2019-08-22

# Sound Transformation: Applying Image Neural Style Transfer Networks to Audio Spectrograms

Xuehao Liu
*Technological University Dublin*, xuehao.liu@tudublin.ie

Susan McKeever
*Technological University Dublin*, susan.mckeever@tudublin.ie

Sarah Jane Delany
*Technological University Dublin*, sarahjane.delany@tudublin.ie

## Recommended Citation

Liu, X., Delany, S.J. & McKeever, S. (2019) Sound Transformation: Applying Image Neural Style Transfer Networks to Audio Spectrograms, *CAIP 2019, International Conference on Computer Analysis of Images and Patterns*, Salerno, 2-6 September, 2019.

# Sound transformation: Applying Image Neural Style Transfer Networks to Audio Spectograms

Xuehao Liu, Sarah Jane Delany, and Susan McKeever

Technological University Dublin, Dublin, Ireland
xuehao.liu@mydit.ie,{sarahjane.delany,susan.mckeever}@dit.ie

**Abstract.** Image style transfer networks are used to blend images, producing images that are a mix of source images. The process is based on controlled extraction of style and content aspects of images, using pre-trained Convolutional Neural Networks (CNNs). Our interest lies in adopting these image style transfer networks for the purpose of transforming sounds. Audio signals can be presented as grey-scale images of audio spectrograms. The purpose of our work is to investigate whether audio spectrogram inputs can be used with image neural transfer networks to produce new sounds. Using musical instrument sounds as source sounds, we apply and compare three existing image neural style transfer networks for the task of sound mixing. Our evaluation shows that all three networks are successful in producing consistent, new sounds based on the two source sounds. We use classification models to demonstrate that the new audio signals are consistent and distinguishable from the source instrument sounds. We further apply t-SNE cluster visualisation to visualise the feature maps of the new sounds and original source sounds, confirming that they form different sound groups from the source sounds. Our work paves the way to using CNNs for creative and targeted production of new sounds from source sounds, with specified source qualities, including pitch and timbre.

**Keywords:** Audio Morphing · Neural Network · Image Style Transfer · Generative Adversarial Network

## 1 INTRODUCTION

With the success of deep learning techniques for image classification [15], researchers have continued to achieve improved classification rates, with image classifiers now outperforming the ability of humans to recognise images. For example, the combination of Res-Net and an Inception V3 Convolutional Neural Network (CNN) can classify images with a 96.91% success rate [22]. Until recently, CNNs have been treated as a black box, with a limited understanding of how images are represented at each layer of the CNN. Gatys at al. [6] addressed this by examining how specific images features are captured at particular layers of the CNN. They used this knowledge to generate images that mix the content and style of two source images. This image generation process, adopting the style or texture of one image and the content or contour of another image is termed *image style transfer*. Gatys et al. noted that higher layers of the CNN preserve the spatial structure or content in comparison to the capture at lower layers of image textures or style qualities in the image [13].

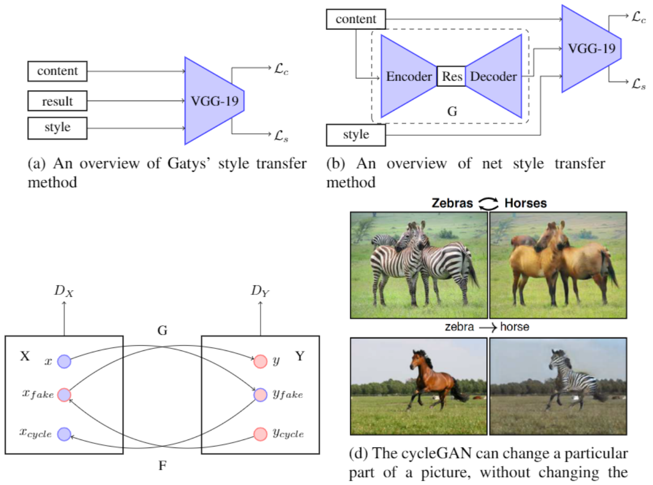
The purpose of this paper is to investigate image style transfer using neural networks for blending of sounds. An audio signal can be represented as a spectrogram, preserving audio frequency and amplitude. Instead of visual image inputs, audio spectrograms are fed into the image transfer process, producing a blended spectogram output. An early investigation of style transfer using spectrograms has been done by [23]. This work used AlexNet [15], a relatively shallow CNN, to extract feature maps and generate two illustrative audio spectrograms using the style transfer method. In addition to the limited number of outputs, there is no evaluation of the resultant spectrograms and thus limited insight into the nature of the generated sounds. In our work, we extend the work of [23], investigating the application of three image neural style transfer techniques [6, 13, 24] to the task of blending audio spectrogram inputs. We use classification models to demonstrate that the generated mixed sounds are new sounds distinguishable from the source sounds.

*Audio morphing* is a closely related field to our work as it focuses on the synthesis of audio signals. Audio morphing aims to find a middle ground between two audio signals, which share the properties from both sides[21]. Image neural style transform, when applied to audio, enables transformation of audio source sounds to produce a new sound, akin to audio morphing.

## 2 RELATED WORK

The traditional approach to audio morphing [21] is to match pitch and temporal components between two audio signals. Another approach is to deduce the sinusoids of one audio signals and fill them with magnitude from another sound's sinusoids [19]. More recently, the methods for determining a mix of two sounds have become more complex. Different kinds of spectral envelopes can be applied on the morphing process[1]. A common factor of these methods is that, unlike neural network style transfer, they require manual feature extraction from the audio signals.

Recent research works have applied CNNs to audio processing tasks. For example, Dieleman [2] tested audio classification models, using CNNS trained on raw audio file input and the corresponding audio spectrogram input. Spectrograms were found to have a slightly higher prediction accuracy than the raw audio, suggesting that spectrograms are a richer information source. Han et al [8] used a CNN (termed ConvNet) to classify musical instrument sounds. Hersehey et al. [11] used GoogleNet and Res-Net to do a similar classification task, but using a much larger video dataset, YouTube-100M. Their results with a 0.930 AUC demonstrated that good classification accuracies can be achieved using spectogram representation of audio signals. These works indicate that audio spectograms are a useful and valid representation of audio inputs with neural networks. In neural style transfer of audio, feature extraction will be done automatically by the network. Our neural style transfer work in this paper is inspired by image style transfer networks. In such networks, specific layers from the CNN are associated with *content* (objects) versus *style* (texture). Gatys et al.[6] demonstrated this by reconstructing an image, preserving its content, but changing the texture of the image to the style of Van Gogh's Starry Night. Johnson et al.[13] produced a faster version of Gatys et al.'s network, reducing time for one image blend from hundreds of seconds to less than one second. Frigo et al.[5] proposed a new style transfer method basing on Johnson et al.'s work, splitting the content and style images into small grids (adaptive quadtrees) and doing the style transfer operations on those similar small parts from the content and style images. Our work is also inspired by *image translation networks* which focus on translating just a specific portion of the image. For example, Isola et al. used conditional GANs to translate street maps to satellite maps. [12] In the work of Zhu et al.[24], they transform style on a portion of the image content using a cyclical generative adversarial network termed cycleGAN. An example of their work is the transformation of a horse in the image to a zebra, without changing the background of the content image, as shown in Figure 1(d) Next, we provide a more detailed explanation of three image style transfer networks that represent a good coverage of the range of networks available and which will feature in our approach:



(a) An overview of Gatys' style transfer method

(b) An overview of net style transfer method

(c) An overview of cycleGAN style transfer method

(d) The cycleGAN can change a particular part of a picture, without changing the remaining area of that picture. These two examples are from [24]

Fig. 1: An overview of three kinds of structures used in this paper, and a GAN example.

*Gatys et al.* [6] image neural style transfer results are used as the baseline by other image style transfer methods [13, 5]. The purpose of their network, as shown in Figure 1(a), is to produce a blended image which consists of the content of one image and the style of another image. Looking at their configuration, the two sources images (one for content, one for style) are fed into the VGG-19 CNN[20]. The CNN extracts the content and style features, generating a blended result image. The result image is initialised as random noise, and on each training iteration the result image is adjusted to minimise the sum of content and style loss comparing to the result image.

*Johnson et al.* [13] proposed a method of image neural style transfer that performs significantly faster than that of Gatys et al. Their method shown in Figure 1(b) produces a blended result image produced from a sample image that will provide the content, but the style is pre-defined. The underlying concept is that style can be pre-captured, whereas content from image to image will vary. Instead of initialising the input as random noise they pass the output of a generative network $G$, into the pre-trained VGG-19. The input to the generative network $G$ is the content image. After each result iteration, the loss in respect to both the fixed style image and the content image is compared and gradient descent is performed on the generative network $G$. The advantage of their method is that style transfer can be applied at close to real-time processing speed.

*Zhu et al.* [24] created a cyclical generative adversarial network, termed *cycleGAN*, for style transfer. The target task in their work is to translate a specific portion of the image, rather than transforming the style of the full image. In Figure 1(d), a sample from their network demonstrates how a horse in an image is transferred to a zebra. The structure of cycleGAN is shown in Figure 1(c). There are two classes $X$ and $Y$ in the dataset where the style of X can be transformed to Y. A generative network $G$ generates a candidate $y_{fake}$ from each $x$ image and a discriminate network $D_Y$ will evaluate whether it is a real $Y$ image. A generative network $F$ then generates $x_{cycle}$ from $y_{fake}$ and uses the discriminate network $D_X$ to evaluate whether it is a real $X$ image.

Applying this to Zhu et al's [24] example in Figure 1(d), classes $X$ and $Y$ are zebra and horse respectively. The output in the figure are $x_{fake}$ and $y_{fake}$. The training process of $G$ and $F$ is to build the mapping between $X$(zebra) and $Y$(horse). $x_{fake}$(fake zebra) may not exist, but we can minimize the distance between $x_{cycle}$ and $x$ to make a more realistic $y_{fake}$(fake horse). With a perfect $G$ and a perfect $F$, $x_{cycle}$ should equal to $x$, and $y_{cycle}$ should equal to $y$. The horse should be the exact same one after changing it to a zebra and changing it back to horse. $x_{fake}$(fake zebra) and $y_{fake}$(fake horse) will be the transfer of $X$ and $Y$ we want.

To evaluate our results, we will need to determine whether coherent consistent sounds are being produced. According to [9], if we describe multiple audio signals as being the same kind of sound, their spectrograms are identical from the point view of timbre. As previously explained, CNNs can be used to successfully classify instruments basing on their timbre[8, 11]. Classification has also been used in the image generation domain as an approach to verifying results of image generation[12]. Given these approaches, we will use classification methods(using CNNs) to test whether our generated sounds can be distinguished from the source sounds, and to test the timbre consistency of generated sounds.

# 3 APPROACH

Our approach used for audio style transfer follows the approaches used for image style transfer in other works [23, 11, 13, 24]. The first step in image style transfer involves training a CNN on labelled images - producing a network with embeded content and style feature maps at known layers in the CNN. Applying this approach to audio, we train a CNN classification network with labelled audio spectogram inputs of the types of source sounds we will later aim to blend. The second step in image style transfer is the blending process itself - using the trained CNN and the relevant image style transfer technique with source inputs to produce a blended result output. For audio neural style transfer, the input to the trained network will be the spectogram representation of the two audio signals to be mixed. Figure2(a) and (b) are two spectrogram examples of input to the CNN. Using the layers of this CNN, we will use three methods of image style transfer to the audio following the baseline transfer method of Gatys et al., the faster transfer method of Johnson et al. and Zhu et al.'s cycleGAN transfer method, as described in the Related Work section.

## 3.1 Datasets

We use the Nsynth corpus[4] which is a high-quality, large-scale musical instrument audio corpus. Every instance in the corpus is a 4 seconds long, 16kHz audio snippet that covers the whole sound envelope. For our work, we will mix two different musical instrument sounds. We extract a subset dataset from the Nsynth corpus consisting of the flute and keyboard classes: 8000 clips randomly selected from acoustic keyboard sounds and 8000 flute clips. The flute clips include all the acoustic flute and a small amount of synthetic flute.

## 3.2 Training the Classification Network

We train the classification network to distinguish between the keyboard and flute sounds. Our CNN follows the structure of VGG-19 [20]. VGG-19 does not have any shortcuts or concatenation of feature maps and it has many layers at different feature levels. Thus it provides a rich source of image feature knowledge and is used in a variety of image style transfer networks [6, 13, 7]. We call our classification network audio-VGG. The only difference between our audio classification network and the original VGG-19 is the first layer. Since we use spectrograms as input, and there is only one channel in the spectrogram (the absolute value of the Short Time Fourier Transfer) the spectrogram will be represented as a grey-scale image. For the training process, 1000 clips from each class were used as the holdout set. The remaining 14000 clips (7000 for each class) were used for a 7-fold cross-validation. The accuracy achieved on the test set was 99.99% with a five epoch training process and a 0.0001 learning rate. This shows that the network can fully distinguish the two instrument sound types, with the internal layers capturing the features of flute and keyboard. Next, we apply three separate transfer networks to blend keyboard and flute sounds.

### 3.3 Using the Neural Style Transfer Networks for Sound Mixing

*Baseline Slow Transfer*: Our first transfer process uses Gatys et al.'s [6] network. The aim is to mix two source sounds, treating the flute spectrogram as the content image and the keyboard spectrogram as the style image. The audio spectrograms of the flute and keyboard sounds are passed as input into our pretrained audio-VGG network. The target blended sound spectrogram is initialised as random Gaussian noise. The training process uses gradient descent on the Gaussian noise based on total loss$\mathcal{L}$, where total loss$\mathcal{L}$ is calculated by comparing the outputs from different layers of audio-VGG network. On completion of the gradient descent, the Gaussian noise input spectrogram has been transformed to a mix of the visual content and style of our source sound and keyboard spectograms. The transfer process is slow, taking several hours to generate a single mixed sound even using the modern GPUs(GTX 1080). The content loss is between two outputs of layers `relu3_3` in that audio-VGG. In the style loss $\mathcal{L}_s$, the Gram matrix loss $\mathcal{L}_g$, energy loss $\mathcal{L}_e$, and frequency loss $\mathcal{L}_f$ are balanced[23]. The style loss is computed on layer `relu1_2, relu2_2, relu3_3` and `relu4_3`. The interpolation factor $\lambda$ is equal to 1e-2.

*Faster Fixed Style Transfer*: The second method of style transfer used is Johnson et al.'s [13] faster fixed-style and generative network approach to speed up the transfer process. With this transfer process, we are generating multiple flute snippets which have the visual spectrogram style feature of a single keyboard snippet. The structure of the network [13] starts with a three-layer encoder and ends with another three-layer decoder, connected by residual blocks. We trained the generative net $G$ using a learning rate of 0.0002 for 10 epochs. The input of the generative network is the content spectrogram (a spectrogram of a flute snippet) and the output is the spectrogram of the mixed or blended sound. The mixed spectrogram, the content spectrogram (flute), and the style spectrogram (keyboard) will be passed into audio-VGG in the same way of the baseline slow transfer process. The difference is that the gradient descent will be done on the generative network. The style spectrogram is a keyboard clip. The content spectrograms are the same 8000 flute clips in the training VGG-19 process.

*CycleGAN Transfer*: Our third style transfer approach uses the structure and parameters of Zhu et al's discriminate and generative networks, more fully described in [24]. $X$ is flute. Two new sounds are generated: The $x_{fake}$ result is the flute with a keyboard content; $y_{fake}$ is the keyboard with a flute content. The generative networks $G$ and $F$ have a similar structure to that used in Faster Fixed Style transfer. Instead of padding before the residual blocks in Johnson et al's work[13], we followed the structure in [24], which does the padding between every layer in the residual blocks. The discriminators $D_X$ and $D_Y$ are two Markovian discriminators(PatchGAN), which is also used in [12]. This Markovian discriminator randomly chops the input image into a smaller 70×70 patch. According to [12], this smaller patch of input will be sufficient for the network to discriminate fake images from real ones. Also it is faster and has a smaller number of parameters. We trained the networks for 10 epochs with a 0.0001 learning rate using the same audio clips above(flute(X) and keyboard(Y)).

# 4 RESULTS

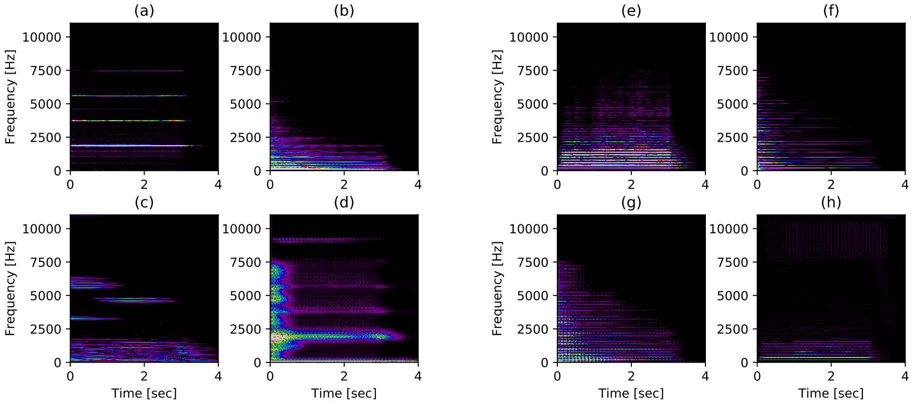


Fig. 2: (a)-(d): The spectrogram examples of slow transfer and faster fixed style transfer from the same source flute and keyboard. (a) the source flute, (b) the source keyboard, (c) the output spectrogram of slow transfer, (d) The output spectrogram of Faster Fixed Style transfer. (e)-(h): Spectrogram examples of the cycleGAN transfer. (e) the original flute $x$, (f) the original keyboard $y$, (g) the $x_{fake}$ output, (h): the $y_{fake}$ output.

The resulting audio clips are are published online[1]. Listening to the resulting sounds, we can hear that they are different to our source sounds. However, we need objective mechanisms to determine whether the resulting sounds are consistent with each other, and distinctly separate from the source sounds. Following previous work on processing and evaluation of audio signals [14, 8, 17, 18], we interpret our generated sound results using three methods: (1) Visual assessment of the generated audio spectrograms (2) Consistency tests on the generated sounds, using classification models and (3) Examination of the audio signal clusters.

## 4.1 Visual Assessment of Generated Spectrograms

Figure 2(a)-(d) shows two sample spectrograms of slow transfer and faster fixed style transfer. We observe that they both have the contour of the flute sound. With the baseline slow transfer, the harmonic is not clear, and the part higher than 6000Hz is discarded. In the faster transfer, the lower frequency is emphasized, and there is an onset(beginning) of the note, which is missing in the

[1] https://www.xuehaoliu.com/audio-show

slow transfer. Although the loss functions are the same, how these two methods learned from the keyboard are entirely different. With the slow transfer method, the generation of mixed spectrograms is initialised from Gaussian noise, whereas in faster fixed style transfer, the transfer has to follow the shape of content.

Figure 2(e)-(h) shows a mixed spectrograms resulting from the cycleGAN transfer. After the transfer, from flute to keyboard, the lower frequencies of the flute are emphasised, and the higher frequencies disappear. From keyboard to flute, the frequencies are denser. These two sample spectrograms show that this transfer will not change the harmonic, but the magnitude of each frequency is changed. This is similar to the model proposed by Serra et al. [19]. It is interesting to see that is how the model interprets the flute to/from keyboard transfers. With image transfer[24], the model should change the horse into zebra with the background intact. In this transfer process, what has been changed is the part that cycleGAN interpret as the key difference between flute and keyboard.

### 4.2 Sound Consistency Testing using Classification

Classification is a common way to evaluate the output from style transfer network[12, 24, 3]. This is a quantitative way to examine the consistency of the outputs of the network. We tested whether our generated sounds are consistent on timbre via classification.

As a first simple test of consistency, we test whether the generated sounds are considered consistently closer to flute or keyboard by our audio-VGG, *which has had no exposure to the new sounds.* We took 1000 random selected clips of each of the three kinds of new generated mix sounds into the audio-VGG classification CNN. The classification returned for these test clips will show whether the audio-VGG defines the mixed sounds as closer to a flute or a keyboard.

Table 1 how of the mixed sounds are classified by the audio-VGG. Almost all of the mixed sounds are classified as a flute. We surmise that the generated mixed sounds therefore have common features that cause the audio-VGG classify them as the same sound.

| | flute | keyboard |
|---|---|---|
| faster fixed style transfer | 1000 | 0 |
| cycleGAN keyboard to flute | 998 | 2 |
| cycleGAN flute to keyboard | 992 | 8 |

Table 1: Number of instances per class when testing generated sounds using original audio-VGG

The next step was to verify whether the generated sounds are consistent with each other, and distinguishable and consistent from the source sounds, when classified by a model that has been exposed to the new generated sounds. For each style transfer method, we trained a classifier to distinguish between the source keyboard sounds, source flute sounds and the generated mixed sound. We are dealing with four types of mixed sounds, from: the baseline slow transfer, the faster fixed style transfer, the two results of cycleGAN: the $x_{fake}$ from $G$ and the $y_{fake}$ from $F$. For the baseline slow transfer, it is impractical to generate a large number of mixed spectrogram results needed for training, due to long processing time. We exclude this transfer process from our classification task. That leaves us with the three remaining generated sound types. For each of these three, we generated 8000 clips, and do the same train-test split as done in the previous training section(6000-1000-1000).

For each of our three generate sound types, we train a four-class classification network. The four classes consist of the original flute sounds, original keyboard sounds, the generated sounds from the relevant style transfer network and a fourth class - guitar sounds from the Nysnth dataset. The guitar class is a dummy class to introduce a class which is neither from the source sounds nor the generated mixed sounds. The structure of the network used was the VGG-19 network structure. The network was trained over 5 epochs, with a 0.0002 learning rate.

Table 2 shows the classification results across for four classes, for each of the faster fixed style transfer, cycleGAN form flute to keyboard and cycleGAN from keyboard to flute approaches.

| | Overall | Recall | | | |
|---|---|---|---|---|---|
| | Accuracy | flute | keyboard | guitar | mixture |
| faster fixed style transfer | 0.9947 | 0.995 | 0.996 | 0.988 | 1.000 |
| cycle GAN flute to keyboard | 0.9872 | 0.986 | 0.985 | 0.978 | 1.000 |
| cycle GAN keyboard to flute | 0.9735 | 0.993 | 0.933 | 0.969 | 1.000 |

Table 2: Class accuracy for 4 class VGG-19 for three generated sound types: Faster fixed style transfer and cycleGANs

The table gives the overall accuracy and the class accuracy for each class for each network. The overall performance of each network is high, indicating that the generated sounds are consistent and distinguishable from the natural sounds. The class accuracy for the generated sounds for all style transfer approaches is perfect. The small number of errors made are between the natural sounds with the keyboard class accuracy the lowest.

We then need to determine whether the generated sounds from the different style transfer approaches are different from each other. To check this, we trained a six-class classifier: the three different natural sounds, keyboard, flute and guitar and the three different generated sounds from the different style transfer methods, faster fixed style transfer, cycleGAN from flute to keyboard and from keyboard to flute. The structure of this network is also VGG-19 It is trained under the same train-test strategy and the same 5 epoch.

Table 3 shows the classification result from the network trained on six classes. The high recall score of three mixed sounds classes shows that the network can

| | Recall |
|---|---|
| flute | 0.914 |
| keyboard | 0.944 |
| guitar | 0.981 |
| faster fixed style transfer | 0.999 |
| cycleGAN from flute to keyboard | 0.996 |
| cycleGAN from keyboard to flute | 1.000 |

Table 3: Class accuracy for the 6 Classification network of source and generated sounds in a single model

distinguish the new generated audio signals from both natural sounds and from each other. We note that there is also some error in distinguishing the natural sounds. This may be because there are overlapping frequencies in the harmonics between those natural sounds[14].

### 4.3 Visualisation of Clusters

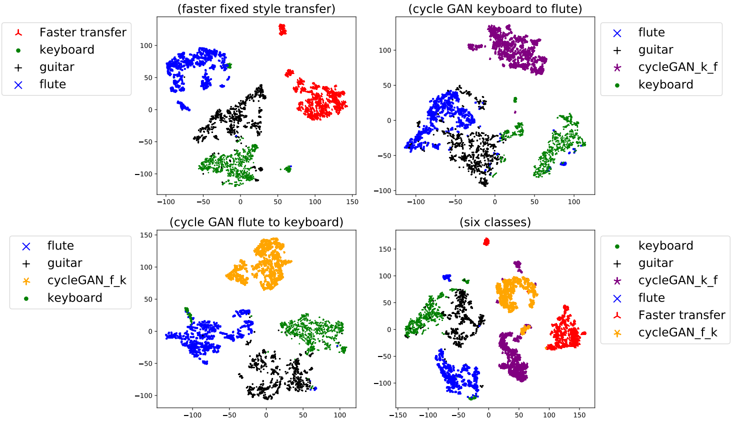


Fig. 3: t-SNE mapping of pool5

T-SNE[3, 16] is a clustering method used in visualization tasks of classification networks[7, 10], which we use to examine whether our new style transfer classes appear as clear clusters - and as separate clusters from the other classes. It is using a student-t distribution on the Stochastic Neighbor Embedding (SNE). SNE calculates the similarity by computing the conditional probability using Euclidean distances between instances. We apply t-SNE on the Pooling Layer 5[10] of every classification network.

Figure 3 shows the t-SNE mapping of layer pooling 5 for each classification network. Each dot represents a sound clip. Each kind of sound has its own color and shape. The "cycleGAN_f_k" denotes "cycleGAN from flute to keyboard" and vice versa. The generated mixed audio signals appear as clearly separated clusters from the original natural sounds clusters. It is interesting to see that for a classification network, it is easier to tell the difference between new generated mixed sounds and natural sounds, but it may get a little confused when classifying the classes which are all natural sounds.

## 5 CONCLUSIONS

Inspired by image neural style transfer, we applied three neural style transfer methods to audio mixing. All three methods can mix audio signals by mixing the visual style and content of audio spectrograms. The new generated audio signals are recognised by CNNs as individual classes. The t-SNE mapping shows that the new sounds are separate groups from the original sounds and from each other.

These new generated audio clips can be seen as a kind of morphing of two different kinds of audio signals, using visual concepts of style and content as our basis for mixing audio spectrograms. The next phase of work is to expand our techniques with a wider variety of audio signals, producing targeted sounds mixes that can be assessed by human listeners. To achieve this, we propose to examine the layers of a CNN trained on known audio signals to distinguish the feature maps at each layer, with a view to mapping timbre, pitch and tone to specific CNN layers.

### Acknowledgement

The authors would like to thank the help of Dr Sean O'Leary.

## References

1. Caetano, M.F., Rodet, X.: Sound morphing by feature interpolation. In: IEEE Inter Conf on Acoustics, Speech and Signal Processing. pp. 11–231 (2011)
2. Dieleman, S., Schrauwen, B.: End-to-end learning for music audio. In: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE Inter Conf on. pp. 6964–6968. IEEE (2014)
3. Donahue, J., Krähenbühl, P., Darrell, T.: Adversarial feature learning. arXiv preprint arXiv:1605.09782 (2016)
4. Engel, J., Resnick, C., Roberts, A., Dieleman, S., Norouzi, M., Eck, D., Simonyan, K.: Neural audio synthesis of musical notes with wavenet autoencoders. In: Proceedings of the 34th Inter Conf on Machine Learning-Volume 70. pp. 1068–1077. JMLR. org (2017)
5. Frigo, O., Sabater, N., Delon, J., Hellier, P.: Split and match: Example-based adaptive patch sampling for unsupervised style transfer. In: Proceedings of the IEEE Conf on Computer Vision and Pattern Recognition. pp. 553–561 (2016)
6. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conf on Computer Vision and Pattern Recognition. pp. 2414–2423 (2016)

7. Ghiasi, G., Lee, H., Kudlur, M., Dumoulin, V., Shlens, J.: Exploring the structure of a real-time, arbitrary neural artistic stylization network. arXiv preprint arXiv:1705.06830 (2017)
8. Han, Y., Kim, J., Lee, K., Han, Y., Kim, J., Lee, K.: Deep convolutional neural networks for predominant instrument recognition in polyphonic music. IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP) **25**(1), 208–221 (2017)
9. Handel, S.: Timbre perception and auditory object identification. Hearing **2**, 425–461 (1995)
10. Haque, A., Guo, M., Verma, P.: Conditional end-to-end audio transforms. arXiv preprint arXiv:1804.00047 (2018)
11. Hershey, S., Chaudhuri, S., Ellis, D.P., Gemmeke, J.F., Jansen, A., Moore, R.C., Plakal, M., Platt, D., Saurous, R.A., Seybold, B., et al.: Cnn architectures for large-scale audio classification. In: Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE Inter Conf on. pp. 131–135. IEEE (2017)
12. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. arXiv preprint (2017)
13. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Euro Conf on Computer Vision. pp. 694–711. Springer (2016)
14. Kaneko, T., Kameoka, H., Hiramatsu, K., Kashino, K.: Sequence-to-sequence voice conversion with similarity metric learned using generative adversarial networks. In: Proc. Interspeech. pp. 1283–1287 (2017)
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
16. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)
17. Marchi, E., Vesperini, F., Eyben, F., Squartini, S., Schuller, B.: A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional lstm neural networks. In: Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE Inter Conf on. pp. 1996–2000. IEEE (2015)
18. Schlüter, J., Grill, T.: Exploring data augmentation for improved singing voice detection with neural networks. In: ISMIR. pp. 121–126 (2015)
19. Serra, X., et al.: Musical sound modeling with sinusoids plus noise. Musical signal processing pp. 91–122 (1997)
20. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
21. Slaney, M., Covell, M., Lassiter, B.: Automatic audio morphing. In: Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conf Proceedings., 1996 IEEE Inter Conf on. vol. 2, pp. 1001–1004. IEEE (1996)
22. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI. vol. 4, p. 12 (2017)
23. Verma, P., Smith, J.O.: Neural style transfer for audio spectograms. arXiv preprint arXiv:1801.01589 (2018)
24. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv preprint (2017)