

DCGAN Modeli ile Yüz Oluřturma

Omar GARIBOV

Dataset: <https://www.kaggle.com/datasets/gasgallo/faces-data-new>

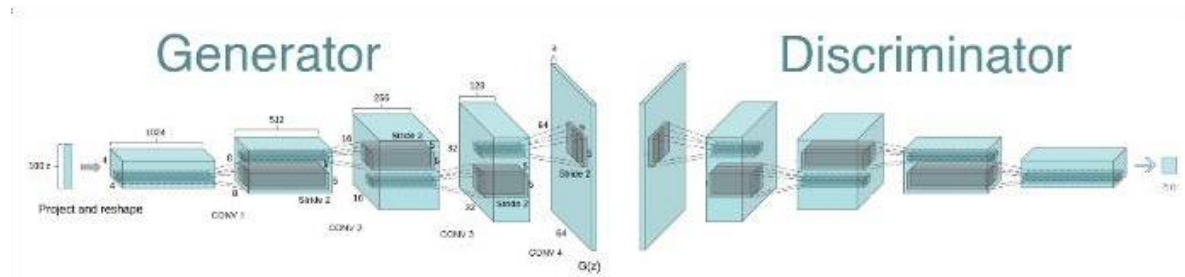
Colab: https://colab.research.google.com/drive/1o4xTFOYqReODTiCOL488ZAN-SoxLD_Gf?usp=sharing

İçindekiler Tablosu

Algoritma tanıtımı	3
Algoritmanın bulunan örneği	3
Colab linki	6
Çıktı	6
Referanslar	7

1. Algoritma tanıtımı

DCGAN (Deep convolutional generative adversarial networks) – Gözetimsiz öğrenmeyi hedefleyen ve bunu gan'daki üretici ve ayırmacı yapay sinir ağı model çifti arasındaki dinamik öğrenme etkileşimi neticesinde başaran, çok katmanlı mimariye ve konvolüsyonel işlemlere dayalı yapay öğrenme tekniği. Gözetimsiz öğrenme aşamasındaki anahtar nokta, üretici sinir ağını hedef sınıflandırma işlemi için klasik şekilde doğrudan eğitmek (hata oranını minimize etmeye çalışmak) yerine, çıktıdaki veri dağılımının mümkün olduğunca ayırmacı ağ modelini yanıltmaya çalışmasını sağlamaktır. Bu sayede her epoch'ta, üretici model hatalı dağılım üretmede (ayırmacı modeli kandırmada) daha başarılı hale gelirken, diğer model bu hataları tespit etmede uzman hale gelir. bu anlamda ayırmacı model shift-invariant bir cnn'dir. Mimaride, perceptron'lar için aktivasyon fonksiyonu olarak genelde relu kullanılmakla birlikte, üreticinin çıkış katmanı için hiperbolik tanjant fonksiyonu kullanılır (negatif değerlerin mapping'i için). Ayırmacı ağdaki katmanlarda ise giriş negatif değerler aldığıında, çıkışta doğrudan sıfır vermek yerine çok küçük negatif değerlere de izin verebilen ($y = \max(0, x)$ yerine, $y = 0.01 * x$, gibi...) ve bir relu varyasyonu olan leaky relu kullanılır. Bu sayede daha dengeli ve hızlı bir öğrenme aşaması hedeflenir.



2. Algoritmanın bulunan örneği

Bu örnekte birçok gerçek ünlünün resimleri gösterildikten sonra bu resimler kullanılarak yeni yüzler oluşturulmuştur.

Verilerin alınması

Bağlantılı siteden veya Google Drive'dan indirilebilen Celeb-A Faces veri seti kullanılmıştır. Veri kümesi, `img_align_celeba.zip` adlı bir dosya olarak indirilecektir. İndirdikten sonra `celeba` adlı bir dizin oluşturulup zip dosyası bu dizine çıkartılır. Oluşan dosya yapısı bu şekilde olacaktır:

```
/path/to/celeba
-> img_align_celeba
    -> 188242.jpg
    -> 173822.jpg
    -> 284702.jpg
    -> 537394.jpg
    ...
```

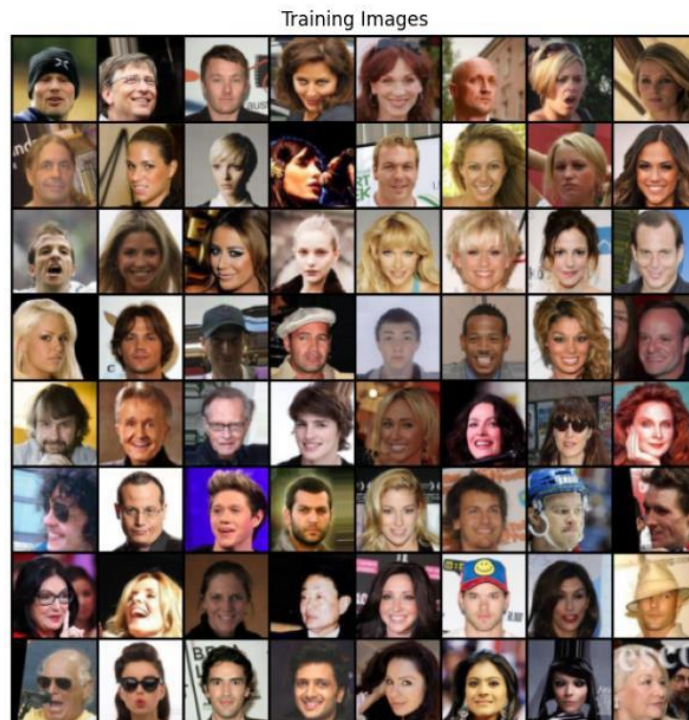
Veri kümesinin kök klasöründe alt dizinler olmasını gerektiren ImageFolder veri kümesi sınıfını kullanılacak. Artık dataset’i ve dataloader’ı oluşturabilir, cihazı çalışacak şekilde ayarlayabilir ve son olarak train verilerinin bir kısmını görselleştirebiliriz.

```
# We can use an image folder dataset the way we have it setup.
# Create the dataset
dataset = dset.ImageFolder(root=dataroot,
                           transform=transforms.Compose([
                               transforms.Resize(image_size),
                               transforms.CenterCrop(image_size),
                               transforms.ToTensor(),
                               transforms.Normalize((0.5, 0.5, 0.5), (0.5, 0.5,
0.5))),
                           )))

# Create the dataloader
dataloader = torch.utils.data.DataLoader(dataset, batch_size=batch_size,
                                         shuffle=True, num_workers=workers)

# Decide which device we want to run on
device = torch.device("cuda:0" if (torch.cuda.is_available() and ngpu > 0) else
"cpu")

# Plot some training images
real_batch = next(iter(dataloader))
plt.figure(figsize=(8,8))
plt.axis("off")
plt.title("Training Images")
plt.imshow(np.transpose(vutils.make_grid(real_batch[0].to(device)[:64], padding=2,
normalize=True).cpu(), (1,2,0))))
```



Girdi parametreleri setimiz ve hazırladığımız dataset ile artık uygulamaya geçebiliriz. GAN framework'ün tüm bölümleri tanımlandıktan sonra onu eğitebiliriz.

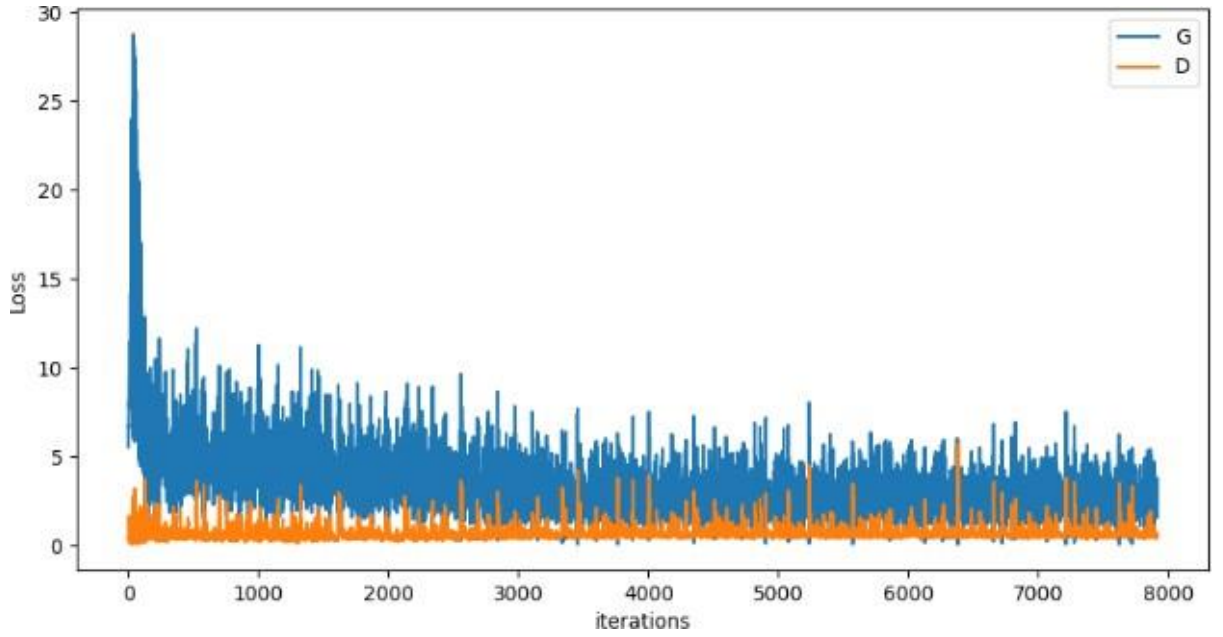
Discriminator'un eğitilmesi

Ayrımcıyı eğitmenin amacı, belirli bir girdiyi gerçek veya sahte olarak doğru bir şekilde sınıflandırma olasılığını en üst düzeye çıkarmaktır.

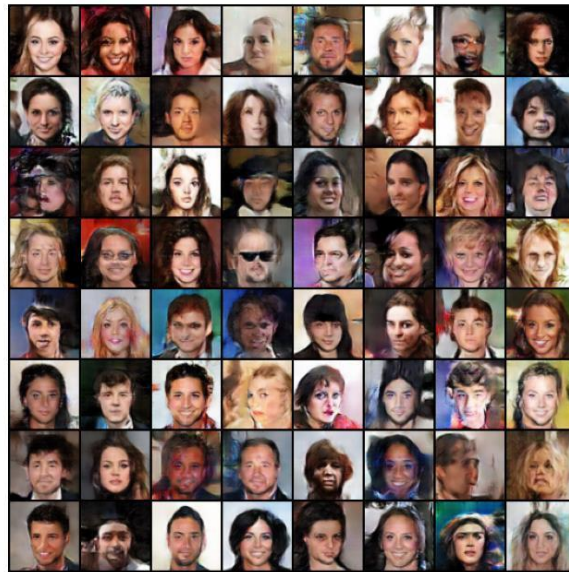
Generator'un eğitilmesi

Daha iyi yüzler oluşturmak için üreticiyi eğitmek gerekiyor.

Eğitim sırasında generator ve discriminator kaybı



Her eğitim adımından sonra generator'un çıktısı kaydedilir. Oluşturulan yüzler:

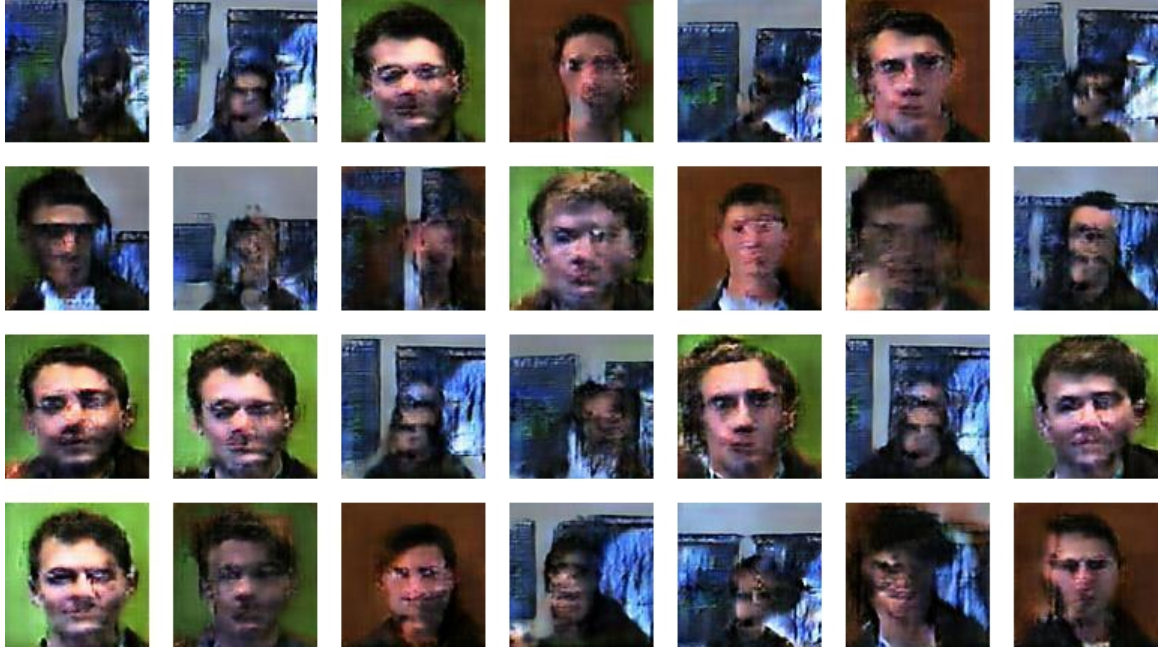


3. Colab linki

https://colab.research.google.com/drive/1o4xTFOYqReODTiCOL488ZAN-SoxLD_Gf?usp=sharing

Eklenen faces_data_new dataset'i için ađ tekrar alıřtırıldığında oluřturulan yzler.

ıktı:



Referanslar

GAN — DCGAN (Deep convolutional generative adversarial networks) by Jonathan Hui

Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks

DCGAN TUTORIAL Nathan Inkawhich