COCO-FUNIT: Few-Shot Unsupervised Image Translation with a Content Conditional Style Encoder

2020. 12. 01

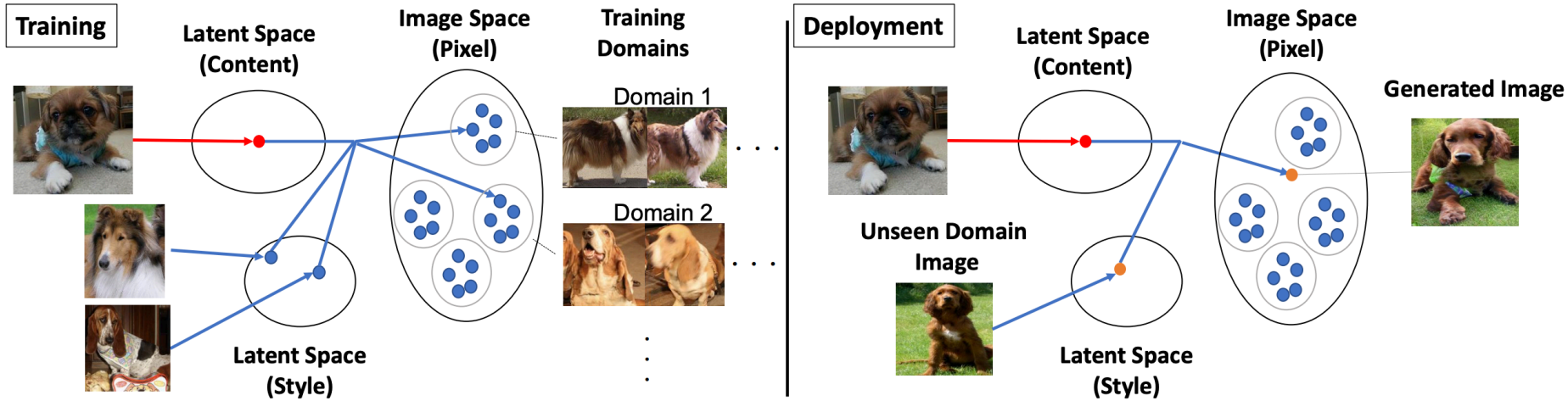
DAVIAN

presented by Junsoo Lee

Image-to-Image Translation

- Image-to-Image translation concerns learning a mapping that can translate an input image in one domain into an analogous image in a different domain.
- However, existing methods still have several drawbacks.
  - They require a large amount of images from the source and target domains for training.
  - They cannot be used to generate images in unseen domains.

Few-Shot Image-to-Image Translation



- Few-shot image translation framework learns to extract the domain-specific style information from a few example images in the unseen domain during test time, mixes it with the domain-invariant content information extracted from the input image, and generates a few-shot translation output.

Problem

- The existing few-shot image translation framework generates unsatisfactory translation outputs when the model is applied to objects with diverse appearance, such as very different poses.

Problem

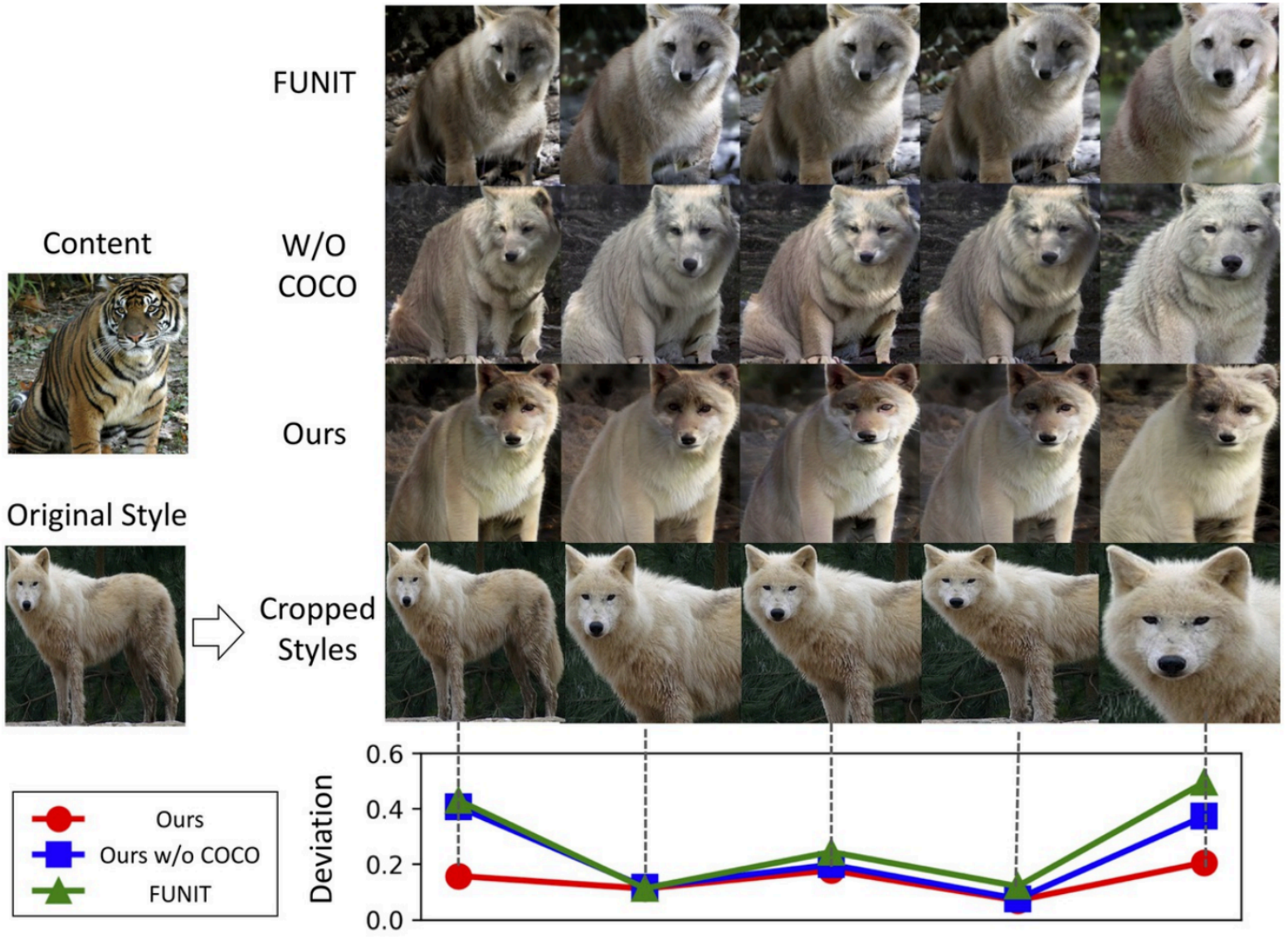
- Often, the translation output is not well-aligned with the input image.
- The domain invariant content that is supposed to remain unchanged disappears after translation.

Motivation

- Authors call this issue the content loss problem and hypothesize that solving *the content loss problem* would produce more faithful and photorealistic few-shot image translation results.

- Then, why does the *content loss problem* occur?
- Since there is no supervision, it is difficult to control what to be transferred precisely.
- Ideally, the transferred appearance should contain just the style.
- Unfortunately, it often contains other information, such as the object pose, in reality.

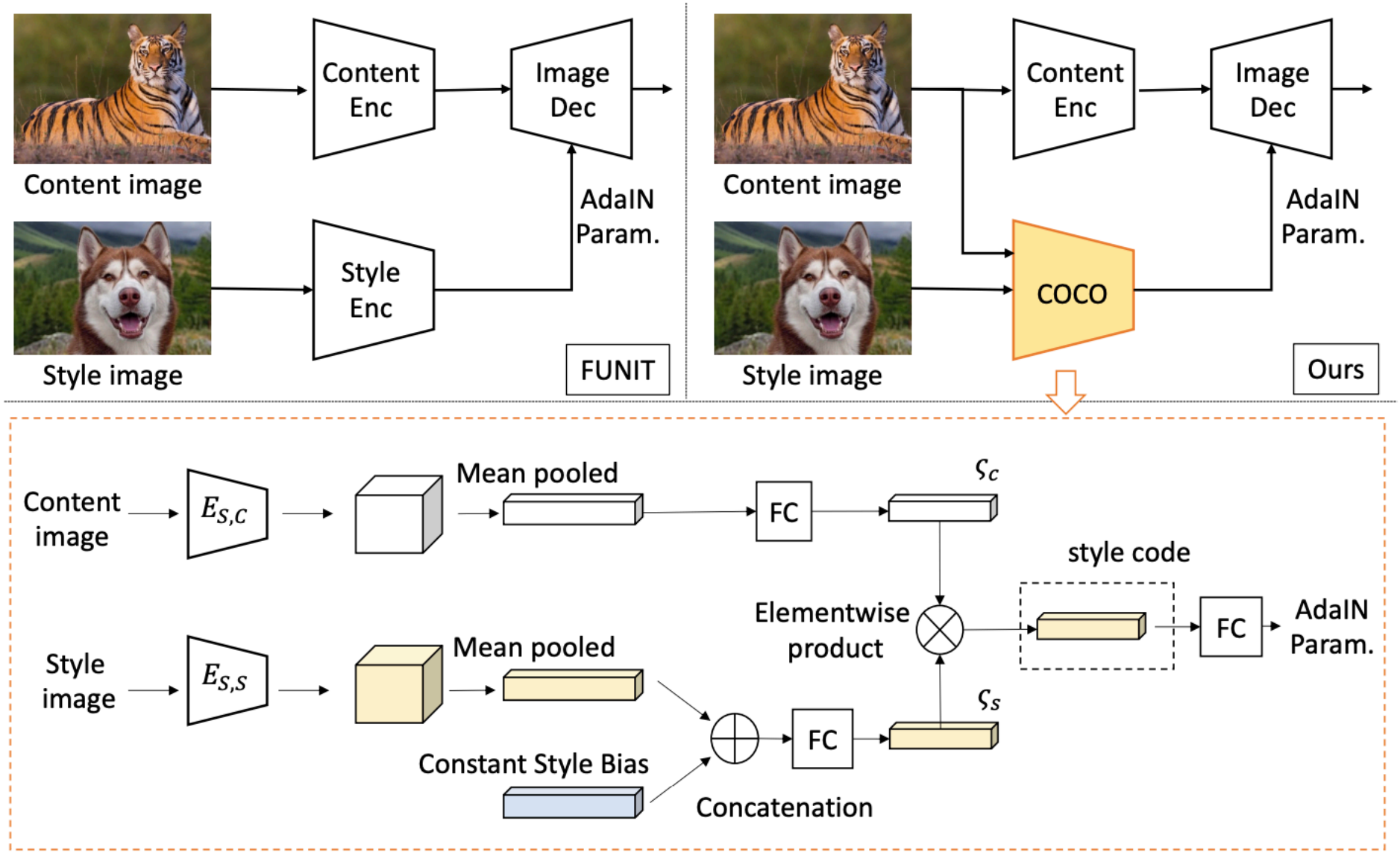
Motivation


FUNIT
Content
W/O COCO
Ours
Original Style
Cropped Styles
Ours
Ours w/o COCO
FUNIT
Deviation
0.6
0.4
0.2
0.0

Problem setting of the few-shot I2I

- Let X be a training set consists of images from K different domains.
- For each image in X, the class label is known.
- In the training phase, a model learns to map a content image in one domain to an analogous image in the domain of the input style examples.
- In the test phase, the model takes a few example images from an unseen domain, not included in K, and performs the translation.

Architecture



COntent-COnditioned style encoder (COCO) (1)

- Unlike the style encoder of the previous work, COCO takes both content and style image as input.
- With this content-conditioning scheme, we create a direct feedback path during learning to let the content image influence how the style code is computed.
- In other words, this mechanism produces a customized style code for the input content image.

COntent-COnditioned style encoder (COCO) (2)

- The constant style bias (CSB)
- The CSB provides a fixed input to the style encoder, which helps compute a style code that is less sensitive to the variations in the style image.
- When the CSB is activated, mostly texture-based appearance information is transferred.

COntent-COnditioned style encoder (COCO) (3)

- Authors found that replacing the vanilla convolutional layers in the original design with residual blocks improves the performances so does replacing the multi-task adversarial discriminator with the project-based discriminator.

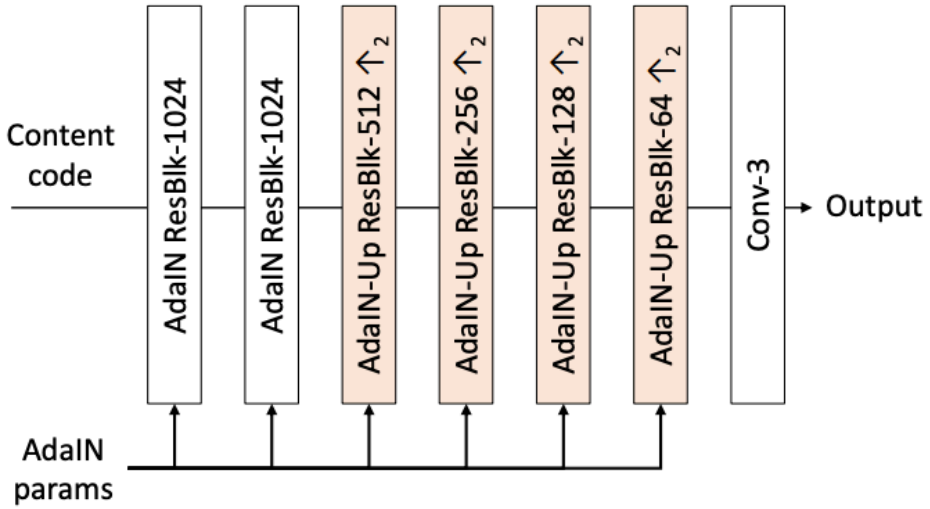


Fig. 14: Architecture of the image decoder F.

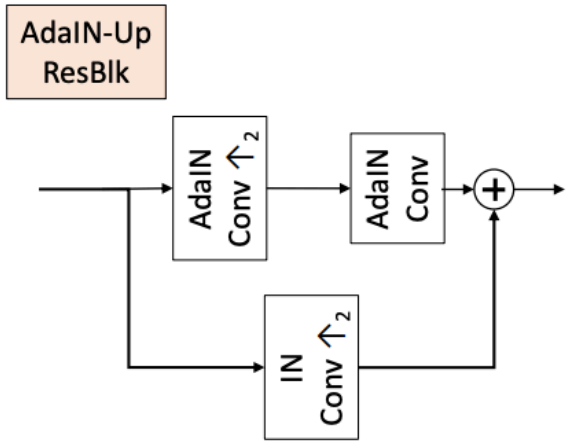


Fig. 15: Architecture of the AdaIN-Up residual block.

Objective functions

- GAN loss (D,G)
  - To ensure the realism of the generated images given the class of the style images.
- Reconstruction loss (G)
  - To reconstruct images when both the content and the style are from the same domain.
- Discriminator Feature matching loss (G)
  - For the effect of stabilizing the adversarial training.

Datasets

- Carnivores
  - Built the dataset using images from the ImageNet
  - 149 = 119 / 30
- Mammals
  - Built the dataset using image from Google search
  - 301 = 236 / 65
- Birds
- Motorbikes

Evaluation protocol

- In the test phase, authors randomly sample 25,000 content images and pair each of them with a few style images from a target class to compute the translation.
- In this work, authors use the one-shot setting as it is the most challenging few-shot setting.

Performance metrics

- To measure style faithfulness:
  - mFID: Compute FID for each of the target class and report their mean value.
- To measure content preservation:
  - An ideal translation should keep the structure of input content image unchanged.
  - Estimate the body-part segmentation masks by using DeeplabV3.
  - Calculate mIOU and pixel accuracy (PAcc) between masks for content and translated images.
- User-study

Qualitative results (1)

Fig. 7: Two-shot image translation results on the Carnivores dataset.

Qualitative results (2)

Fig. 8: Two-shot image translation results on the Birds dataset.

Qualitative results (3)

Fig. 9: Two-shot image translation results on the Mammals dataset.

Quantitative comparison

Table 1: Results on the benchmark datasets.

Dataset	Method	mFID ↓	PAcc ↑	mIoU ↑	User Style Preference ↑	User Content Preference ↑
Carnivores	FUNIT	147.8	59.8	44.6	16.5	11.9
	Ours	**107.8**	**66.5**	**52.1**	**83.5**	**88.1**
Mammals	FUNIT	245.8	35.3	23.3	23.6	27.8
	Ours	**109.3**	**48.8**	**35.5**	**76.4**	**72.2**
Birds	FUNIT	89.2	52.4	37.2	38.5	37.5
	Ours	**74.6**	**53.3**	**38.3**	**61.5**	**62.5**
Motorbikes	FUNIT	275.0	85.6	73.8	17.8	17,4
	Ours	**56.2**	**94.6**	**90.3**	**82.2**	**82.6**

Exp: Ablation study

Table 2: Ablation study on the Carnivores and Birds dataset. "Ours w/o CC" represents a baseline where the content conditioning part in COCO is removed. "Ours w/o CSB" represents a baseline where the CSB is removed. Detailed architecture of these baselines are given in Appendix A

Method	Carnivores			Birds		
	mFID↓	PAcc↑	mIou ↑	mFID↓	PAcc↑	mIou ↑
Ours w/o COCO	**99.6**	62.5	47.8	**68.8**	52.8	37.9
Ours w/o CSB	107.1	61.8	46.9	74.1	52.5	37.7
Ours w/o CC	110.0	**66.7**	**52.1**	75.3	52.8	37.9
Ours	107.8	66.5	**52.1**	74.6	**53.3**	**38.3**

Exp: effect of the CSB

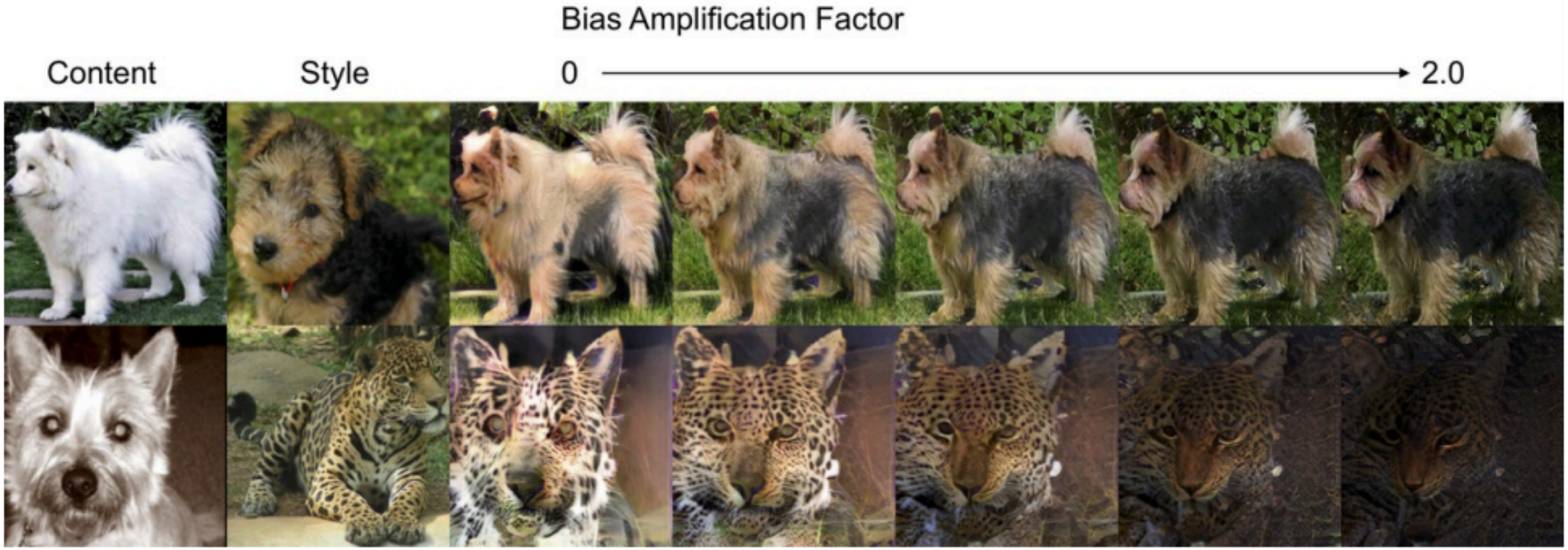


Fig. 10: By changing the amplification factor λ of the CSB, our model generates different translation outputs for the same pair of content and style images.

Exp: interpolation of the style codes

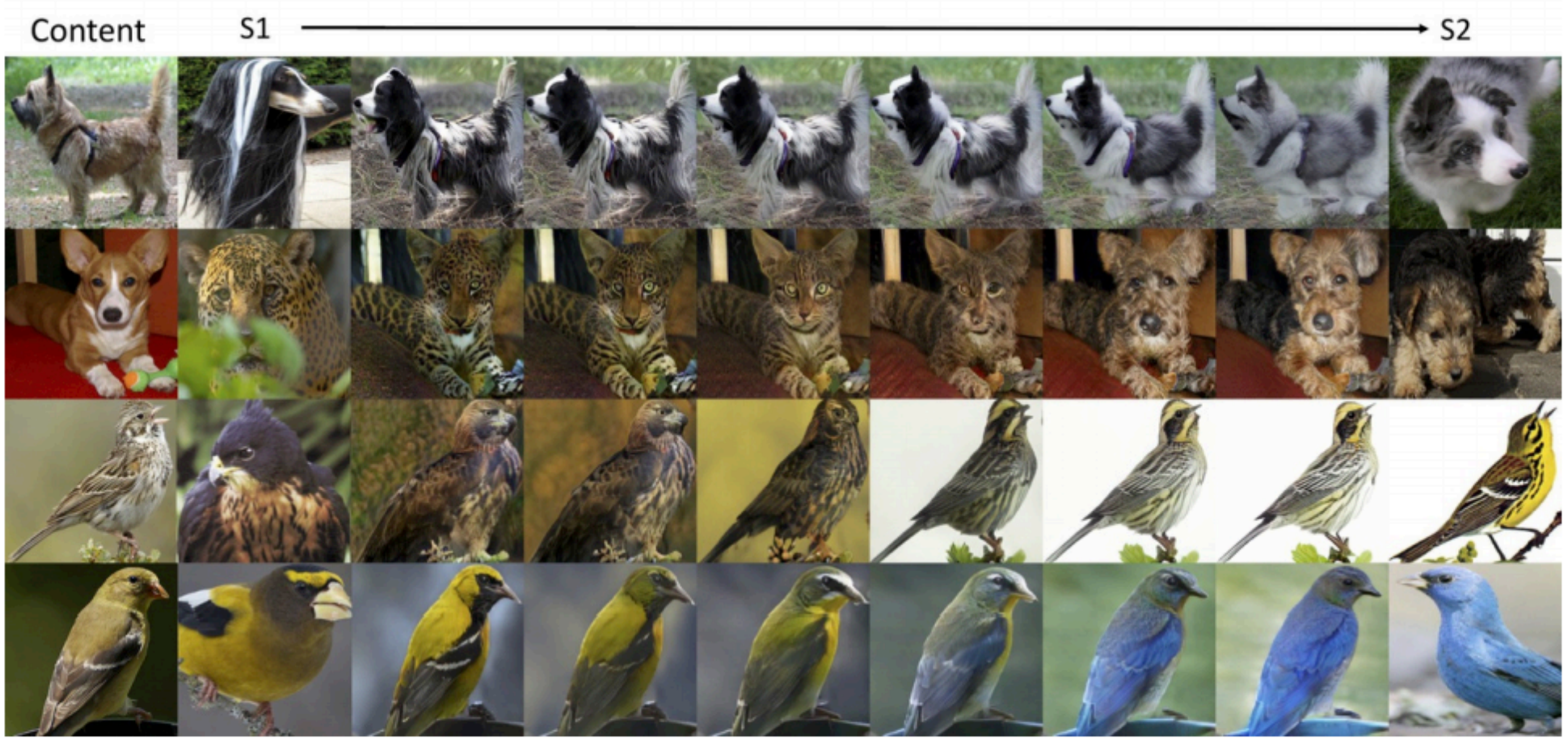


Fig. 11: We interpolate the style codes from two example images from two different unseen domains. Our model can generate photorealistic results using these interpolated style codes. More results are in the supplementary materials.

Summary