Transferring Unconditional to Conditional GANs with Hyper-Modulation Supplementary Material

Héctor Laria[1] Yaxing Wang[2] Joost van de Weijer[1] Bogdan Raducanu[1]
[1] Computer Vision Center, Barcelona, Spain
[2] Nankai University, China
{hlaria, yaxing, joost, bogdan}@cvc.uab.es

A. Discriminator details

A diagram of the discriminator and its modifications detailed in the experiments section can be seen in Figure 3. As underlined in the paper, it is worth noting that this module is finetuned from the source domain. Introducing weight modulation into it leads to training instability and worse generation quality overall.

B. Additional baseline comments

Results of mentioned baselines are shown in Table 4 for experiments on several domains. From these results, we can appreciate that *cGANTransfer* performs badly when the number of previous learned classes is low, since its generation power comes precisely from combining these previous classes. For instance, the first column sets the problem to perform transfer learning from FFHQ (1 class) to AFHQ (3 classes), where this method seems to perform considerably worse than the proposed work, and also worse than simply learning the batch normalization statistics, as *GAN Memory*. When the number of source domain classes for the transfer is slightly higher, as AFHQ (3 classes) to CelebA-HQ (2 classes), the result is somewhat better since it is allowed more expressiveness, but still lacking quality depending on the closeness of the target domain.

Results for several metrics on transfer learning from pre-trained FFHQ model to AFHQ dataset are shown in Table 5. We can see how the proposed method performs better than simply learn the normalization statistics for each class as in *GAN Memory* in terms of quality and diversity, since the knowledge of other classes learned concurrently can be propagated among all of them, resulting in improved time and data efficiency during training. We have already mentioned how *cGANTransfer* quality degrades when not enough pre-trained classes are given to produce an interpolation. However, we can see how the diversity is better than its quality. We assume this occurs because it can produce close-enough interpolations to resemble the target class, but it doesn't have a meaningful basis (*i.e.* a sufficient number of pre-trained classes) in order to form a combination of significant quality.

To perform experiments for cGANTransfer, a BigGAN model trained ImageNet was finetuned on FFHQ dataset. We used the checkpoint[1] of the highest resolution publicly available. The model with the best FID before collapse was used as the base for this method. The same was performed for AFHQ dataset.

C. Implementation details

As the base of our method, we use a public StyleGAN implementation [2], which while it is not official, it mostly reproduces results from the original paper. As already mentioned, we keep all hyperparameters from the original paper but fix the resolution growth to the final one, then apply all the methods explained.

In the original StyleGAN paper, it is mentioned training instability due to the depth of the mapping network. We experience a similar incident and therefore take the same solution of reducing the learning rate for the class network two orders of magnitude relative to the main network.

For the evaluation metrics, we use a ready-made package [2] for *FID*, *KID* and *Precision & Recall*, which uses the original Inception feature extractor weights, ported to PyTorch. *Density & Coverage* metrics have been implemented as a package extension, also included in this paper. Perceptual path similarity implementation is taken from [3] applying default center crop.

D. Architecture specifics

In this paper, we propose a novel transfer learning strategy from unconditional GAN to conditional GAN by introducing hypernetwork-based adaptive weight modulation.

[1] https://github.com/ajbrock/BigGAN-PyTorch
[2] https://github.com/rosinality/style-based-gan-pytorch
[3] https://github.com/rosinality/stylegan2-pytorch/blob/master/ppl.py

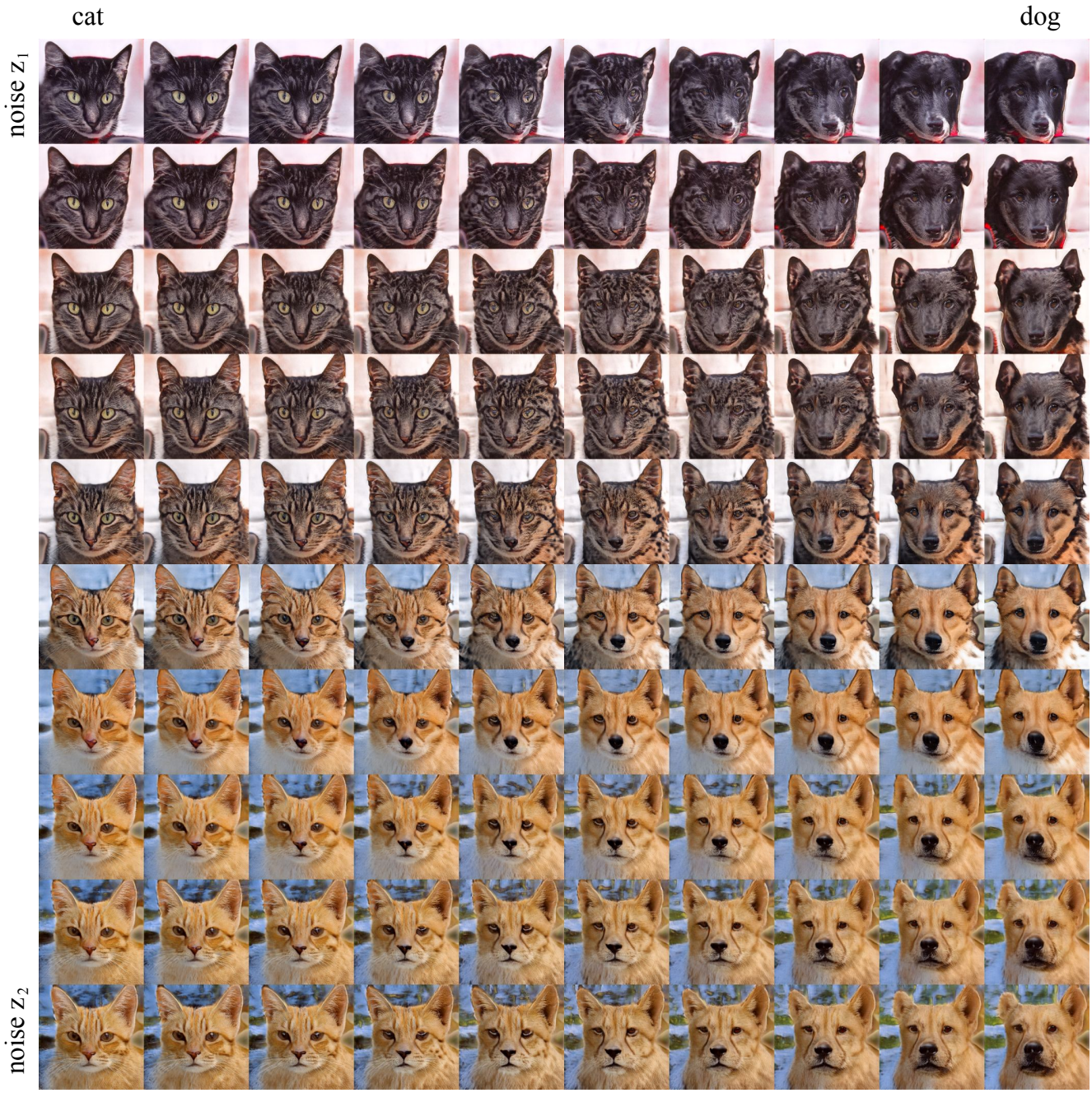


Figure 1. Sampling of class interpolation (left to right) versus noise interpolation (up to down). The hypernetwork has learned to keep every aspect of the style of an image intact, including background, while changing the class. The style mechanism was frozen since the beginning of the transfer learning training from human to animal faces.

Here we will detail the concrete architecture we used, and the changes applied to it.

Figure 4 shows the changes made to a vanilla StyleGAN [1]. The style branch is frozen since we want to keep the learned transformations (pose rotations, color changes, etc.) from the source domain, *i.e. FFHQ*, unchanged. We do not see a loss in performance when transferred to other datasets (see Fig. 1). The class network C is very similar to the original mapping network and also generates an embedding space, in this case $\mathcal{V}$, for classes, with the difference that the input comes from a learned class-embedding. The information then comes into each convolution layer to modulate the weights, as explained in the main paper.

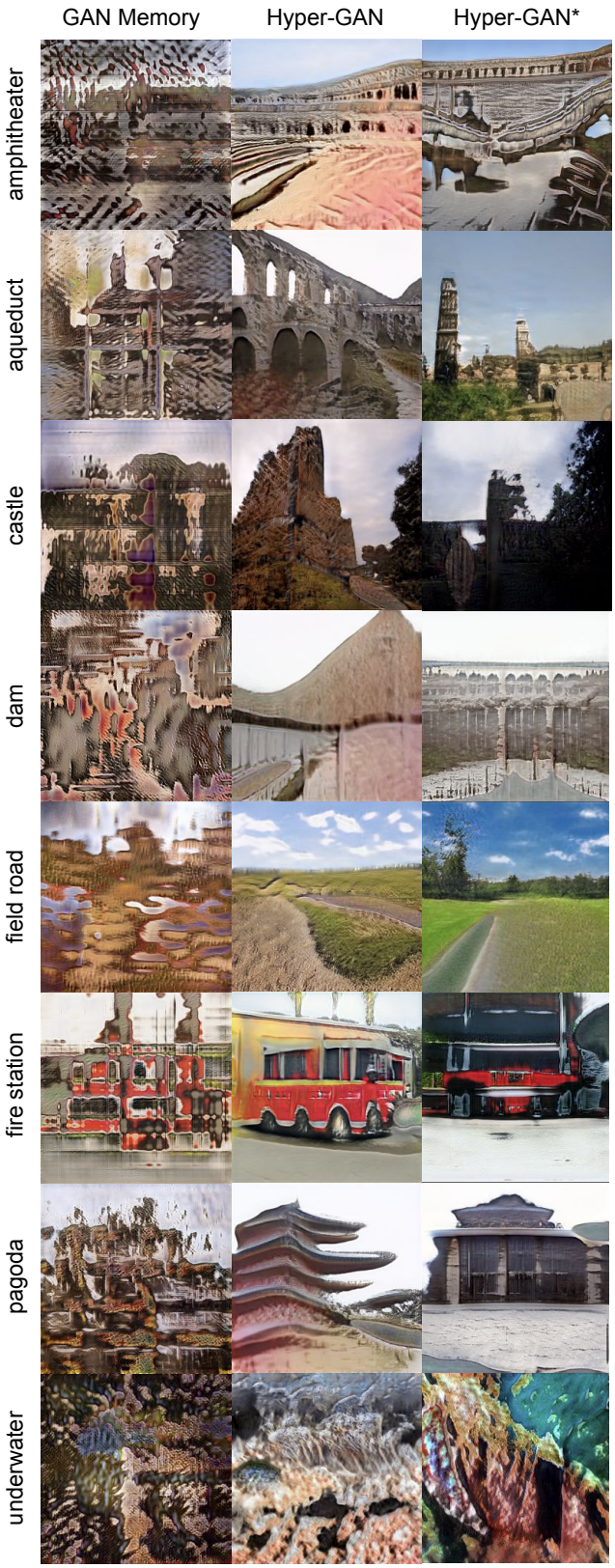


Figure 2. Transfer learning from Animal Faces (AFHQ) to a very distant domain (Places365).

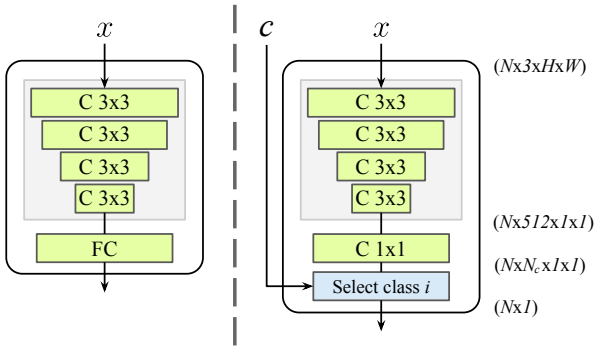


Figure 3. Original discriminator (left), modified discriminator (right). Batch dimensions at the right-most side for easier visualization. The final layer that comes from the convolutions is modified to output N_c number of classes, and the correct one is picked to compute the final loss.

E. Domain information injection

We specify here a different class information introduction technique to the one in the main paper. Since the normalization from one block will destroy the information included by the other (Fig. 5a), we can fix this and simplify the formulation by combining style and class as $\boldsymbol{\gamma}_{(\mathbf{s},\mathbf{v})}, \boldsymbol{\beta}_{(\mathbf{s},\mathbf{v})} = g(\mathbf{s}) + g(\mathbf{v}; \Phi)$, corresponding to Eq. (3), as seen in Fig. 5b. Nevertheless, we have experienced consistent underperformance when compared to the current weight modulation technique used.

References

[1] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 2, 4

[2] Sergey Kastryulin, Dzhamil Zakirov, and Denis Prokopenko. PyTorch Image Quality: Metrics and measure for image quality assessment, 2019. Open-source software available at https://github.com/photosynthesis-team/piq. 1

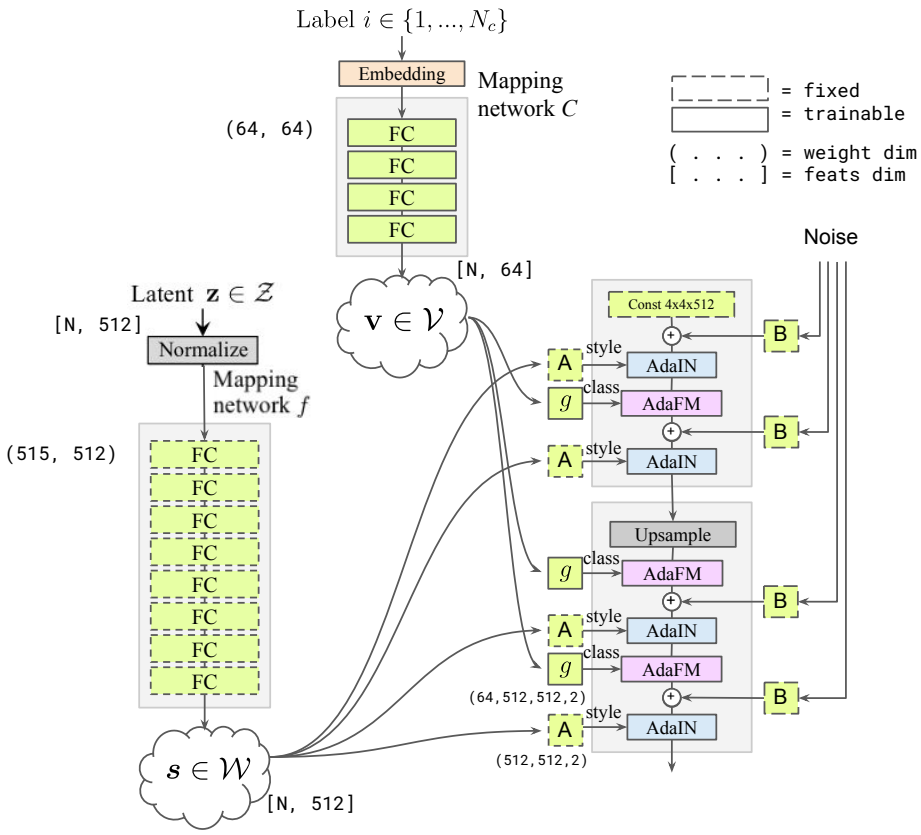
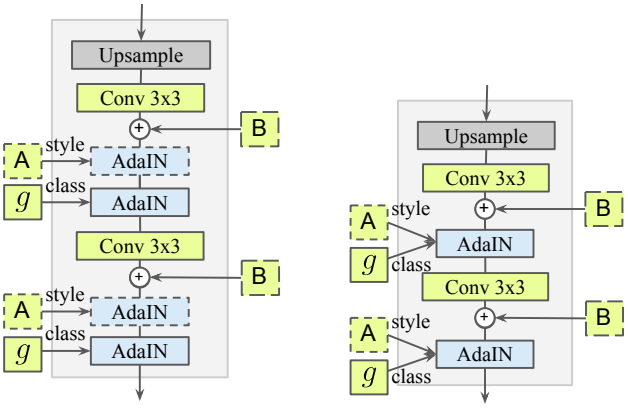


Figure 4. Hypernetwork architecture for transfer learning from unconditional GAN to conditional GAN, which wrap around the pre-trained weights (*i.e.*, the convolutional and fully connected layers). The network C takes as input the class index i, and outputs the class embedding $\mathbf{v}$, which is fed into the hypernetwork generators g. Adapted from [1].



(a) Naive class introduction. (b) Simplified.

Figure 5. Different fusion strategies. Dashed boxes are frozen during training.

Figure 6. Unfiltered conditional generations. Three rows of cats, dogs and wildlife respectively. The same noise is applied among classes (poses, background, etc).

Figure 7. Unfiltered conditional generations (config. B, worse than Fig. 6). Three rows of cats, dogs and wildlife respectively. The same noise is applied among classes (poses, background, etc.).

Figure 8. Sampling of class interpolation (left to right) versus noise interpolation (up to down).

Figure 9. Sampling of class interpolation between cat and dog.

Figure 10. Sampling of class interpolation between dog and wild life.

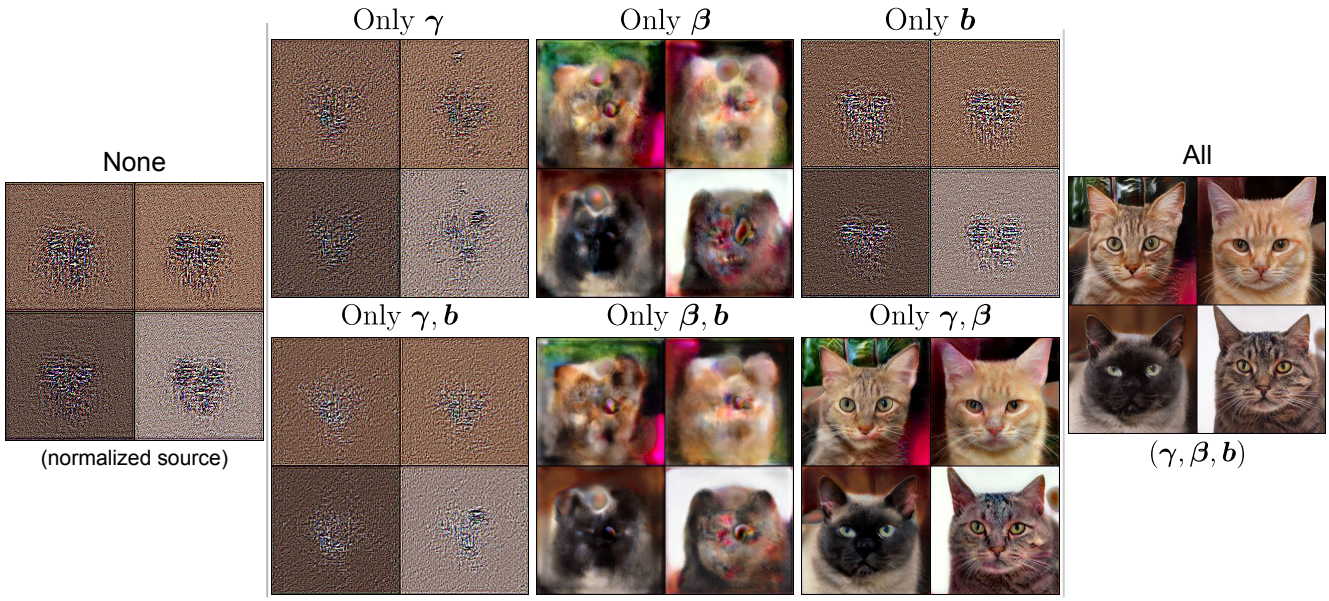


Figure 11. Effect of the modulation parameters on the domain transfer.