UNSUPERVISED REPRESENTATION LEARNING WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS

Alec Radford & Luke Metz
indico Research
Boston, MA
falec,lukeg@indico.io

Soumith Chintala
Facebook AI Research
New York, NY
soumith@fb.com

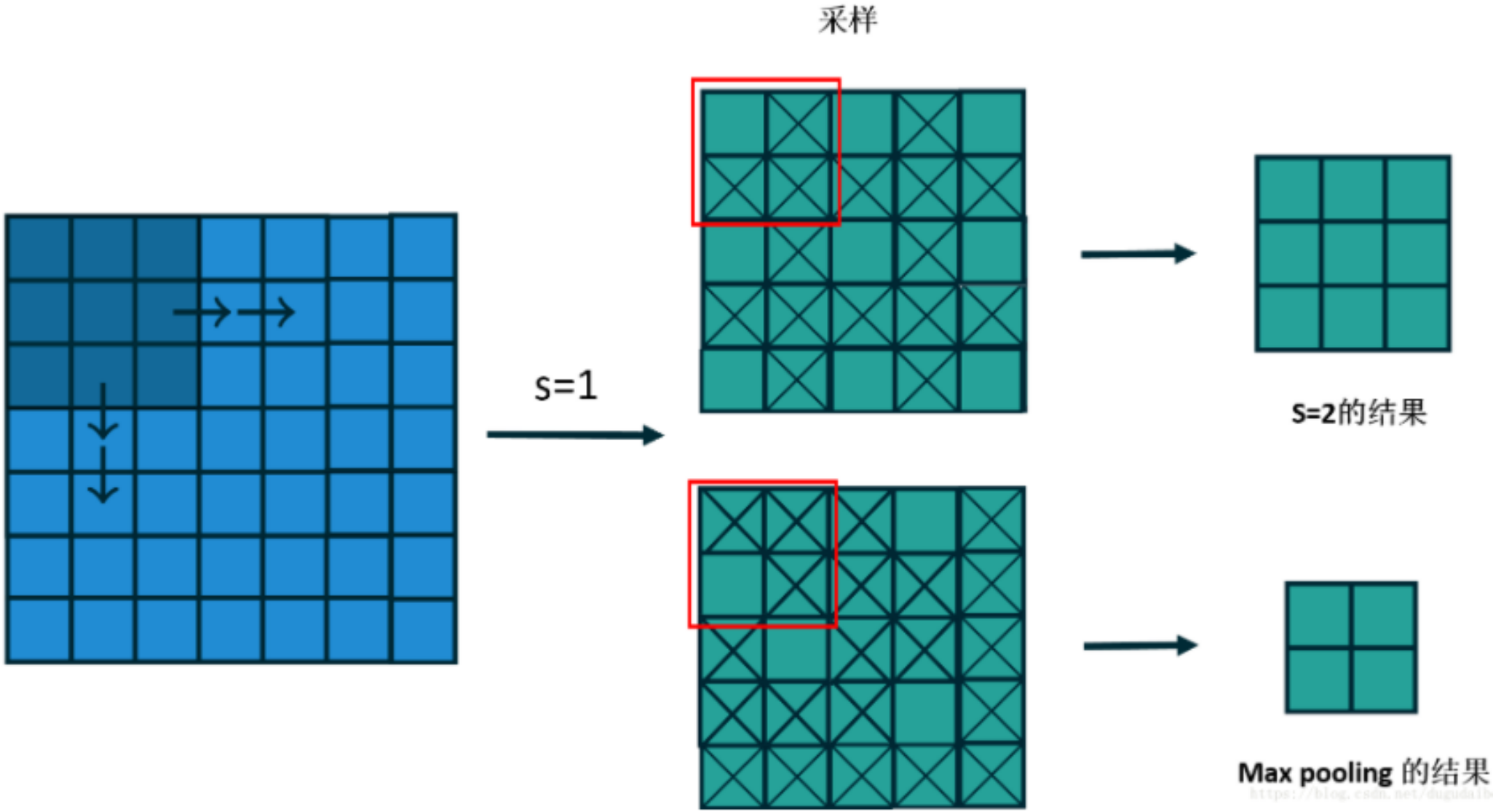
Contributions

1. 提出了一类基于卷积神经网络的GANs，称为DCGAN，它在多数情况下训练是稳定的。
2. 与其他非监督方法相比，DCGAN的discriminator提取到的图像特征更有效，更适合用于图像分类任务。
3. 通过训练，DCGAN能学到有意义的 filters。
4. DCGAN的generator能够保持latentspace到image的“连续性”。

Trick1

采用全卷积神经网络。不使用空间池化，取而代之使用带步长的卷积层（strided convolutions）。这么做能让网络自己学习更合适的空间下采样方法。

对于generator来说，要做上采样，采用的是分数步长的卷积（fractional-strided convolutions）(实做上，将数据的行和列分别重复2次)；对于discriminator来说，一般采用整数步长的卷积。


采样
s=1
S=2的结果
Max pooling 的结果

Trick2

避免在卷积层之后使用全连接层。全连接层虽然增加了模型的稳定性，但也减缓了收敛速度。一般来说，generator的第一层可以采用全连接，因为只是矩阵乘法，但结果需要reshape为4维tensor；discriminator的最后一个卷积层一般先摊平（flatten），然后接一个单节点的sigmoid。

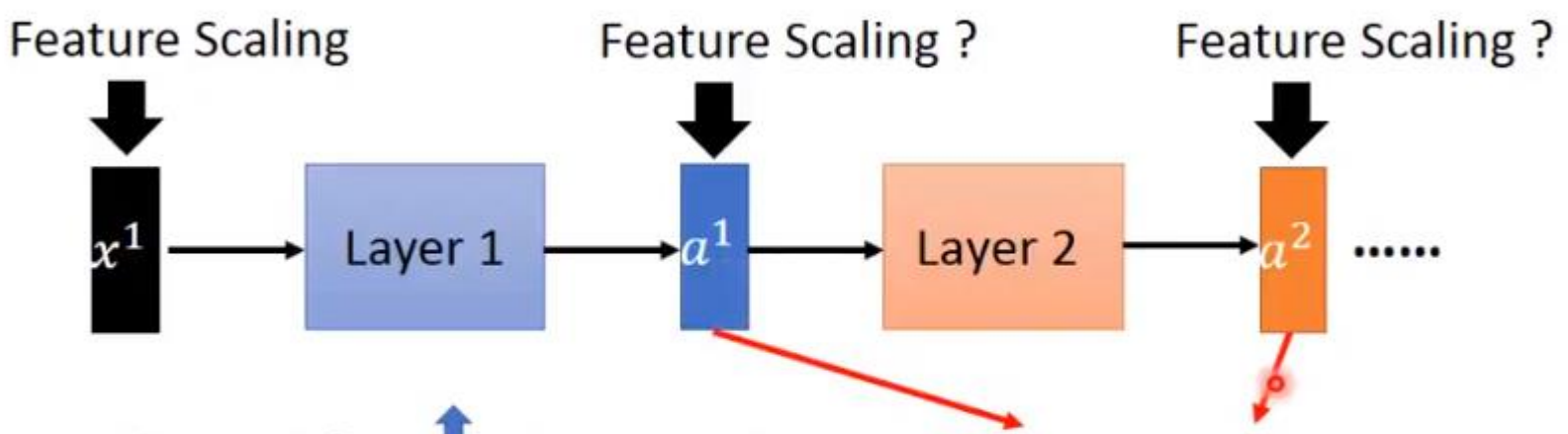
Trick3

除了generator的输出层和discriminator的输入层以外，其他层都是采用batch normalization。Batch normalization能确保每个节点的输入都是均值为0，方差为1。即使是初始化很差，也能保证网络中有足够强的梯度，让generator更deep，避免mode collapse。

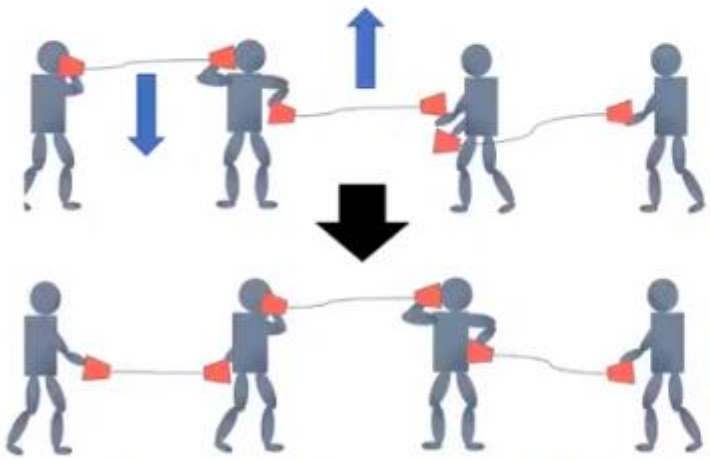
BN的作用就是将输入值进行标准化，降低scale的差异至同一个范围内。这样做的好处在于一方面提高梯度的收敛程度，加快训练速度；另一方面使得每一层可以尽量面对同一特征分布的输入值，减少了变化带来的不确定性，也降低了对后层网络的影响，各层网络变得相对独立。

Trick

How about Hidden Layer?



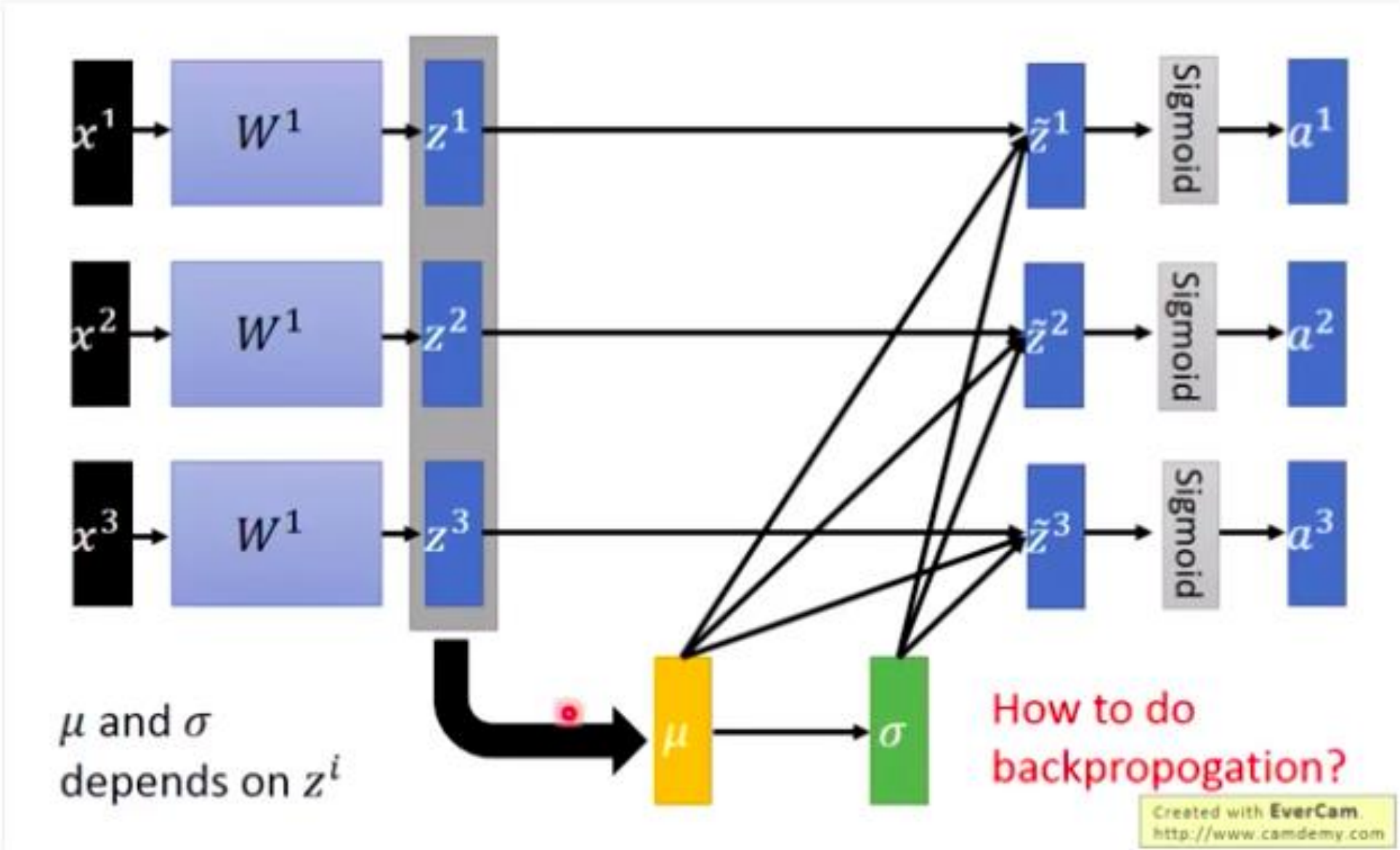
Difficulty: their statistics change during the training ...



Internal Covariate Shift

Smaller learning rate can be helpful, but the training would be slower.

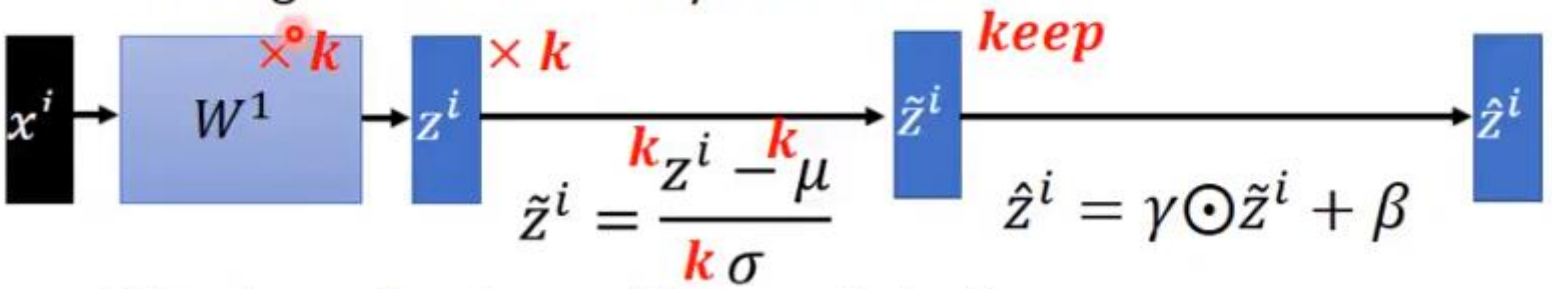
Tr


x^1
x^2
x^3
W^1
W^1
W^1
z^1
z^2
z^3
$\tilde{z}^1$
$\tilde{z}^2$
$\tilde{z}^3$
Sigmoid
Sigmoid
Sigmoid
a^1
a^2
a^3
μ and σ
depends on z^i
μ
σ
How to do
backpropogation?
Created with EverCam.
http://www.camdemy.com

Trick

Batch normalization - Benefit

- BN reduces training times, and make very deep net trainable.
  - Because of less Covariate Shift, we can use larger learning rates.
  - Less exploding/vanishing gradients
    - Especially effective for sigmoid, tanh, etc.
- Learning is less affected by initialization.

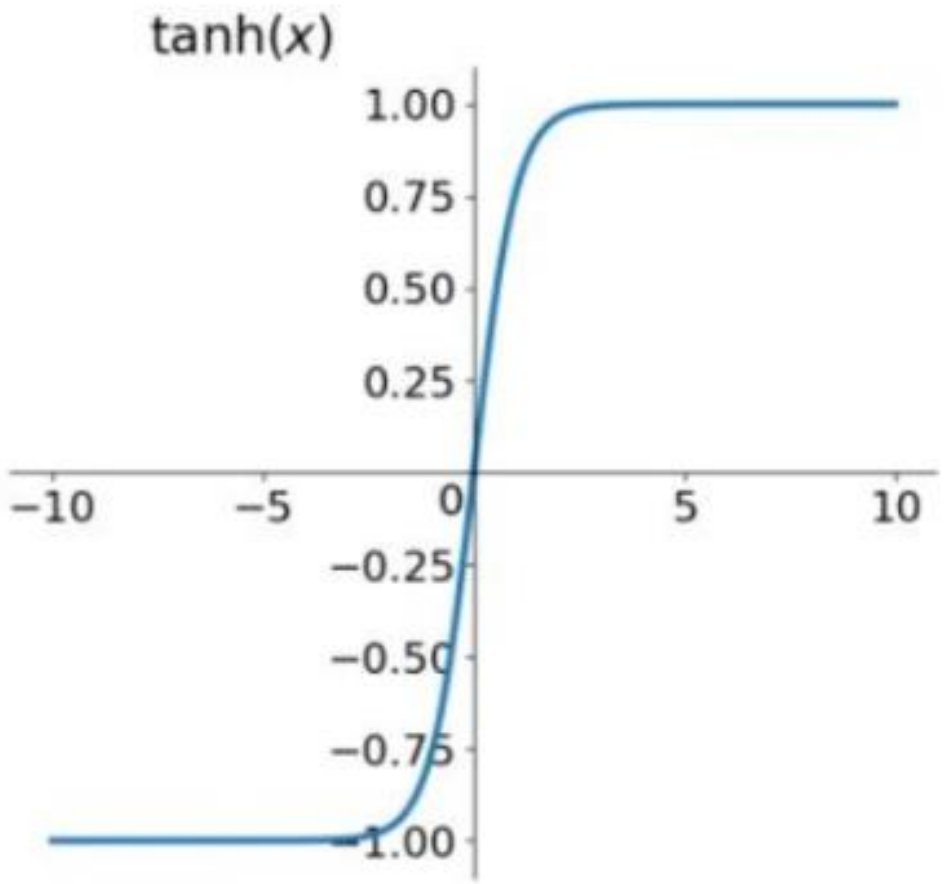


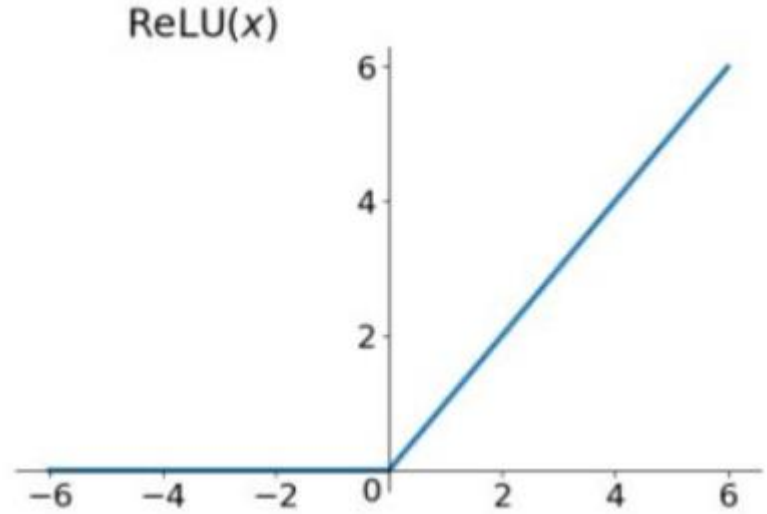
- BN reduces the demand for regularization.

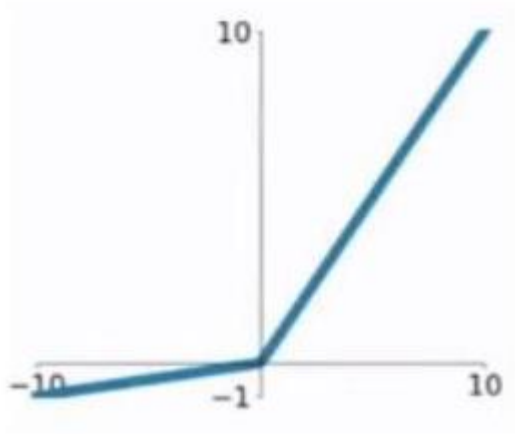
Trick4

对于generator，输出层的激活函数采用Tanh，其它层的激活函数采用ReLU。ReLU函数的输出可能会很大，而tanh函数的输出是在-1～1之间的，只要将tanh函数的输出加1再乘以127.5可以得到0～255 的像素值。

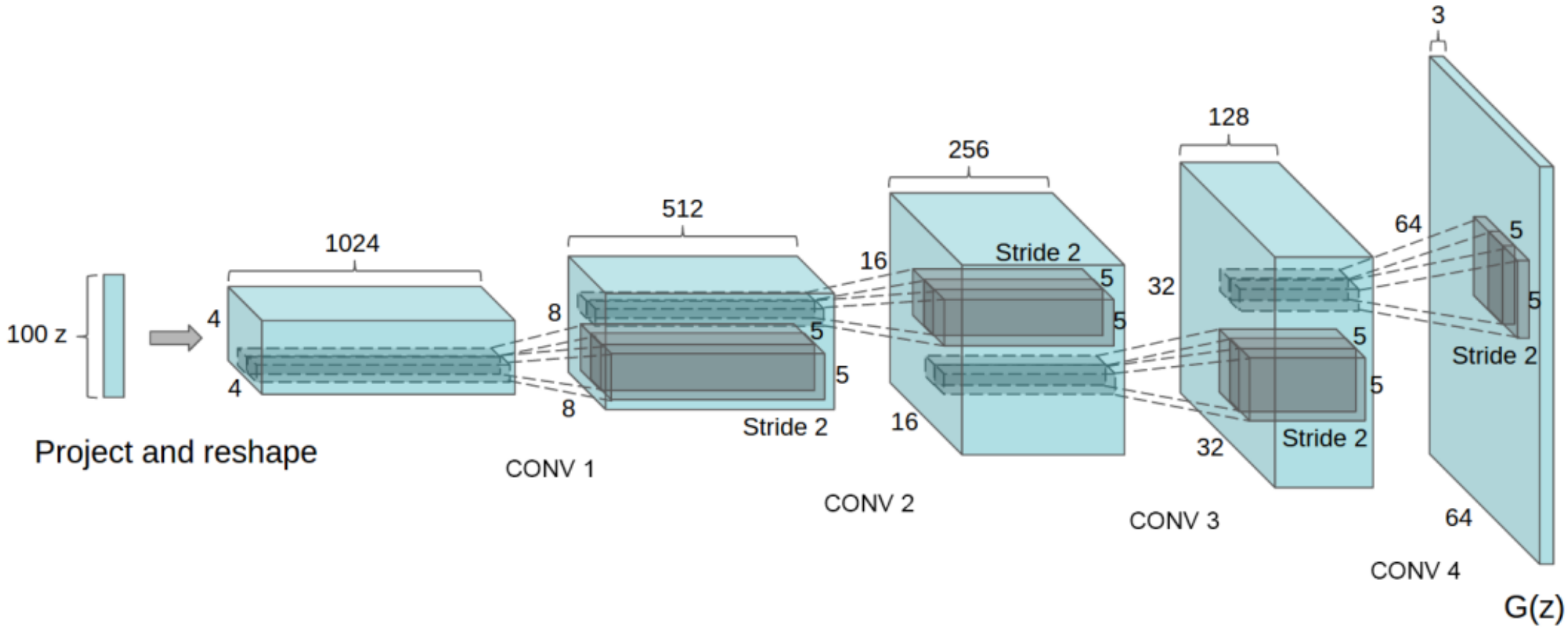
对于discriminator，激活函数采用LeakyReLU。


tanh(x)
1.00
0.75
0.50
0.25
−0.25
−0.50
−0.75
−1.00
−10
−5
0
5
10


ReLU(x)
6
4
2
−6
−4
−2
0
2
4
6


10
−10
−1
10

LeakyReLU


100 z
1024
4
4
Project and reshape
512
8
8
5
5
Stride 2
CONV 1
256
16
16
Stride 2
5
5
CONV 2
128
32
32
5
5
Stride 2
CONV 3
3
64
64
5
5
Stride 2
CONV 4
G(z)

DETAILS OF ADVERSARIAL TRAINING

1. Datasets: Large-scale Scene Understanding (LSUN), Imagenet-1k and a newly assembled Faces dataset
2. mini-batch SGD Batch size: 128;
3. Learning rate: 0.0002
4. Leak of leaky ReLU: 0.2
5. 输入线性映射到[-1, 1]
6. 所有权重用均值为0，方差为0.02的正态分布随机初始化
7. optimizer采用Adam，其中beta_1 = 0.5

Over-fitting?

model with a trating

Over-fitting?

DEDUPLICATION

To further decrease the likelihood of the generator memorizing input examples (Fig.2) we perform a simple image de-duplication process. We fit a 3072-128-3072 de-noising dropout regularized RELU autoencoder on 32x32 downsampled center-crops of training examples. The resulting code layer activations are then binarized via thresholding the ReLU activation which has been shown to be an effective information preserving technique (Srivastava et al., 2014) and provides a convenient form of semantic-hashing, allowing for linear time de-duplication . Visual inspection of hash collisions showed high precision with an estimated false positive rate of less than 1 in 100. Additionally, the technique detected and removed approximately 275,000 near duplicates, suggesting a high recall.

CLASSIFYING CIFAR-10 USING GANS AS A FEATURE EXTRACTOR

To evaluate the quality of the representations learned by DCGANs for supervised tasks,we train on Imagenet-1k and then use the discriminator's convolutional features from all layers,maxpooling each layers representation to produce a 4 × 4 spatial grid. These features are then flattened and concatenated to form a 28672 dimensional vector and a regularized linear L2-SVM classifier is trained on top of them

CLASSIFYING CIFAR-10 USING GANS AS A FEATURE EXTRACTOR

Table 1: CIFAR-10 classification results using our pre-trained model. Our DCGAN is not pre-trained on CIFAR-10, but on Imagenet-1k, and the features are used to classify CIFAR-10 images.

Model	Accuracy	Accuracy (400 per class)	max # of features units
1 Layer K-means	80.6%	63.7% (±0.7%)	4800
3 Layer K-means Learned RF	82.0%	70.7% (±0.7%)	3200
View Invariant K-means	81.9%	72.6% (±0.7%)	6400
Exemplar CNN	84.3%	77.4% (±0.2%)	1024
DCGAN (ours) + L2-SVM	82.8%	73.8% (±0.4%)	512

Additionally, since our DCGAN was never trained on CIFAR-10 this experiment also demonstrates the domain robustness of the learned features.

CLASSIFYING SVHN DIGITS USING GANS AS A FEATURE EXTRACTOR

Table 2: SVHN classification with 1000 labels

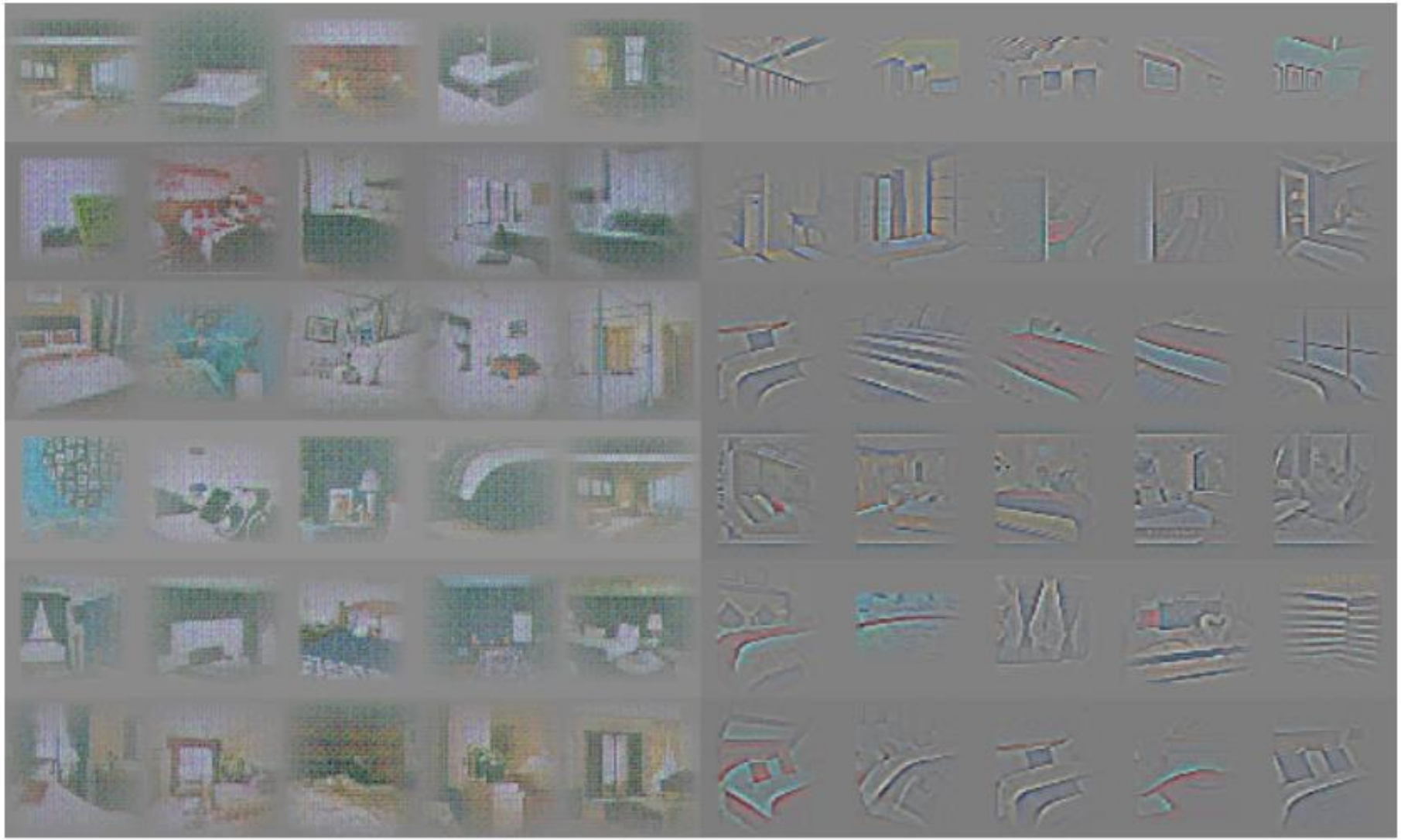
Model	error rate
KNN	77.93%
TSVM	66.55%
M1+KNN	65.63%
M1+TSVM	54.33%
M1+M2	36.02%
SWWAE without dropout	27.83%
SWWAE with dropout	23.56%
DCGAN (ours) + L2-SVM	22.48%
Supervised CNN with the same architecture	28.87% (validation)

WALKING IN THE LATENT SPACE (Generator的“连续性”)

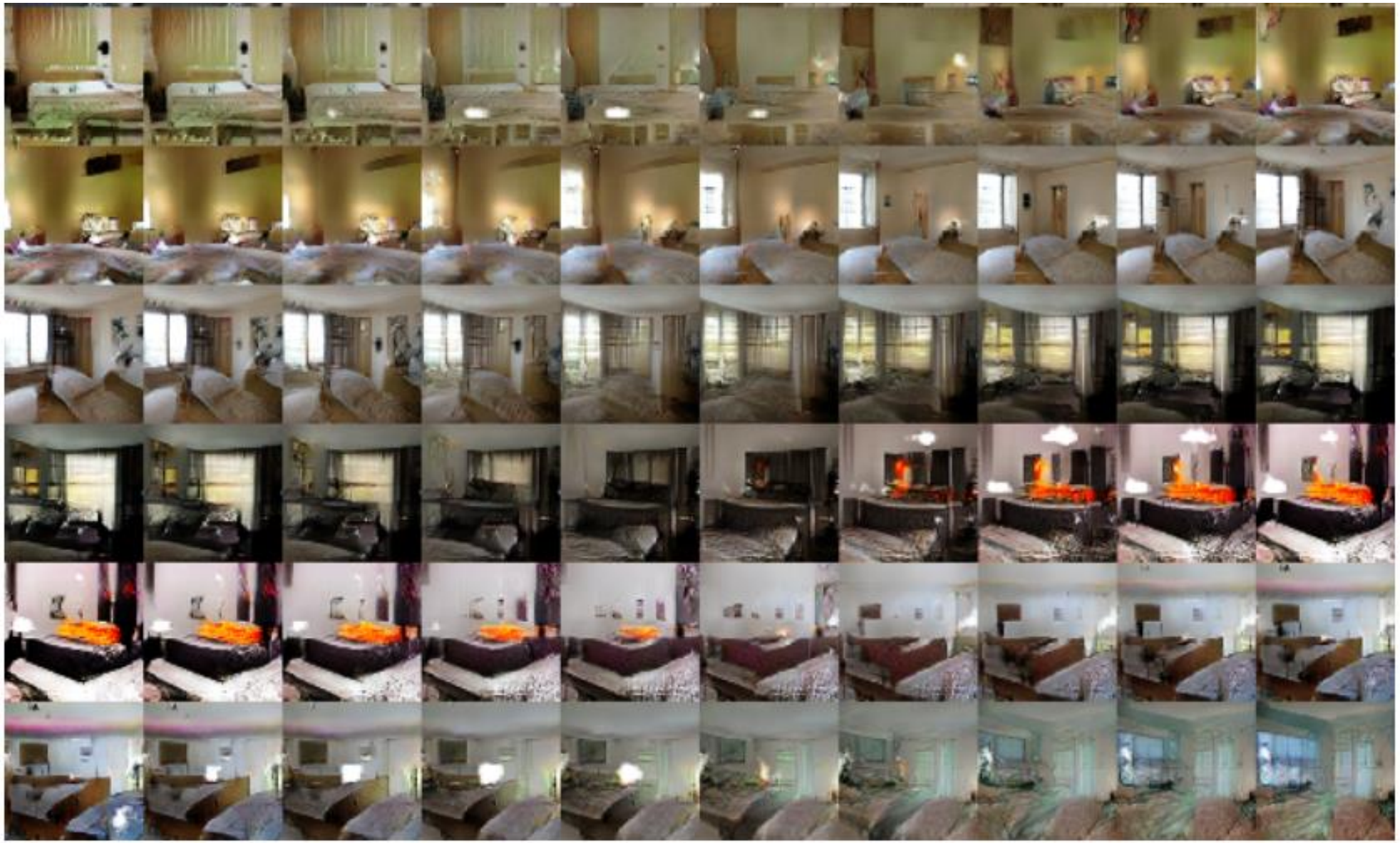
用DCGAN训练的generator能够保持从latent space到image space的“连续性”，这里用了“连续性”，并不是说满足严格的分析上的连续性，而是在latent space上做一点小变化，不会引起image space的大变化。相似latent vector通过generator会产生相似的图像。

作者在两个不同的latent vector之间插值得到新的vector，发现它们对应的图像具有一个平滑变换的过程

V
F



Random filters Trained filters



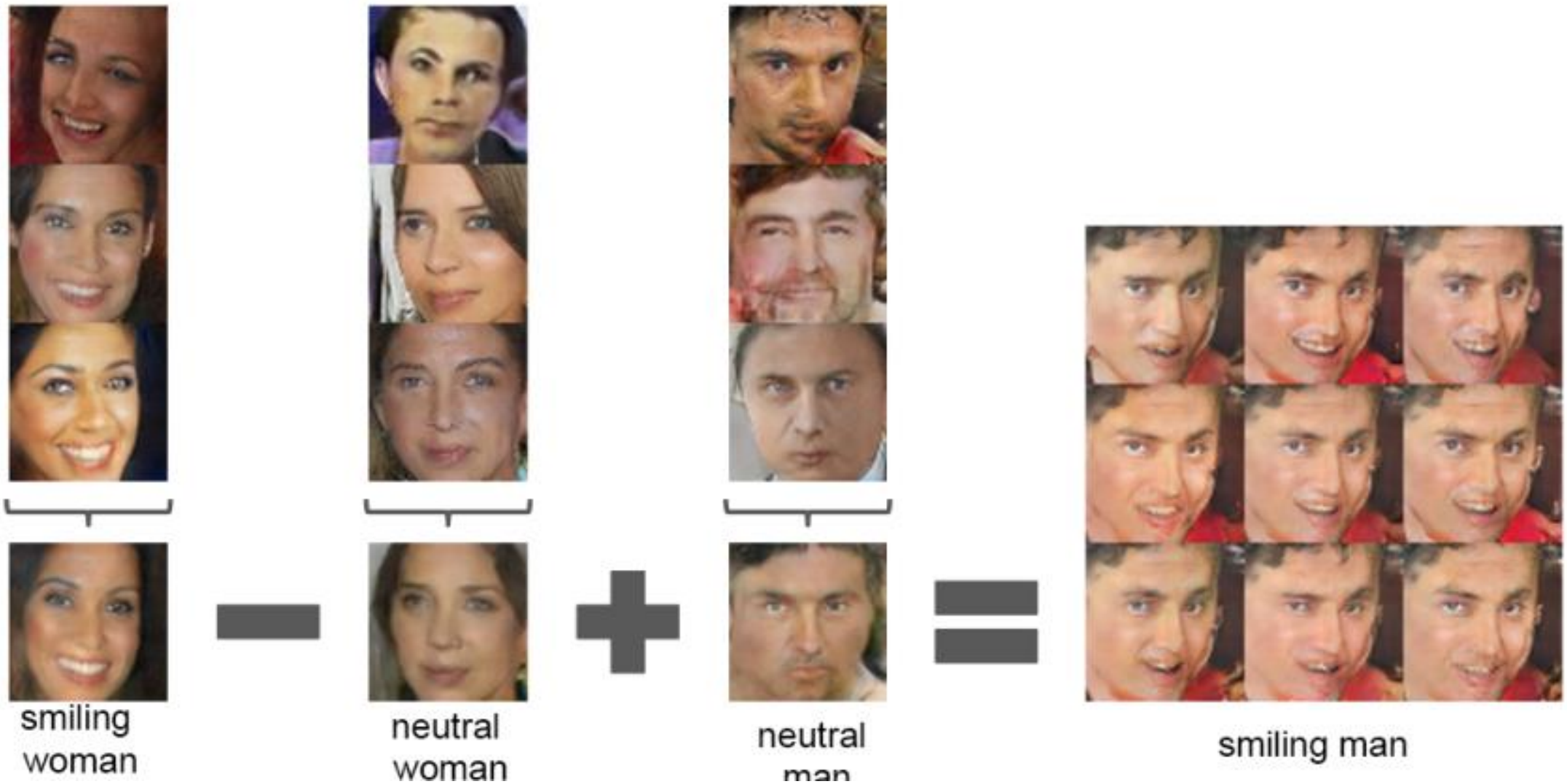
FORGETTING TO DRAW CERTAIN OBJECTS

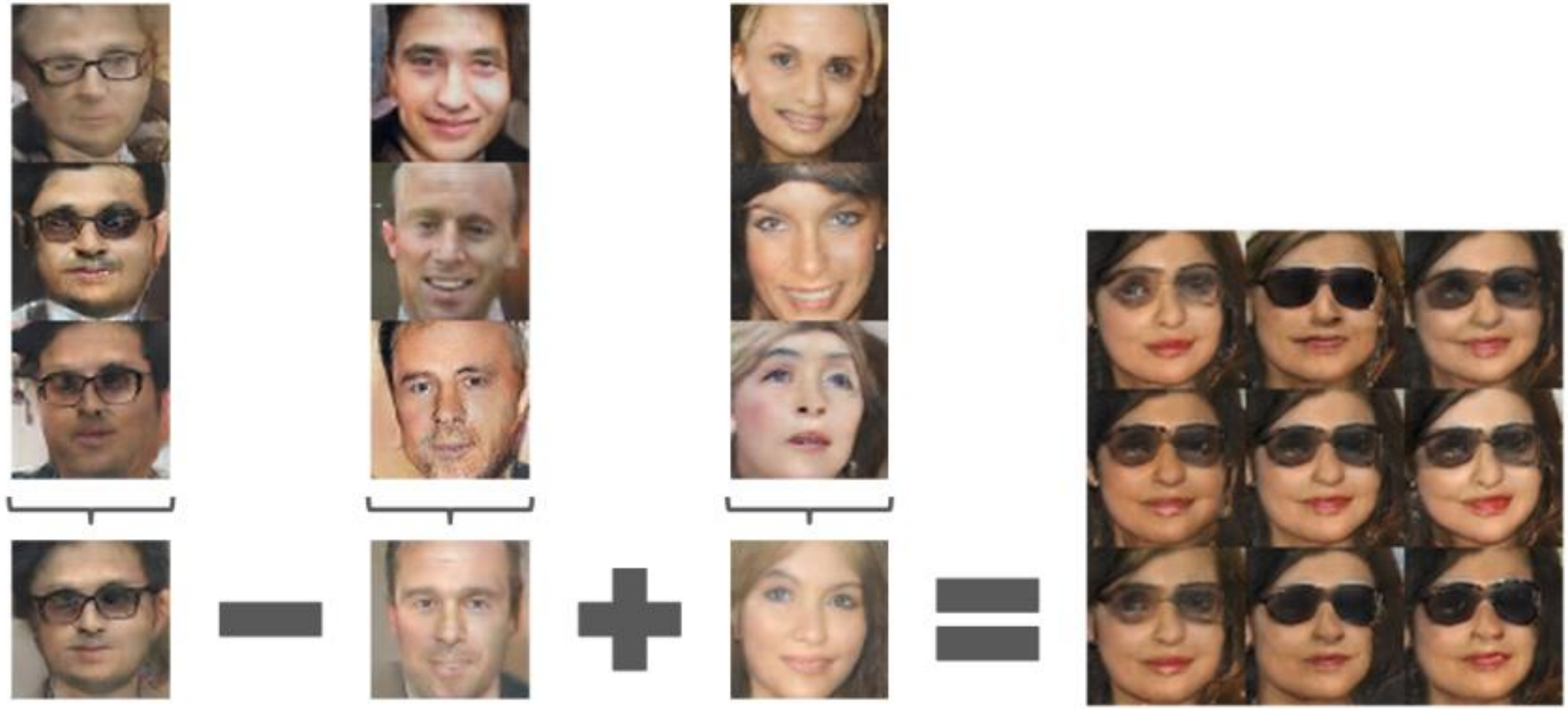
On 150 samples, 52 window bounding boxes were drawn manually. On the second highest convolution layer features, logistic regression was fit to predict whether a feature activation was on a window (or not), by using the criterion that activations inside the drawn bounding boxes are positives and random samples from the same images are negatives. Using this simple model, all feature maps with weights greater than zero (200 in total) were dropped from all spatial locations. Then,
random new samples were generated with and without the feature map removal.

FORGETTING TO DRAW CERTAIN OBJECTS

VECTOR ARITHMETIC ON FACE SAMPLES (Generator能保留“图向量”的语义信息）

One canonical example demonstrated that the vector(” King”) - vector(” Man”) + vector(” Woman”) resulted in a vector whose nearest neighbor was the vector for Queen. We investigated whether similar structure emerges in the Z representation of our generators. We performed similar arithmetic on the Z vectors of sets of exemplar samples for visual concepts. Experiments working on only single samples per concept were unstable, but averaging the Z vector for three examplars showed consistent and stable generations that semantically obeyed the arithmetic.


smiling woman
neutral woman
neutral man
smiling man


man
with glasses
man
without glasses
woman
without glasses
woman with glasses

好词好句

1. We use this approach in our generator, allowing it to learn its own spatial upsampling, and discriminator.

2. We scraped images containing human faces from random web image queries of peoples names

谢谢