# Neural Style Transfer on Audio Signals

***Abstract*—Neural Style Transfer on images has been a vastly researched topic and has been used to generate novel arts. In this project we attempt to extend the concept of Neural style transfer on audio signals.**

## I. Objective

Perform Neural Style Transfer on Audio signals and attempt to transfer the style of Classical music to Jazz music.

## II. Theory

### A. Neural Style Transfer on Images

In Neural style transfer on images, we attempt to transfer the style of a Style Image to a Content Image. A content image is the image on which the style has to be transferred. A style image is the image whose style we want to transfer to the content image. This process results in a final Generated image which will be a blend of content and style image. In this project we initialize our Generated image to be the Content Image added with Gaussian Noise. We use a pre-trained neural network to generate latent representation of the images and calculate loss by the loss function as described below. Using this loss function and gradient descent we update each pixel minimizing our loss adopting the style of the Style image.

### B. Loss Function for Style Transfer on Images

- **Content Loss:** The content loss is the mean squared error between the encoding of the white noise image and the content image. For a layer $l$ and the input image $\vec{x}$, let the number of filters be $N_l$ . The output (or encoded) image will have $N_l$ feature maps, each of size $M_l$ , where $M_l$ is the height times width. So, the encoded image of layer can be stored in a matrix $F_l$.
  Where $F^l_{ij}$ is the activation of $i^{th}$ filter at position $j$ in layer $l$.

$$\mathcal{L}_{content}(\vec{p},\vec{x},l) = \frac{1}{2}\sum_{i,j}(F^l_{ij}\text{-}P^l_{ij})^2$$

- **Style Loss:** For capturing the style of an artist, a style representation is used. It computes the correlations between the different filter responses, where the expectation is taken over the spatial extent of the input image. These feature correlations are given by Gram Matrix $G^l$, where $G^l_{ij}$ is the inner product between the feature maps i and $j$ represented by vectors in layer $l$ and $N_l$ is the number of feature maps.

$$G^l_{ij} = \sum_k F^l_{ik}F^l_{jk}$$

  The style loss is the mean squared error between the gram matrices of style image and the white noise image. Let $\vec{a}$ be the style image and $\vec{x}$ be the white noise image. Let A‘ and X‘ be the style representations of style image and white noise image in layer $l$. So, total style loss of a layer $l$ is $E_l$.

$$E_l = \frac{\sum_{ij}(X^l_{ij}-A^l_{ij})^2}{4N_l^2M_l^2}$$

  The total style is:

$$L_{style}(\vec{a},\vec{x}) = \sum_{l=0}^{L} w_l E_l$$

  where $w_l$ is the weighting factor of each layer.

- **Total Loss:** Let $\vec{p}$ be the content image, $\vec{a}$ be the style image, and $\vec{x}$ be the white noise image (i.e., the generated image) that will constitute the final image. Total loss is the sum of content loss and style loss.

$$\mathcal{L}_{total}(\vec{p},\vec{a},\vec{x}) = \alpha\mathcal{L}_{content}(\vec{p},\vec{x}) + \beta\mathcal{L}_{style}(\vec{a},\vec{x})$$

### C. Neural Style Transfer on Audio Signals

For neural style transfer on images we will be first taking our audio signals to a visual representation and applying the concepts of neural style transfer on images to the visual representation of the audio signals. For converting our audio signal to visual representation we use mel-spectogram.

### D. Mel Spectogram

Mel Spectogram is representation of strength of a signal over time at a waveform's different frequencies on a logarithmic scale. Though, humans can detect differences in lower frequencies, but we can't distinguish between differences in higher frequencies. Hence, Mel Spectogram does this representation of frequencies on a logarithmic scale. Representation of audio signals to Mel Spectogram follows a three step approach-

*1) Short Time Fourier Transformation:* Short time Fourier Transformation or STFTs are applied on time varying audio signals. It is basically decomposition of signal into its constituent frequencies and displaying each frequency in terms of its amplitude. The normal spectogram is chopped in segments and then Fourier transformations are applied.

*2) Amplitude in Decibels:* Amplitude is a measure of loudness and loudness is also perceived on a logarithmic scale, hence, amplitudes are converted to Decibels(DBs).

*3) Converting frequencies to Mel Scale:* Lastly the frequencies are converted to Mel Scale using the formula-

$$m = 2595log_{10}(1 + \frac{f}{500})$$

where,

m = Mel Scale value of frequency
f = frequency in Hertz

## III. Procedure

### A. Converting Audio signals to Melspectogram

We start with loading our audio file one of Jazz music(Content audio) and one of Classical music(Style audio) and initialize our Generated audio with the Content audio only. Next we convert our audio signals to Mel Spectograms using librosa library functions.

### B. Model selection for generating latent representation

Here in our case we can not use the same model used for generating latent representation in Neural style transfer for images because of two reasons. First of all that model is trained on images and hence its latent representation would not be appropriate for mel spectograms of audio signals. Second images have high correlation among the pixels present along its height as well as width, but in case of audio signals we have high correlation along time axis which translates to correlation among pixels along width of the image. For this purpose we use a two layer 1D convolutional neural network, shown in the following figure.

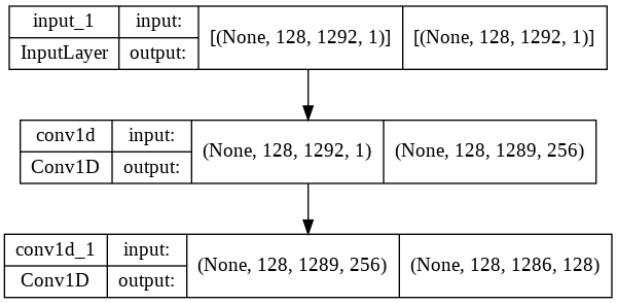


Fig. 1. 1D Convolutional Model for Audio signals

### C. Calculation of Content and Style loss on Generated image

We pass our mel spectograms as input to the model and use the outputs of different layer to calculate the style and content loss the same way we calculate in the case of Neural style transfer of images.

### D. Gradient descent for minimizing loss

Finally, using gradient descent we calculate the gradient of the loss function with respect to each pixel of the Mel Spectogram and update each pixel to minimize our loss over each iteration.

## IV. Results

With a learning rate of 0.01 and 20,000 iteration the loss was reduced to 0.0034 and the Mel spectograms given below show that the above technique was able to effectively transfer the style of classical music to jazz music. However when we converted the Mel spectogram back to audio signals the resulting audio signals had a lot of noise in it.

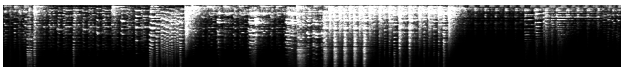

Fig. 2. Mel Spectogram of Jazz audio(Content Audio)

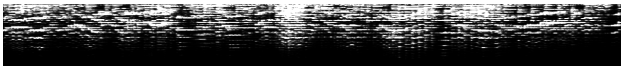

Fig. 3. Mel Spectogram of Classical audio(Content Audio)

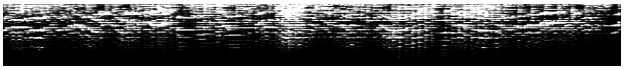

Fig. 4. Mel Spectogram of Generated audio

## V. Conclusion

On performing Neural style transfer on audio signals the final audio had a lot of noise as audio signals are very susceptible to noise. More over we need to work on with the calculation of style loss as we use the style loss for image in audio signals which might not be able to effectively capture the inherent style of the audio signals.

## VI. Project Link

Link for the code used for above project can be found **here**