Consistency Regularization for Generative Adversarial Networks
(2020)

Han Zhang, Zizhao Zhang, Augustus Odena, Honglak Lee

Summary

Contributions

GaNs are known for their instability during training, and a number of regularization and normalization techniques were introduced to help with such a problem. For example, a popular regularization technique is penalizing the squared gradient norm for both the training data and the generated data, on the other hand, spectral normalization is a popular normalization. The problem with most of these regularization techniques is that, when combined with normalization, they don't add in any significant manner to the performance, and they might even negatively impact it. The author propose to use consistency regularization, a method popularized in semi-supervised learning to leverage unlabeled data during training. In this case, the discriminator is pushed to give similar results (real or fake image) with and without data augmentation. This will push the semantic feature space to have similar properties to the manifold were the input images reside. For example, as illustrated in the figure bellow, the image and its zoomed version needs to be close in the feature space as they are in the manifold.

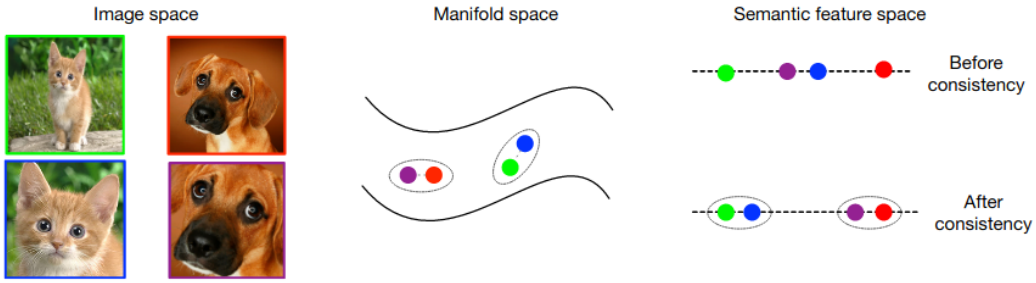

Figure 1: An illustration of consistency regularization for GANs. Before consistency regularization, the zoomed-in dog and the zoomed-in cat (bottom left) can be closer than they are to their original images in feature space induced by the GAN discriminator. This is illustrated in the upper right (the semantic feature space), where the purple dot is closer to the blue dot than to the red dot, and so forth. After we enforce consistency regularization based on the implicit assumption that image augmentation preserves the semantics we care about, the purple dot pulled closer to the red dot.

Method

A GAN contains a generator and a discriminator, the generator G takes as input a noise vector sampled from a prior p(z) $z \sim p(z)$ and generates an image. The discriminator D then tries to differentiate between generated examples and real ones $x \in \mathcal{X}$. By pushing the generator to confuse the discriminator, we will be able to generate images similar to the ones in $\mathcal{X}$. This is in a minimax training framework

$$L_D = -\mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] - \mathbb{E}_{z\sim p(z)}[1 - \log D(G(z))]$$
$$L_G = -\mathbb{E}_{z\sim p(z)}[\log D(G(z))]$$

The role of the discriminator is to be able to differentiate between real and fake images, and be invariant to any valid domain-specific data augmentations. The fact that an image is real or not should not be changed if we apply an additional augmentation like H-FLip or zoom.

So to achieve this, we add an additional consistency term to force the discriminator to have similar prediction over an image x and its transformed version $T(x)$

$$\min_D L_{cr} = \min_D \sum_{j=m}^{n} \lambda_j \left\| D_j(x) - D_j(T(x)) \right\|^2$$

The loss above consists of forcing not only the discriminator to have similar predictions, but even similar activations over different layers j. The authors however only use the last layer.

The algorithm is this case is as follows:

Algorithm 1 Consistency Regularized GAN (CR-GAN). We use $\lambda = 10$ by default.

Input: generator and discriminator parameters θ_G, θ_D, consistency regularization coefficient λ, Adam hyperparameters α, β_1, β_2, batch size M, number of discriminator iterations per generator iteration N_D

for number of training iterations **do**
  for $t = 1, ..., N_D$ **do**
    for $i = 1, ..., M$ **do**
      Sample $z \sim p(z)$, $x \sim p_{\text{data}}(x)$
      Augment x to get $T(x)$
      $L_{cr}^{(i)} \leftarrow \left\| D(x) - D(T(x)) \right\|^2$
      $L_D^{(i)} \leftarrow D(G(z)) - D(x)$
    end for
    $\theta_D \leftarrow \text{Adam}(\frac{1}{M} \sum_{i=1}^{M} (L_D^{(i)} + \lambda L_{cr}^{(i)}), \alpha, \beta_1, \beta_2)$
  end for
  Sample a batch of latent variables $\{z^{(i)}\}_{i=1}^{M} \sim p(z)$
  $\theta_G \leftarrow \text{Adam}(\frac{1}{M} \sum_{i=1}^{M} (-D(G(z))), \alpha, \beta_1, \beta_2)$
end for

Results

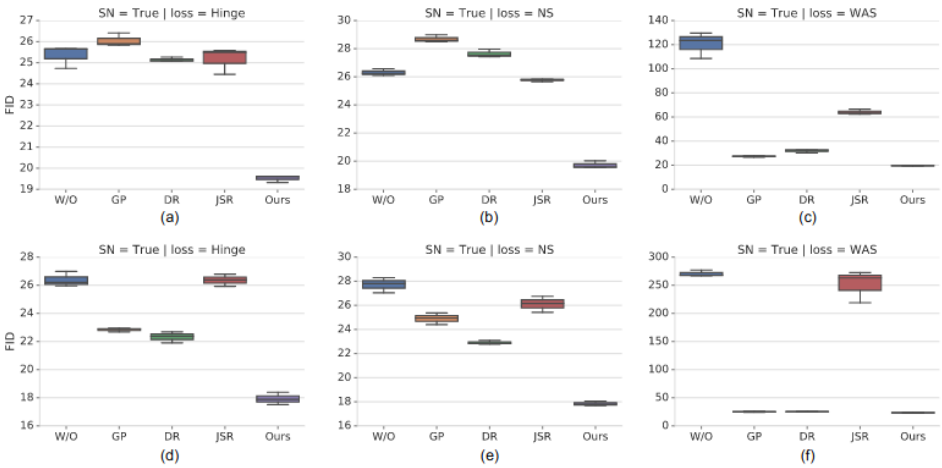

Figure 2: Comparison of our method with existing regularization techniques under different GAN losses. Techniques include no regularization (W/O), Gradient Penalty (GP) (Gulrajani et al., 2017), DRAGAN (DR) (Kodali et al., 2017) and JS-Regularizer (JSR) (Roth et al., 2017). Results (a-c) are for CIFAR-10 and results (d-f) are for CelebA.

Setting	W/O	GP	DR	JSR	Ours (CR-GAN)
CIFAR-10 (SNDCGAN)	24.73	25.83	25.08	25.17	**18.72**
CIFAR-10 (ResNet)	19.00	19.74	18.94	19.59	**14.56**
CelebA (SNDCGAN)	25.95	22.57	21.91	22.17	**16.97**

Table 1: Best FID scores for unconditional image generation on CIFAR-10 and CelebA.

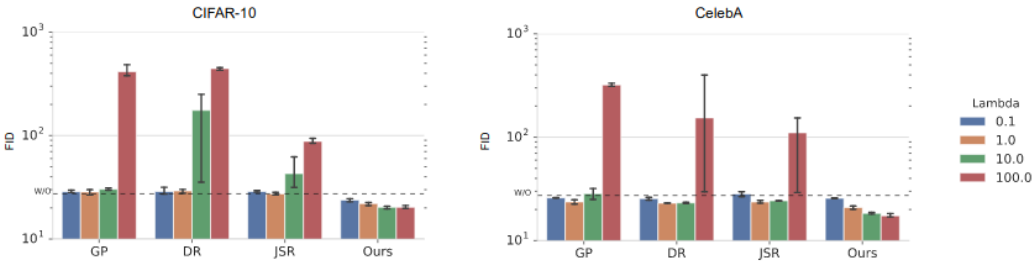

Figure 3: Comparison of FID scores with different values of the regularization coefficient λ on CIFAR-10 and CelebA. The dotted line is a model without regularization.

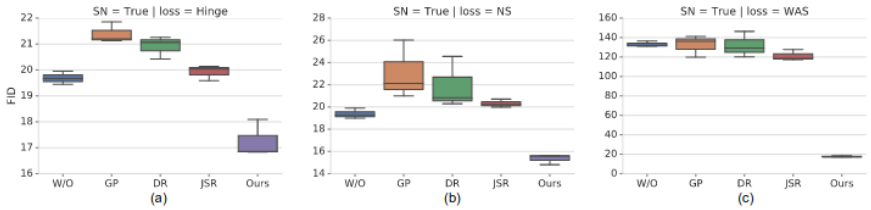

Figure 4: Comparison of FID scores with ResNet structure on different loss settings on CIFAR-10.

Dataset	SNGAN	SAGAN	BigGAN	BigGAN*	CR-BigGAN*
CIFAR-10	17.5	/	14.73	20.42	**11.48**
ImageNet	27.62	18.65	8.73	7.75	**6.66**

Table 2: Comparison of our technique with state-of-the-art GAN models including SNGAN (Miyato & Koyama, 2018), SAGAN (Zhang et al., 2019) and BigGAN (Brock et al., 2018) for class conditional image generation on CIFAR-10 and ImageNet in terms of FID. BigGAN* is the BigGAN implementation of Kurach et al. (2019). CR-BigGAN* has the exactly same architecture as BigGAN* and is trained with the same settings. The only difference is CR-BigGAN* adds consistency regularization.

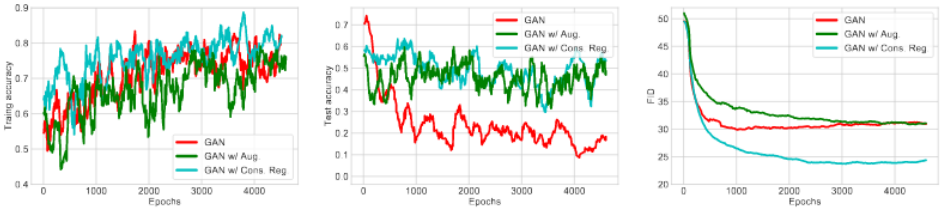

Figure 5: A study of how much data augmentation matters by itself. Three GANs were trained on CIFAR-10: one baseline GAN, one GAN with data augmentation only, and one GAN with consistency regularization. (**Left**) Training accuracy of the GAN discriminator. (**Middle**) Test accuracy of the GAN discriminator on the held out test set. The accuracy is low for the baseline GAN, which indicates it suffered from over-fitting. The accuracy for the other two is basically indistinguishable for each other. This suggests that augmentation by itself is enough to reduce discriminator over-fitting, and that consistency regularization by itself does little to address over-fitting. (**Right**) FID scores of the three settings. The score for the GAN with only augmentation is not any better than the score for the baseline, even though its discriminator is not over-fitting. The score for the GAN with consistency regularization is better than both of the others, suggesting that the consistency regularization acts on the score through some mechanism other than by reducing discriminator over-fitting.

Metric	Gaussian Noise	Random shift & flip	Cutout	Cutout w/ random shift & flip
FID	21.91±0.32	16.04±0.17	17.10±0.29	19.46±0.26

Table 3: FID scores on CIFAR-10 for different types of image augmentation. Gaussian noise is the worst, and random shift and flip is the best, consistent with general consensus on the best way to perform image optimization on CIFAR-10 (Zagoruyko & Komodakis, 2016).