The New Involution Operator and Its Application on Generative Adversarial Networks

Group 33: Xiaozhu Fang, Minghao Li, Rui Jin

The Chinese University of Hong Kong, Shenzhen, China

May 17, 2021

Involution Operator

CVPR2021(June 19th) announced the accepted papers in March. A group from *HKUST&ByteDance* addressed an innovative operator, involution, which functions similarity to the typical convolution kernel.

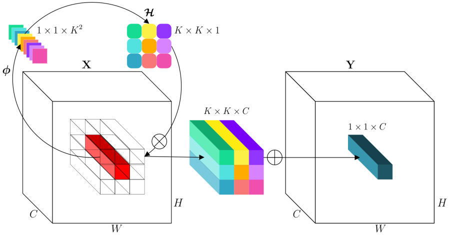


Figure 1: Schematic illustration of involution. The involution kernel $\mathscr{H}_i, j \in \mathbb{R}^{K\times K\times 1}$ ($G = 1$ in this example for ease of demonstration) is yielded from the function ϕ conditioned on a single pixel at (i, j), followed by a channel-to-space rearrangement.

$$\mathscr{H}_{i,j} = \phi(X_{i,j}) = W_1\sigma(W_0X_{i,j}), \quad Y_{i,j} = \sum_{(p,q)\in\Omega} \mathscr{H}_{i,j,p,q}X_{p,q} \tag{1}$$

Motivation

Involution is designed to be more efficient than convolution.

- Convolution:
  1. spatial-agnostic: shares parameters in space to satisfy the shift invariant system.
  2. channel-specific: exists redundant parameters in channel dimension.
- Involution:
  1. spatial-specific: distinguishes kernel in space, but share the hyperparameters to generate kernel(like self-attention).
  2. channel-agnostic: reduce the memory in channel to increase the spatial kernel size.

Involution is a more general form of self-attention.

- Self-attention:

$$Y_{i,j} = \sum_{(p,q)\in\Omega} \underbrace{((XW^Q)(XW^K)^T)_{i,j,p,q}}_{\approx \mathscr{H}_{i,j,p,q}} ((XW^V))_{p,q} \tag{2}$$

- Involution:

$$\mathscr{H}_{i,j} = \phi(X_{i,j}) = W_1\sigma(W_0X_{i,j}), \quad Y_{i,j} = \sum_{(p,q)\in\Omega} \mathscr{H}_{i,j,p,q}X_{p,q} \tag{3}$$

GAN

- WGAN-GP
  the optimal function $f^*(x)$ optimizing $\max_{||f||_L \leq 1} \mathbb{E}_{x\sim\mathbb{P}_r}[f(x)] - \mathbb{E}_{x\sim\mathbb{P}_g}[f(x)]$ has gradient norm 1 at almost everywhere under $\mathbb{P}_r$ and $\mathbb{P}_g$. This insight leads to the implementation of a soft gradient penalty norm to constrain the gradient norm to be close to 1. The new loss is as follows,

$$L = \mathbb{E}_{\tilde{x}\sim\mathbb{P}_g}[D(\tilde{x})] - \mathbb{E}_{x\sim\mathbb{P}_r}[D(x)] + \lambda \mathbb{E}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}[(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2] \tag{4}$$

- FID
  The core idea for FID is using Fréchet Distance to test the similarity between model samples and real samples.

$$d^2((m,C),(m_w,C_w)) = ||m - m_w||_2^2 + \mathrm{Tr}(C + C_w - 2(CC_w)^{1/2})$$

  The smaller FID we get from the generative model, represent a better performance of the model to generate the simulation images, which means more ”real” those generated images are.

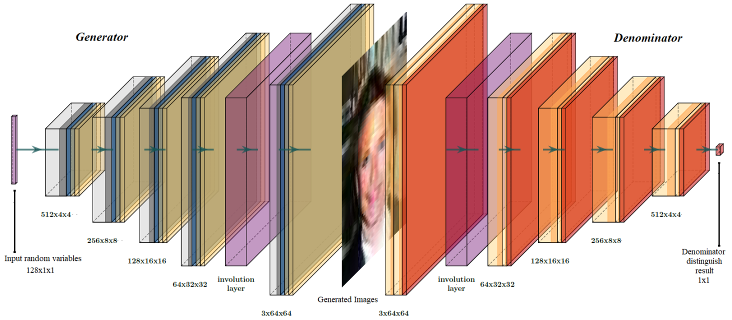
Experiments setting

- Dataset:#202k 64*64*3 pixels images
  CelebA dataset (http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html).
  (MNIST dataset) (http://yann.lecun.com/exdb/mnist/)

Experiments setting

Table 1: Training parameters for GAN models.

attribute	value	attribute	value
batch size	64	total steps	50000
λ(penalty)	10	latent dimension	128
optimizer	Adam	learning rate decay	0.95
β_1(Adam)	0	β_2(Adam)	0.9
generator Learning rate	0.0001	discriminator Learning rate	0.0004



Result

Figure 3: Generative images by iGAN(left),SAGAN(mid) and DCGAN(right).

Result

Figure 4: SAGAN(left) and iGAN(right), the images of iGAN are more saturated and smoother in color

Result

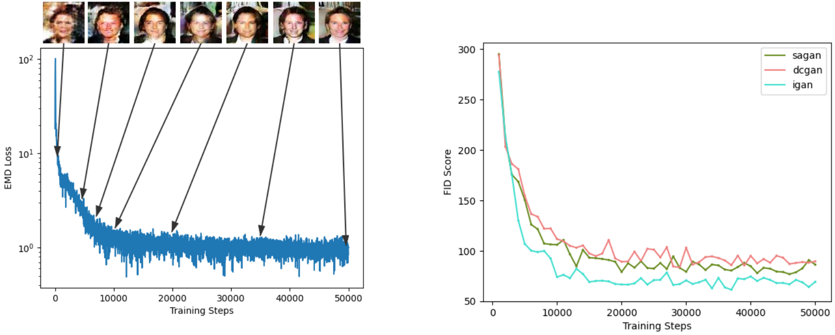


Figure 5: EMD loss of iGAN and FID Score of different GANs(left) and of iGAN with various hyperparameters(right).

Result: hyperparameters

Table 2: Hyperparameters of iGAN and its influence. Channel for the involution layer is C=64.

Case	kernel size	group	reduction ratio	Accuracy	Memory	Computation
#	K	G	R	min(FID)	#parameters	time per step
1	3× 3	4	2	61.51	800	0.154
2	7× 7	4	8	76.00	520	0.333
3	7× 7	4	2	112.85	2080	0.342
4	3× 3	4	8	76.01	200	0.152
5	3× 3	1	2	62.34	2336	0.152
6	1× 1	4	2	68.73	644	0.120
7	5× 5	4	2	63.37	1312	0.230

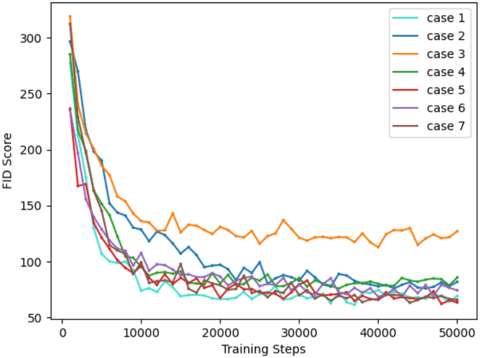
1)Accuracy is equivalent to min(FID), the minimum value of FID curve(Fig.5-right).
2)Memory is estimated by the number of weight parameters of the involution layer, which is expressed by

$$\#\text{parameters} \propto \left(\frac{C}{G} + K*K\right) \times \frac{C}{R \times G} \times G \tag{5}$$

3)Computation is represented the total time cost of each steps while other layers are fixed.

Result: hyperparameters

Case	kernel size	group	reduction ratio	Accuracy	Memory	Computation
#	K	G	R	min(FID)	#parameters	time per step
1	3× 3	4	2	61.51	800	0.154
2	7× 7	4	8	76.00	520	0.333
3	7× 7	4	2	112.85	2080	0.342
4	3× 3	4	8	76.01	200	0.152
5	3× 3	1	2	62.34	2336	0.152
6	1× 1	4	2	68.73	644	0.120
7	5× 5	4	2	63.37	1312	0.230



Result: hyperparameters

Case	kernel size	group	reduction ratio	Accuracy	Memory	Computation
#	K	G	R	min(FID)	#parameters	time per step
1	3× 3	4	2	61.51	800	0.154
2	7× 7	4	8	76.00	520	0.333
3	7× 7	4	2	112.85	2080	0.342
4	3× 3	4	8	76.01	200	0.152
5	3× 3	1	2	62.34	2336	0.152
6	1× 1	4	2	68.73	644	0.120
7	5× 5	4	2	63.37	1312	0.230

- The best K is around 4×4, which is close to the common convlution kernel size. Larger K costs more computation and memory, but it does not arise FID as expected.
- The group G hardly affects FID but the larger one save more memory
- The reduction ratio R affects FID, but the trade off exists between the memory and FID.

Result: stability issues

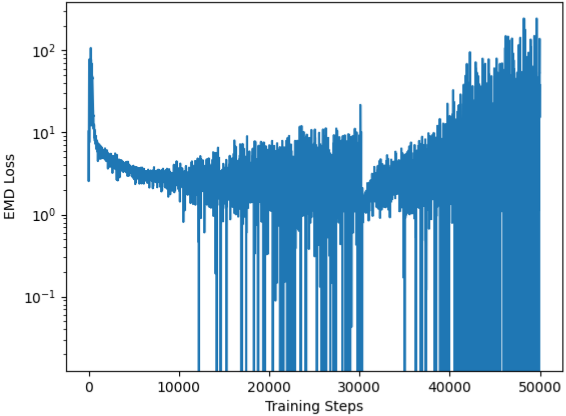


Figure 7: loss of collapsed model due to successive two involution layers

Result: stability issues

Figure 8: Generative Figures by collapsed model due to successive two involution layers

The contradiction of normalization and WGAN-GP makes it harder to ensure the stability.

- Number of involution layers. Two successive layers are unstable
- Normalization. Batch normalization, spectral normalization, softmax does not work.

Thank you

- Li, Duo, et al. "Involution: Inverting the Inherence of Convolution for Visual Recognition." arXiv preprint arXiv:2103.06255 (2021). CVPR2021.