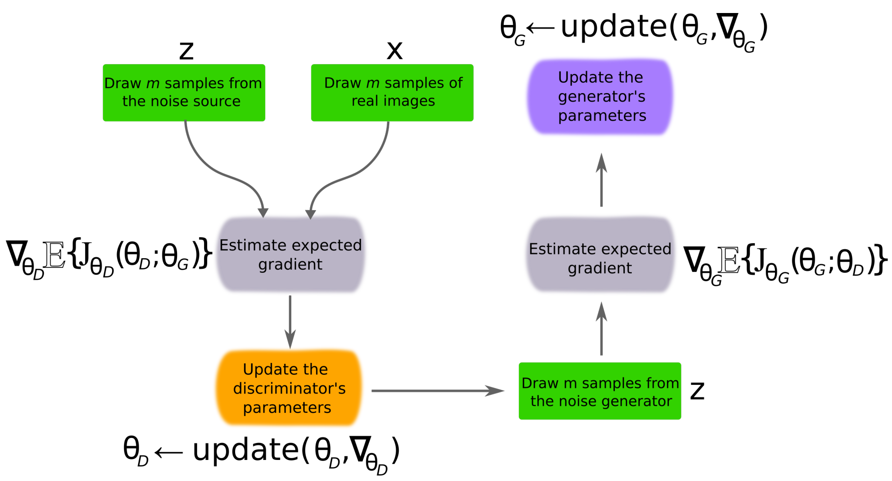

z
x
Draw *m* samples from the noise source
Draw *m* samples of real images
$\theta_G \leftarrow \text{update}(\theta_G, \nabla_{\theta_G})$
Update the generator's parameters
$\nabla_{\theta_D} \mathbb{E}\{J_{\theta_D}(\theta_D;\theta_G)\}$
Estimate expected gradient
Estimate expected gradient
$\nabla_{\theta_G} \mathbb{E}\{J_{\theta_G}(\theta_G;\theta_D)\}$
Update the discriminator's parameters
Draw m samples from the noise generator
z
$\theta_D \leftarrow \text{update}(\theta_D, \nabla_{\theta_D})$

Fig. 5. The main loop of GAN training. Novel data samples, $\mathbf{x}'$, may be drawn by passing random samples, $\mathbf{z}$ through the generator network. The gradient of the discriminator may be updated k times before updating the generator.

B. Training Tricks

One of the first major improvements in the training of GANs for generating images were the DCGAN architectures proposed by Radford et al. [5]. This work was the result of an extensive exploration of CNN architectures previously used in computer vision, and resulted in a set of guidelines for constructing and training both the generator and discriminator. In Section III-B, we alluded to the importance of strided and fractionally-strided convolutions [27], which are key components of the architectural design. This allows both the generator and the discriminator to learn good up-sampling and down-sampling operations, which may contribute to improvements in the quality of image synthesis. More specifically to training, batch normalization [28] was recommended for use in both networks in order to stabilize training in deeper models. Another suggestion was to minimize the number of fully connected layers used to increase the feasibility of training deeper models. Finally, Radford et al. [5] showed that using leaky ReLU activation functions between the intermediate layers of the discriminator gave superior performance over using regular ReLUs.

Later, Salimans et al. [25] proposed further heuristic approaches for stabilizing the training of GANs. The first, feature matching, changes the objective of the generator slightly in order to increase the amount of information available. Specifically, the discriminator is still trained to distinguish between real and fake samples, but the generator is now trained to match the discriminator's expected intermediate activations (features) of its fake samples with the expected intermediate activations of the real samples. The second, mini-batch discrimination, adds an extra input to the discriminator, which is a feature that encodes the distance between a given sample in a mini-batch and the other samples. This is intended to prevent mode collapse, as the discriminator can easily tell if the generator is producing the same outputs.

A third heuristic trick, heuristic averaging, penalizes the network parameters if they deviate from a running average of previous values, which can help convergence to an equilibrium. The fourth, virtual batch normalization, reduces the dependency of one sample on the other samples in the mini-batch by calculating the batch statistics for normalization with the sample placed within a reference mini-batch that is fixed at the beginning of training.

Finally, one-sided label smoothing makes the target for the discriminator 0.9 instead of 1, smoothing the discriminator's classification boundary, hence preventing an overly confident discriminator that would provide weak gradients for the generator. Sønderby et al. [29] advanced the idea of challenging the discriminator by adding noise to the samples before feeding them into the discriminator. Sønderby et al. [29] argued that one-sided label smoothing biases the optimal discriminator, whilst their technique, instance noise, moves the manifolds of the real and fake samples closer together, at the same time preventing the discriminator easily finding a discrimination boundary that completely separates the real and fake samples. In practice, this can be implemented by adding Gaussian noise to both the synthesized and real images, annealing the standard deviation over time. The process of adding noise to data samples to stabilize training was, later, formally justified by Arjovsky et al. [26].

C. Alternative formulations

The first part of this section considers other information-theoretic interpretations and generalizations of GANs. The