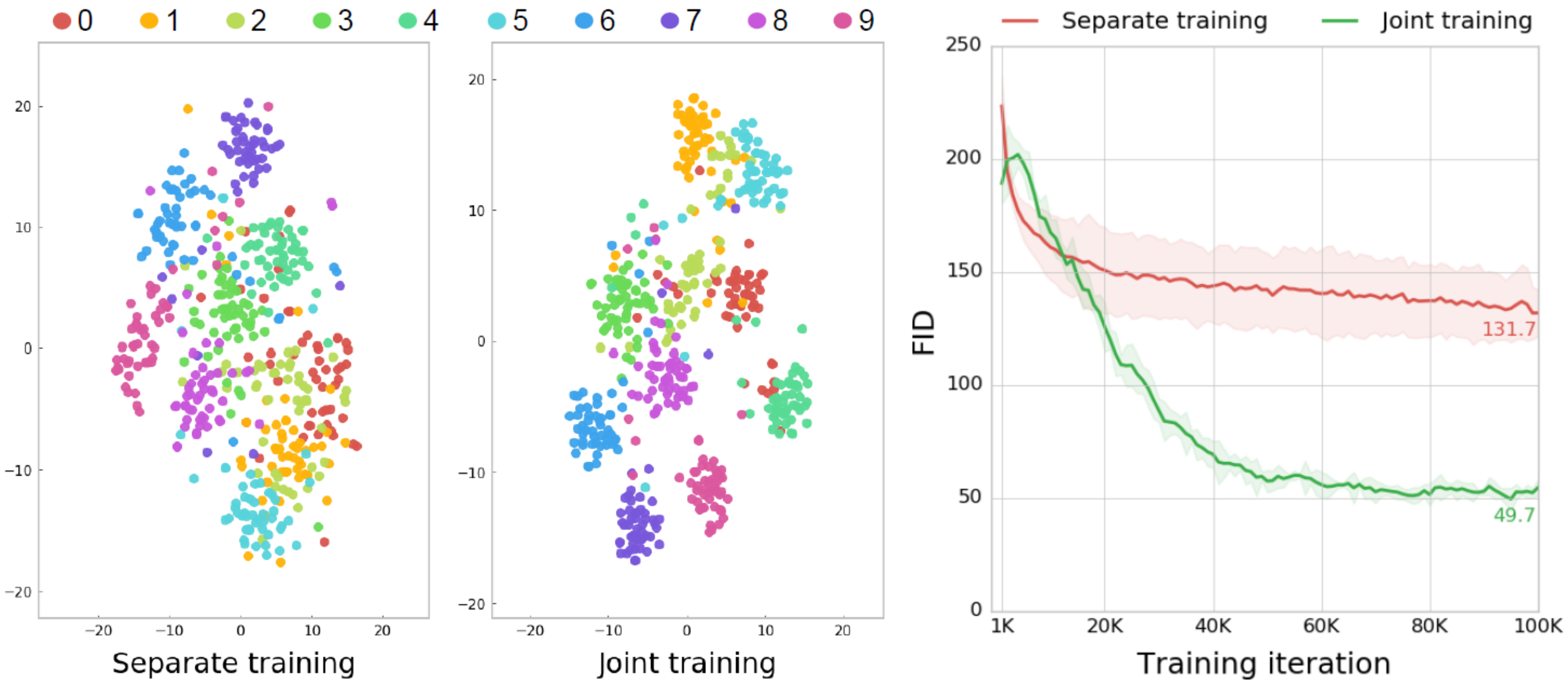


Figure 4. **Comparison of separate and joint training.** **(Left)** t-SNE visualization of style codes extracted by our guiding network. The ground truth domains of all test images in AnimalFaces-10 are represented in different colors. **(Right)** FID curves over training iterations. Joint training significantly outperforms separate training where the guiding network cannot receive feedback from the translation loss.

Experiments

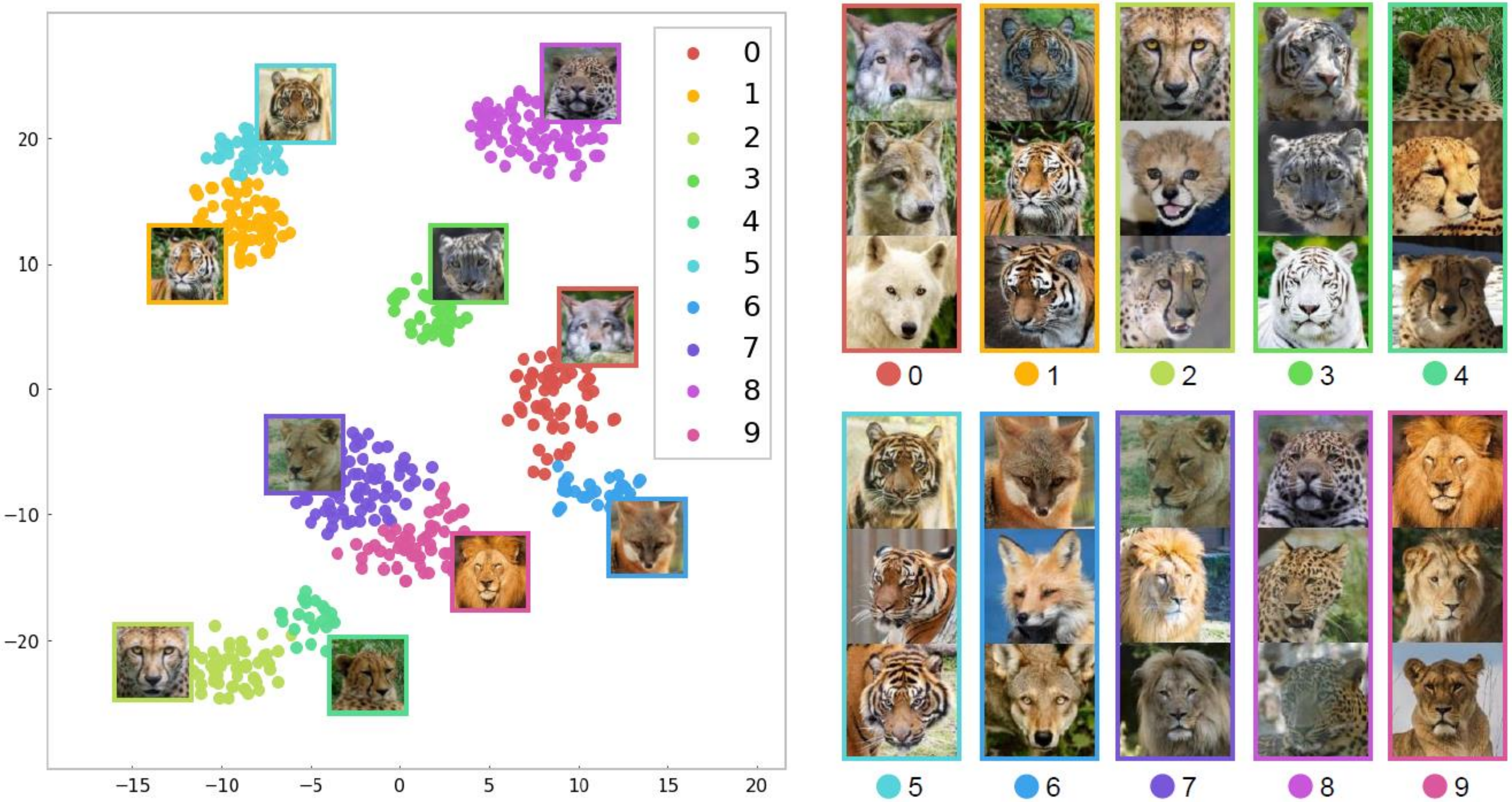


Figure 6. **t-SNE visualization of the style space** of our guiding network trained on AFHQ wild. Since AFHQ wild does not have ground-truth domain labels, each data point is colored with the guiding network's prediction. Although we set the number of domains to be quite large ($K = 10$), the network separates one species into two domains, which are so closely located that the model successfully creates six clusters.

Experiments

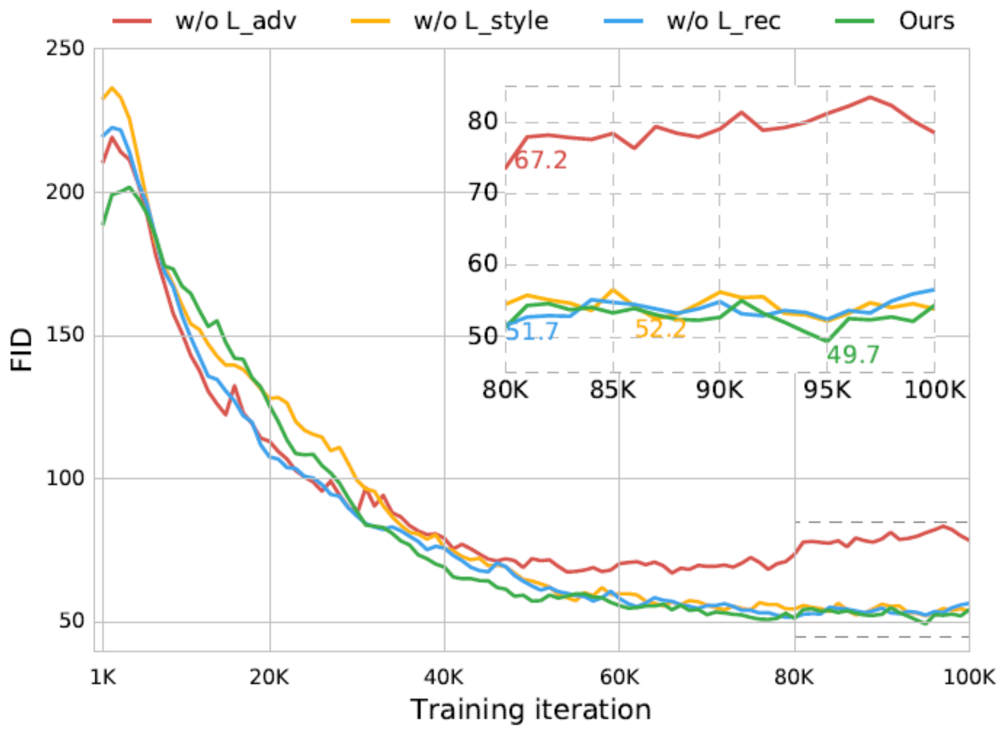


Figure 5. **Effects of translation loss on AnimalFaces-10.** During joint training, the generator is trained with entire translation loss (L_{adv}, L_{style}, and L_{rec}), but the guiding network is not received feedback from one of three losses. The FID score significantly increases when the guiding network is unable to receive feedback from the adversarial loss L_{adv}. Inset shows the zoomed-in final iterations.

Experiments

(a) AFHQ Cat.

(b) AFHQ Dog.

(c) AFHQ Wild.

Figure 7. **Unsupervised image-to-image translation results on AFHQ.** We set the number of domains (K) to ten in all cases. The top row shows representative images of ten domains estimated by our guiding network. Each source image is translated using the average style codes for each domain in test dataset. We note that all images are uncurated. The t-SNE visualization for wild can be found in Fig. 6.

(a) FFHQ

(b) LSUN Car

Figure 8. **Unsupervised image-to-image translation results on FFHQ and LSUN Car.** The experimental settings are the same as in Fig. 7.

Experiments

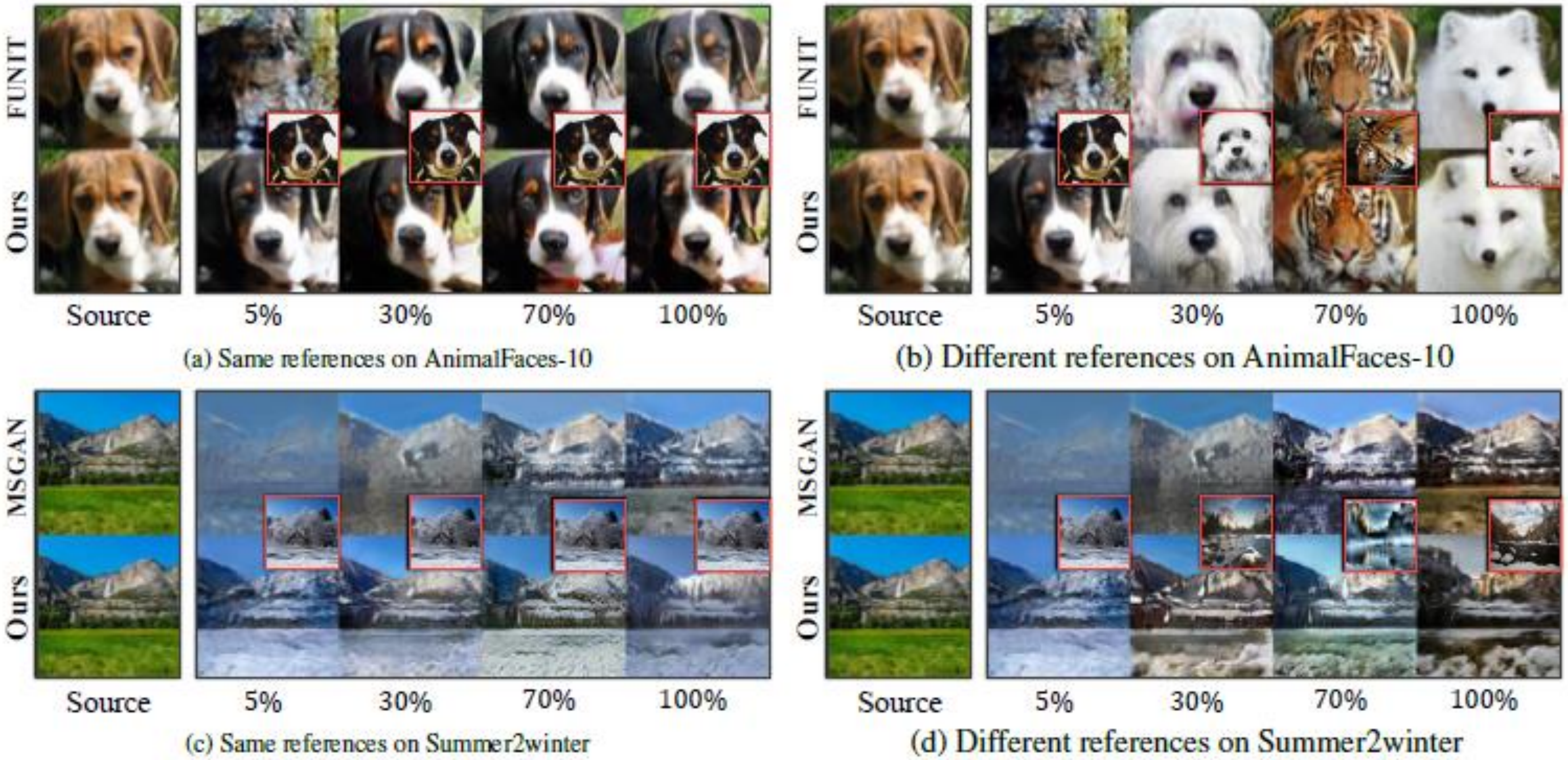
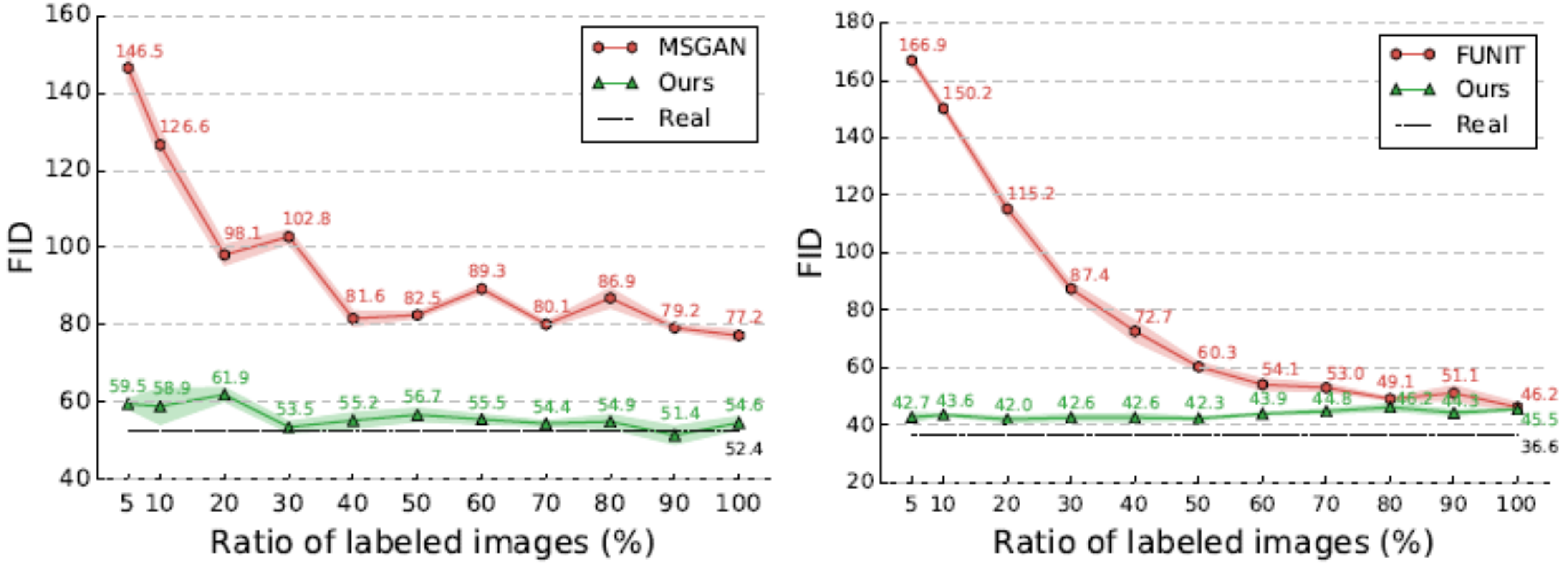


Figure 9. **Qualitative comparison for varying ratios of labeled images.** Red box indicates a reference image and the value under each image indicates the ratio of $\mathcal{D}_{sup}$.

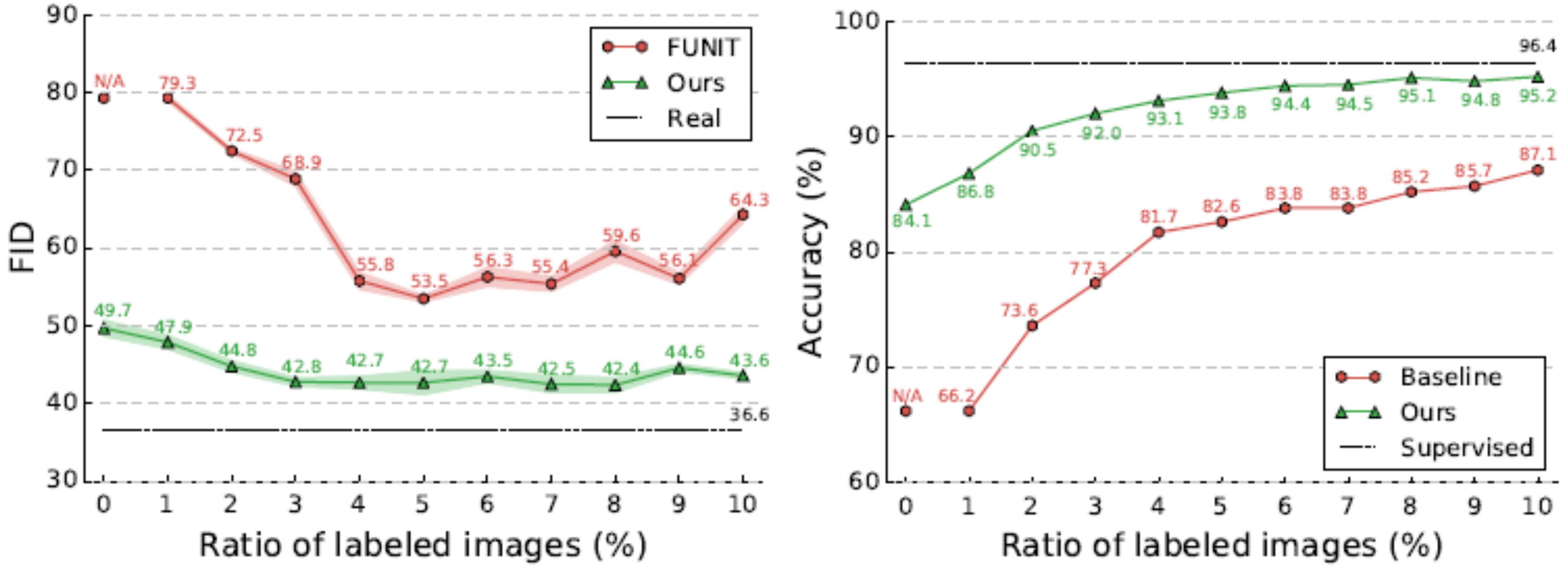
Experiments



(a) Summer2winter (b) AnimalFaces-10

Figure 10. **FID curves for varying ratios of labeled images under naïve scheme.** The dashed line indicates the expected lower bound (Real), which is calculated by dividing the training data in half. Our method is able to generate high-fidelity images using only 5% of the labeled data and outperforms the baselines in all ratios.

Experiments



(a) FID (b) Accuracy

Figure 12. **(a) FID curves on AnimalFaces-10 under alternative scheme.** Even if we introduce an auxiliary classifier to make FUNIT stronger, our method outperforms FUNIT in all ratios. **(b) Classification accuracy on AnimalFaces-10.** Our guiding network produces much more accurate domain labels compared to the baseline classfier. The dashed line indicates the accuracy when the baseline classifier utilizes the entire labels for training. We note that IIC clustering achieves 50.4% accuracy and FUNIT with IIC achieves 112.2 of FID.

Thank you!