Techno Music Mel Spectrogram Generation with GANs

Christos Kyriazopoulos

MSc Artificial Intelligence

June 27, 2023

Contents

Current Section

1. Introduction
2. Methodology
3. Results
4. Conclusion

Requirements

- PyTorch
- NumPy
- Matplotlib
- Librosa
- Pytube
- Pydub

Current Section

Data Collection

- Main Source –>Youtube Playlists
- Over 3000 techno songs
- Mean Duration $\approx$ 5 min
- 5 second segmentation (without overlap)
- Transformation to Mel Spectograms –>

Final Dataset –> 260k Mel Spectograms

What are DCGANs

- Generator competes Discriminator
- Discriminator–> Conv Layers
- Generator–> Transposed Conv Layers
- No Pooling Layers
- Only Strided ConvLayers

DCGAN 1

```
---------------------
      Layer (type)
=====================
          Conv2d-1
       LeakyReLU-2
          Conv2d-3
     BatchNorm2d-4
       LeakyReLU-5
         Dropout-6
          Conv2d-7
     BatchNorm2d-8
       LeakyReLU-9
         Conv2d-10
    BatchNorm2d-11
      LeakyReLU-12
         Conv2d-13
        Sigmoid-14
=====================
```

Figure 1: Discriminator Structure

```
---------------------
      Layer (type)
=====================
 ConvTranspose2d-1
     BatchNorm2d-2
            ReLU-3
 ConvTranspose2d-4
     BatchNorm2d-5
            ReLU-6
 ConvTranspose2d-7
     BatchNorm2d-8
            ReLU-9
ConvTranspose2d-10
    BatchNorm2d-11
           ReLU-12
ConvTranspose2d-13
           Tanh-14
=====================
```

Figure 2: Generator Structure

```
------------------------
        Layer (type)
========================
            Conv2d-1
         LeakyReLU-2
            Conv2d-3
       BatchNorm2d-4
         LeakyReLU-5
           Dropout-6
            Conv2d-7
       BatchNorm2d-8
         LeakyReLU-9
          Dropout-10
           Conv2d-11
          Sigmoid-12
========================
```

Figure 3: Discriminator Structure

```
------------------------
        Layer (type)
========================
   ConvTranspose2d-1
       BatchNorm2d-2
              ReLU-3
   ConvTranspose2d-4
       BatchNorm2d-5
              ReLU-6
           Dropout-7
   ConvTranspose2d-8
       BatchNorm2d-9
             ReLU-10
  ConvTranspose2d-11
             Tanh-12
========================
```

Figure 4: Generator Structure

WGANs

What are WGANs

- Generator competes Discriminator

- Make use of Wasserstein Distance for Loss

$$W(\mathbb{P}_r, \mathbb{P}_\theta) = \sup_{\|f\|_L \leq 1} \mathbb{E}_{x \sim \mathbb{P}_r}[f(x)] - \mathbb{E}_{x \sim \mathbb{P}_\theta}[f(x)]$$

  - Discriminator Loss –>

$$\nabla_w \frac{1}{m} \sum_{i=1}^{m} \left[f(x^{(i)}) - f(G(z^{(i)})) \right]$$

  - Generator Loss –>

$$\nabla_\theta \frac{1}{m} \sum_{i=1}^{m} f(G(z^{(i)}))$$

- 2 Implementations –>
  - with weight clipping –> more unstable
  - with gradient penalty –> more stable

WGAN

```
Layer (type)
====================
Conv2d-1
LeakyReLU-2
Conv2d-3
InstanceNorm2d-4
LeakyReLU-5
Dropout-6
Conv2d-7
InstanceNorm2d-8
LeakyReLU-9
Dropout-10
Conv2d-11
====================
```

Figure 5: Discriminator Structure

```
Layer (type)
====================
ConvTranspose2d-1
BatchNorm2d-2
ReLU-3
ConvTranspose2d-4
BatchNorm2d-5
ReLU-6
Dropout-7
ConvTranspose2d-8
BatchNorm2d-9
ReLU-10
ConvTranspose2d-11
Tanh-12
====================
```

Figure 6: Generator Structure

Training Models

- DCGAN 1 –>
  - Trained for 7 epochs
    1. 260k samples used
- DCGAN2 –>
  - Trained for 2 epochs
    2. 260k samples used
  - Trained for 6 epochs
    3. 60k samples used
  - Trained for 7 epochs
    4. 30k samples used
- WGAN –>
  - Trained for 5 epochs (weight clipping)
    5. 60k samples used
  - Trained for 5 epochs (gradient penalty)
    6. 60k samples used

Current Section

Generated Mel Spectrogram Examples 1

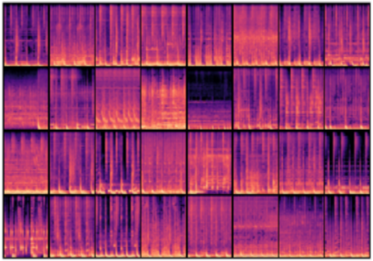

Figure 7: DCGAN1 Generated 260k 7ep

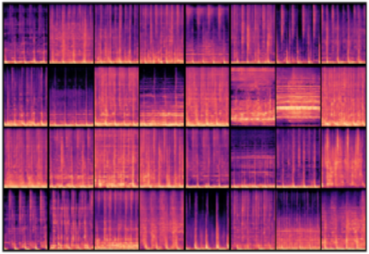

Figure 8: Training Examples

Generated Mel Spectrogram Examples 2

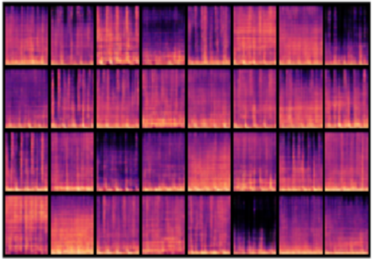

Figure 9: DCGAN2 Generated 60k 6ep

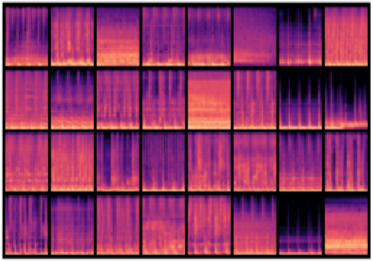

Figure 10: Training Examples

Generated Mel Spectrogram Examples 3

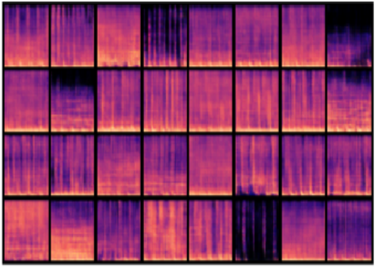

Figure 11: DCGAN2 Generated 260k 2ep

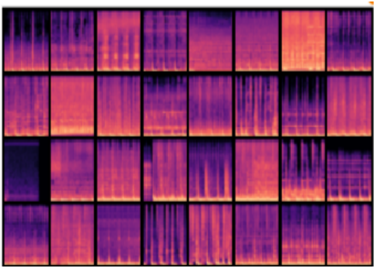

Figure 12: Training Examples

Generated Mel Spectrogram Examples 4

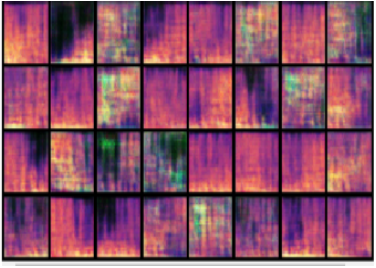

Figure 13: WCGAN wc Generated 60k 5ep

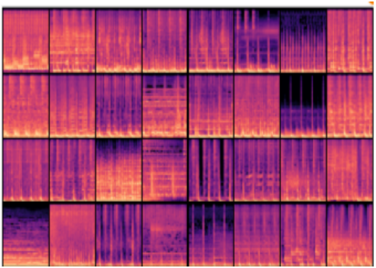

Figure 14: Training Examples

Generated Mel Spectrogram Examples 5

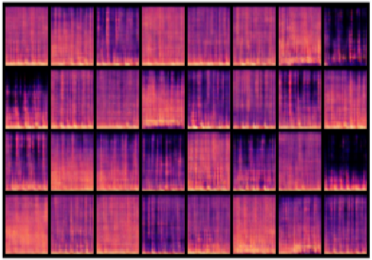

Figure 15: WCGAN gp Generated 60k 5ep

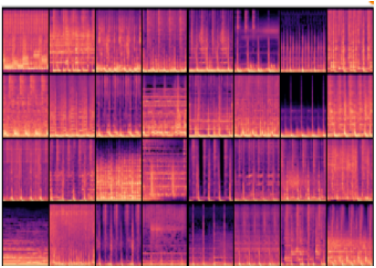

Figure 16: Training Examples

Losses Visualization 1

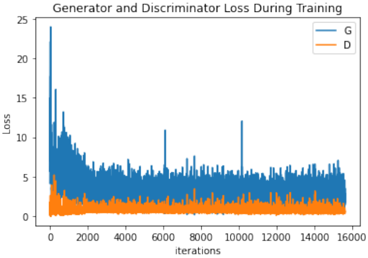

Figure 17: DCGAN1 260k 7ep

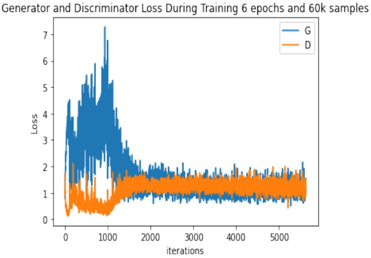

Figure 18: DCGAN2 60k 6ep

Losses Visualization 2

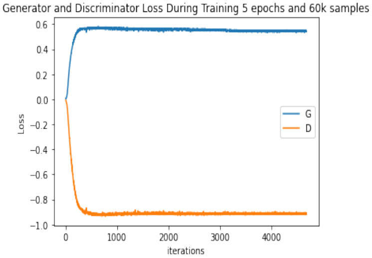

Figure 19: WGAN wc 60k 5ep

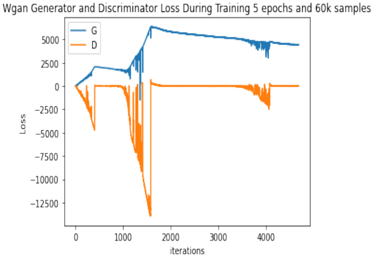

Figure 20: WGAN gp 60k 5ep

Current Section

Summary & Future Work

- Models
  - Best Models (empirically) —>

    1. DCGAN1 260k 7ep
    2. DCGAN2 60k 6ep

  - WGAN GP —> Need more training

  - WGAN WC —> Diverges after 4ep
- Future Work
  - Inverse Generated Mel Spect. to Audio
  - Metrics Calculation (Inception Distance (FID) Inception Score (IS))
  - Train WGAN GP for more epochs
  - Use Pre trained autoencoders to enhance resolution

References

- Arjovsky M., Chintala S. and Bottou L. (2017) Wasserstein Gan. Available at: https://arxiv.org/abs/1701.07875.
- Improved training of Wasserstein Gans. Available at: https://arxiv.org/pdf/1704.00028v3.pdf.
- Radford, A., Metz, L. and Chintala, S. (2016) Unsupervised representation learning with deep convolutional generative Adversarial Networks. Available at: https://arxiv.org/abs/1511.06434.

Thank You