Audio Inpainting based on Self-similarity for Sound Source Separation Applications

Dan Barry
School of Computer Science
University College Dublin
Dublin, Ireland
dan.barry@ucd.ie

Alessandro Ragano
School of Computer Science
University College Dublin
Dublin, Ireland
alessandro.ragano@ucdconnect.ie

Andrew Hines
School of Computer Science
University College Dublin
Dublin, Ireland
andrew.hines@ucd.ie

***Abstract*—Sound source separation algorithms have advanced significantly in recent years but many algorithms can suffer from objectionable artefacts. The artefacts include phasiness, transient smearing, high frequency loss, unnatural sounding noise floor and reverberation to name a few. One of the main reasons for this is due to the fact that in many algorithm, individual time-frequency bins are often only attributed to one source at a time, meaning that many time-frequency bins will be set to zero for a separated source. This leads to an impressive signal to interference ratio but at the cost of natural sounding resynthesis. Here, we present a simple algorithm capable of audio inpainting based on self-similarity within the signal. The algorithm attempts to use the non-zero bin values observed in similar frames as substitutes for the zero bin values in the current analysis frame. We present results from subjective listening tests which show a preference for the inpainted audio over the original audio produced from a simple source separation algorithm. Further, we use the Fréchet Audio Distance metric to evaluate the perceptual effect of the proposed inpainting algorithm. The results of this evaluation support the subjective test preferences.**

***Index Terms*—audio, source separation, inpainting, self-similarity**

I. Introduction

Audio signals such as music and speech are commonly subjected to processing that can lead to a loss of subjective audio quality. This degradation in quality is largely due to artefacts introduced by a specific process. Typical examples include: missing chunks of audio as a result of packet loss in VOIP systems, spectral and temporal artefacts as a result of low bitrate encoding and bandlimiting due to channel limitations to name but a few. In recent years, various algorithms have been proposed to restore audio fidelity in the above examples [1]–[5]. Many of the approaches use a concept known as inpainting, which has its roots in image signal processing. Inpainting is essentially the process of estimating missing data in an image or audio signal.

In this paper we focus on a specific audio process known as sound source separation. In general, sound source separation refers to the task of extracting individual sound sources from mixtures of those sound sources. Applications of source separation include speech enhancement, noise reduction, upmixing, remixing and remastering.

Like the processing examples above, sound source separation typically leads audible artefacts in the output. You can listen to and compare various source separation examples here [6] to get an idea of the sorts of artefacts to be expected. There are different approaches to sound source separation, however, many algorithms use some form of binary mask estimation in the time-frequency domain which leads to significant numbers of time-frequency values being set to zero. This creates little gaps in the time-frequency domain of each source which in turn leads to audible artefacts upon resynthesis. This is not too dissimilar to the image inpainting problem but instead of filling gaps in an image, we need to fill the gaps in the time-frequency domain with data such that it produces a more natural resynthesis of the separated sources. In this paper, we present an inpainting algorithm to mitigate the artefacts associated with binary mask based sound source separation processes.

II. Background

The field of sound source separation has advanced significantly since its inception. A good review of the area can be found in [7]. Much of the focus in recent years has been on applying machine learning approaches, however, availability of training material remains an issue. Deezer recently released Spleeter [8] which is widely regarded as state of the art for the purpose of vocal extraction. The separation quality for other instruments has not yet reached a comparable level owing to the lack of training data. As a result, traditional source separation models requiring no training data can still provide favourable results in certain cases. The self-similarity based inpainting algorithm presented in section III aims to mitigate some of the artefacts created by many time-frequency domain source separation algorithms. The artefacts include phasiness, transient smearing, high frequency loss, unnatural sounding noise floor and reverberation to name a few. Previous attempts to provide inpainting solutions for sound source separation include [9] but no results were presented.

To evaluate the proposed inpainting algorithm we use a source separation mixing model based loosely on [10]–[12] and described in (1).

Funded in part by the Science Foundation Ireland (SFI), and the European Regional Development Fund under Grant 12/RC/2289-P2 and Grant 13/RC/2077

$$X_c(k,m) = \sum_{j=1}^{J} a_{cj} S_j(k,m), \quad c = 1,2 \tag{1}$$

where $X_c(k,m)$ is the magnitude spectrogram of the left or right channel of a stereo recording denoted by c. The bin and frame indices are denoted by k and m respectively. The j^{th} source magnitude spectrogram is denoted as $S_j(k,m)$ and a_{cj} is an amplitude coefficient.

For many source separation algorithms, the approach is to find a binary mask, $B_j(k,m)$, for each source, such that when multiplied by one of the original mixture magnitude spectrograms $X(k,m)$, approximates the j^{th} source spectrogram as

$$S_j(k,m) = B_j(k,m) \times X(k,m). \tag{2}$$

There are various methods for calculating binary masks but here we will assume the intensity panned stereo mixing model and use a simple intensity ratio between left and right channels for each bin in order to cluster and separate frequency components panned to a single direction in the stereo field based on [10]–[12]. Many of the current 2 channel separation algorithms use some derivative of the above in order to calculate a binary mask for each source.

In the following sections, we describe a self-similarity based inpainting algorithm designed to be used as a post-process for any time-frequency based source separation algorithm with the aim of improving perceptual quality of the output.

III. Inpainting Methodology

A. Algorithm Overview

The self-similarity based inpainting algorithm we propose here assumes a separated spectrogram as its input. A block diagram of the algorithm is shown in Fig. 1.

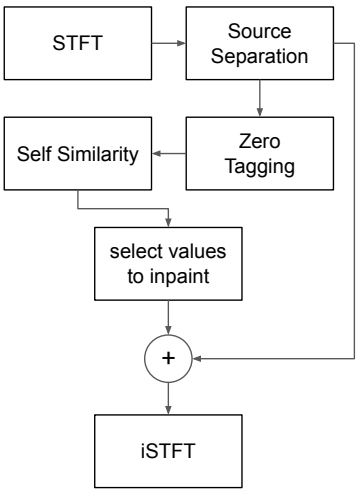

Fig. 1. Block diagram of the proposed self-similarity inpainting algorithm.

The first step is to identify which of the zero bin values should be inpainted and which should be left alone. This is discussed in section III-B. Next, we identify similar spectrogram frames in order to find non-zero value candidates to inpaint for our current frame, as discussed in section III-C. Signal reconstruction is briefly discussed in section III-D.

B. Zero Tagging

Within the context of a separated source spectrogram, two types of zero valued bins are observed; zeros resulting from the source truly having little no activity in the bin and zeros resulting from failures in the source separation algorithm to identify that some energy in the spectrogram should have been attributed to this source. By identifying and tagging bins containing 'true zeros', we can avoid inpainting those bins which would ultimately lead to objectionable artefacts. For the simple source separation model we are using to test our inpainting algorithm, we use a simple metric to decide weather a bin should be tagged as a true zero. For every zero bin in the separated spectrogram, $S_{k,m}$, we check the phase difference between the left and right mixture phase spectrograms for that bin. If the phase difference is close to zero, we assume that only one source contributed to that bin. The fact that our separated spectrogram contained a zero value in this bin means that another source was likely the sole contributor at this bin and thus it is a true zero for the current source. These true zeros are then given a very small value (3) so they will not be identified as zeros for inpainting in the next stage.

$$S_{k,m} = 0.001 \quad if \quad S_{k,m} = 0 \ and \ \Delta\Phi_{k,m} > T_1 \tag{3}$$

where $\Delta\Phi$ is the phase difference between the left and right phase spectrum and where $0 < T_1 < 2\pi$.

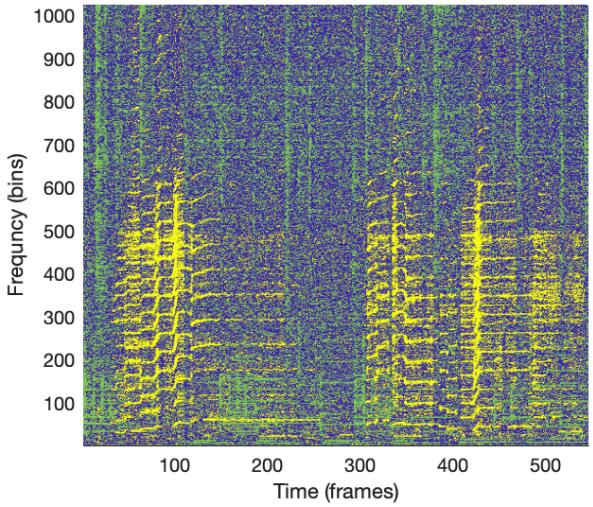

Fig. 2. Spectrogram (0 - 10 kHz) showing 12 seconds of separated source audio in yellow, true zero bins in green and valid bins to be inpainted in dark blue.

Fig. 2 shows an example of this process in action. Time-frequency bins in yellow are the separated source (a saxophone in this case), bins in green are all other source bins known to belong to only one source and therefore are true zeros for the current source. They should not be inpainted. All remaining bins in dark blue are then candidates for inpainting.

C. Self-Similarity Calculation and Zero Replacement

In order to find similar spectrogram frames from which to reconstruct the frame of interest, we use the correlation function (4) in order to create a correlation matrix. An example of such a correlation matrix can be seen in Fig. 3. The correlation matrix shows clear similarity between several frames in the spectrogram. Having compared every frame to every other frame in the spectrogram, we select the P most similar frames from which to reconstruct the current frame (5).

$$r(i,j) = \frac{\sum_{k=1}^{N} \left(S_{k,i} - \bar{S}_i\right)\left(S_{k,j} - \bar{S}_j\right)}{\sqrt{\sum_{k=1}^{N} \left(S_{k,i} - \bar{S}_i\right)^2 \sum_{k=1}^{N} \left(S_{k,j} - \bar{S}_j\right)^2}} \quad (4)$$

where r is the normalised correlation coefficient in the range $-1 < r < +1$ between the i^{th} and j^{th} frame of the spectrogram S containing N bins.

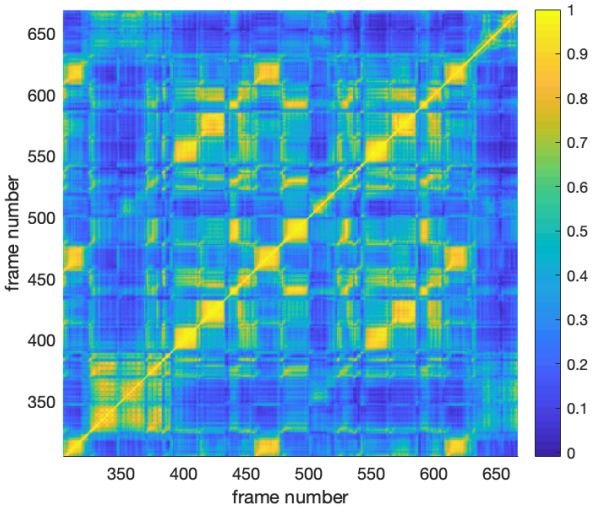

Fig. 3. The correlation matrix clearly shows the similarity between every possible pair of frames within the spectrogram. Brighter points correspond to higher correlation and therefore higher similarity between frames.

We select a set of P frames which are most similar to the current analysis frame provided the similarity exceeds a user defined threshold, T_2, such that $0 < T_2 < 1$ as described in (5).

$$Z_{k,p} = S_{k,j} \quad if \quad r(i,j) > T_2 \quad (5)$$

where Z is a matrix containing the P most similar frames to the current analysis frame, i.e., j = p_1, p_2, p_3 ... P.

Then, we use the minimum non-zero values from the matrix Z as the inpainting values for the current analysis frame as shown in (6) and illustrated in Fig. 4. We choose the minimum values to minimise the undesirable distortion introduced by the inpainting process. These values are almost certainly sub-optimal but we need to strike a balance between selecting better values than the existing zeros but at the same time not introducing too much undesirable noise.

$$S_{k,i} = min(Z_{k,p}) \quad if \quad S_{k,i} = 0 \quad (6)$$

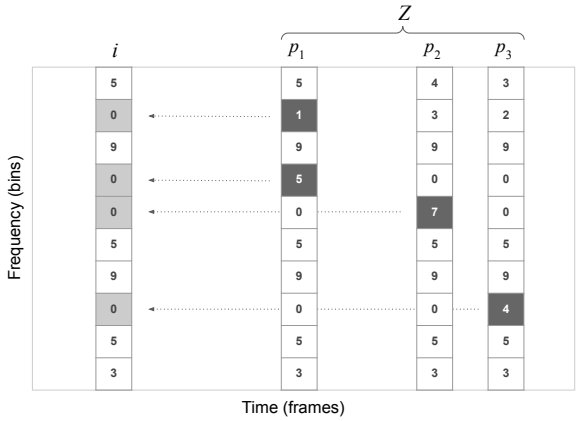

Fig. 4. Zero values in the current spectrogram frame, i, are replaced with the minimum non-zero values observed in the P most similar frames in the spectrogram. The minimums are depicted in the dark cells.

Currently, the algorithm parameters T_1, T_2 and P need to be set manually for each example. In practice we found that the following values worked well for the majority of the examples we used in the listening tests: $T_1 = 0.1$, $T_2 = 0.5$ and $P = 3$. When P is set too low, there is a higher probability that the similar frames will share the same zero locations as the current analysis frame resulting in little or no inpainting. If P is set too high, the algorithm has many more candidates (with reducing similarity) to choose minimum values from which eventually becomes equivalent to inpainting with the global minimums of the spectrogram. In other words, a static noise floor is painted in. The highest value used in these examples was $P = 6$. In general, setting $T_2 < 0.5$ results in non-similar frames being used for inpainting. This leads to objectionable artefacts in the output. T_1 is only really relevant if you are using a stereo mixing model in the separation stage, however, the authors would encourage users to find a way to identify true zero bins which should not be inpainted within their specific separation models.

D. Signal Reconstruction

Once the inpainting stage is complete, the signal is reconstructed using an inverse Fourier transform with the same parameters used during the Fourier analysis stage (4096 window length with 75% overlap). Additionally, we also re-window each frame before overlap-add for reasons described in [12]. An example of reconstructed audio can be seen in Fig. 5 where the top plot shows the output of the source separation algorithm (drums and bass) before inpainting, the middle plot shows the additional audio created by the inpainting algorithm and the bottom plot shows percentage of non-zero bins before and after inpainting for each frame of the spectrogram. As can be seen, only 25% of the original time-frequency bins before inpainting are non-zero but inpainting estimates a further 50% of bin values.

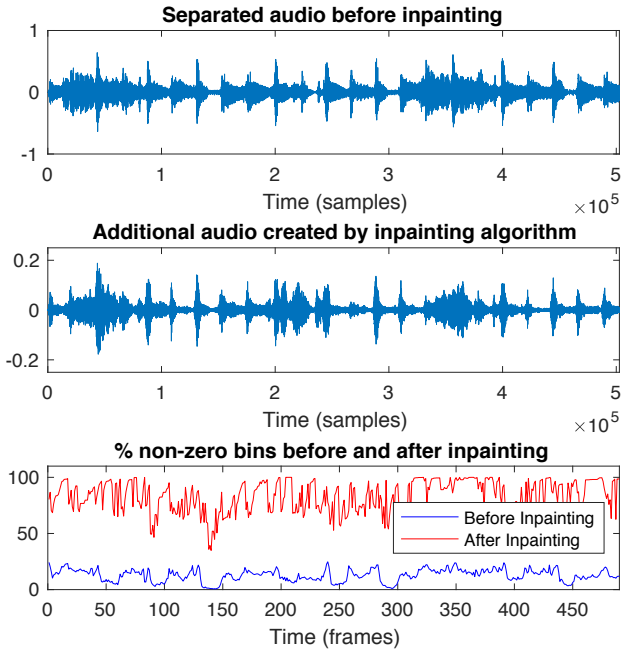

Fig. 5. Inpainted audio: top plot shows separated audio before inpainting, middle plot shows additional audio created by inpainting (note scale), bottom plot shows the % zero bins before and after inpainting

IV. Testing

A. Subjective Testing

In order to test the efficacy of the inpainting algorithm, we conducted subjective listening tests online using 18 subjects. Each subject was presented with 10 audio examples and asked for their preference between an inpainted version and an original version of the same audio clip.

The audio examples chosen are real commercial audio recordings (16 bit, 44.1 kHz). Table I shows the recordings used along with the instruments separated for test purposes. The inpainting algorithm parameters for each example, T_1, T_2 and P were set manually by the authors. The examples, range in length from 15 seconds to 1 minute.

TABLE I
Test Material

No.	Artist	Song	Separation
1	Pink Floyd	Money 1	Guitar
2	Pink Floyd	Money 2	Drums-Bass
3	Pink Floyd	Money 3	Drums-Bass-Vox
4	Miles Davis	Pee Wee 1	Bass
5	Miles Davis	Pee Wee 2	Saxophone
6	David Bowie	Ziggy Stardust	Voice-Guitar
7	Frank Zappa	Montana	Drums-Guitar
8	Foo Fighters	Rope	Drums-Bass
9	Queen	Bohemian Raph.	Voice
10	Jimi Hendrix	Ain't No Tellin	Drums-Bass

The subjects were between the ages of 26 and 60 with 60% under the age of 40. The subjects were asked to describe their knowledge of audio as follows; 40% identified as professionals working in the industry, 35% identified as audio enthusiasts and 25% identified as appreciating good quality audio but rarely paying attention to it. All subjects undertook the test in a quiet room using headphones (40% in ear and 60% on/over ear).

The listening test interface was built using HTML and Javscript and was presented online such that subjects could undertake the test remotely. For each example, subjects were presented with a simple interface containing 5 buttons to control audio playback. Buttons allowed the subjects to play and pause the audio example and skip back 5 seconds. Two larger buttons labeled A and B allow the subjects to synchronously switch between option A and B seamlessly. The audio automatically loops until stopped. The inpainted version was presented in position A and B an equal number of times. Using a simple interface, subjects preferences were logged as: 'A', 'B' or 'No Preference' for each example. They were also asked to briefly comment on their preference for each example. Results of subjective listening tests are presented in section V.

B. Objective Testing

Without a reference signal to compare to, objective audio quality models like PEAQ [13] or ViSQOLAudio [14], [15] cannot be used to estimate the quality. The community has relied on signal to interference ratio as an estimate but it can be misleading. We used the recently proposed Fréchet Audio Distance (FAD) [16] which is a reference-free objective metric for music enhancement algorithms. FAD compares the embedding statistics generated on a large reference set of clean music with the embedding statistics generated on an evaluation set of enhanced noisy signals. First, embeddings are generated by the VGGish[1] model. Next, multivariate Gaussians are computed on both the evaluation set embeddings $\mathcal{N}_e(\mu_e, \Sigma_e)$ and the reference set embeddings $\mathcal{N}_r(\mu_r, \Sigma_r)$. Finally the Fréchet distance between the two Gaussians is computed as

$$F(\mathcal{N}_r, \mathcal{N}_e) = \|\mu_r - \mu_e\|^2 + tr(\Sigma_r + \Sigma_e - 2\sqrt{\Sigma_r \Sigma_e}). \quad (7)$$

When comparing several music enhancement algorithms, the one with the lowest FAD score shows highest sound quality. FAD has been tested on different distortions such as Gaussian noises, pops, frequency filter, quantization, Griffin-Lim distortion, mel encoding, speed change, reverberations and pitch change which makes it suitable for evaluating the inpainting algorithm.

In our experiment we create one reference set and two evaluation sets that we want to compare:

- The reference set consists of 12 hours of modern commercial rock and pop recordings.
- The original evaluation set consists of the separated tracks shown in Table I before inpainting.
- The inpainted evaluation set consists of the separated tracks shown in Table I after inpainting.

An important aspect to consider when using the FAD [16] is the size of the evaluation set as it affects the index of

[1]https://github.com/tensorflow/models/tree/master/research/audioset

dispersion of the Gaussians. The authors recommend using at least 25 minutes of audio to obtain reliable results. We used 34 minutes of audio in our evaluation set. To evaluate the effect of the size in the evaluation set, we created two more evaluation sub sets that only include 10 minutes of audio from the larger evaluation sets. We refer to these as the large and small evaluation sets respectively. The results are presented in section V-A below.

V. Results and Discussion

Table II summarises the subjective preference results for each audio example along with the percentage of zeros inpainted for reference. Across 180 tests, the mean scores show that the overall preference is for the inpainted algorithm at 45%. Preference for the original audio with no inpainting was 30%. Finally, 25% of tests resulted in no preference.

TABLE II
Results

	%Preference			Reconstruction
No	***Original***	***Inpainted***	***No Pref***	***% Inpainted Zeros***
1	**50**	28	22	44
2	17	**22**	61	33
3	33	**50**	17	78
4	**50**	44	6	60
5	**39**	33	28	49
6	17	**56**	28	50
7	33	**56**	11	52
8	6	**83**	11	44
9	**39**	22	39	63
10	28	**67**	6	73
Mean	30	45	25	55

Breaking down the results by example, it can be seen that 6 out of the 10 examples resulted in a preference for the inpainted audio. The authors have observed that all 6 examples showing a preference for the inpainted audio contained multiple instruments in the separation (typically centre channel extractions) and 5 out of those 6 examples contained drums. The 4 examples in which the original audio was preferred were single instrument source separations. Fig. 6 provides a visual representation of the absolute preferences for each audio example.

We carried out a t-test to investigate the statistical significance of the results presented in Table II. The p-value obtained for all 10 examples is 0.072 but grouping the examples according to whether they contained drums or not, we observe the a p-value of 0.022 for examples with drums and a p-value of 0.801 for examples with no drums. This finding indicates that the inpainting algorithm leads to preferable results for separations containing multiple sources and drums.

The magnitude and direction of preference for each audio example can be seen in Fig. 7. On average, when there is a preference for the inpainted algorithm, it is notably larger than the preference for the original algorithm.

After completing each audio example in the listening test, subjects were asked to comment on their choice. From this we gathered a total of 180 comments about preference which

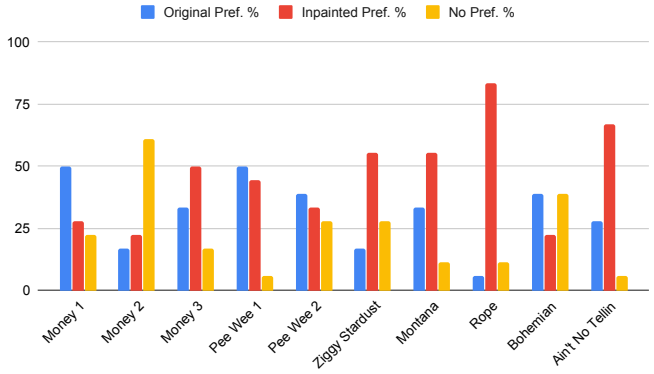

Fig. 6. Bar chart showing the % preferences for the original audio (blue), the inpainted audio (red) and no preference (orange)

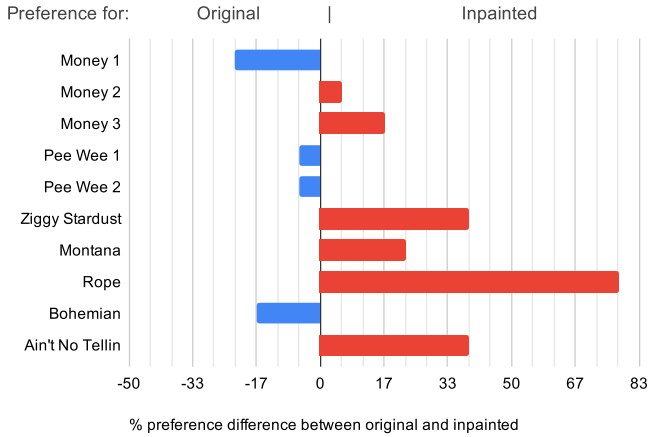

Fig. 7. Bar chart showing the % spread of preference for each audio example. Preference for the original algorithm is shown in blue using negative numbers and preference for the inpainted algorithm is shown in red using positive numbers

we summarise as follows. Where the preference was for the inpainted audio, the most frequent reasons cited were:

- increased bandwidth, clarity and high-frequency content
- more natural and breathy
- less artefacts and musical noise

Where the preference was for the original audio, the most frequent reasons cited were:

- less noise
- less distortion
- cleaner sounding

These comments reflect what the authors have observed. The inpainting algorithm creates its own artefacts but in most cases these artefacts are less objectionable and serve to mask the original source separation artefacts. However, in certain cases the inpainting artefacts are more objectionable than the original artefacts. The algorithm is prone to inpainting irrelevant signals from similar frames. For example, frames which are largely silent can often get inpainted with random signal parts from other frames. This is particularly objectionable but

depends greatly on the signal. The principal observation is that the inpainting algorithm is capable of restoring transients, high-frequency content, ambience and generating more natural noise floors.

A. Objective Quality

In Figure 8 we compare the original audio with the inpainted audio using the FAD scores.

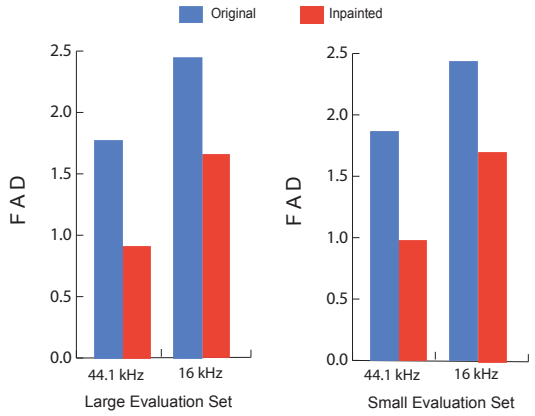

Fig. 8. Bar chart showing the FAD scores. We compare the original and the inpainted audio by varying the sampling rate and the evaluation set size. Lower FAD scores indicate better quality

In each experiment the inpainted algorithm shows superior performance compared to the original separated tracks. This means that the quality of the inpainted audio is closer to the commercial music reference set than the audio before inpainting. We found that the small evaluation set of 10 minutes did not influence the results. It must be noted that the recommendation by the developers of FAD to use at least 25 minutes came from evaluating the index of dispersion and not from testing output FAD scores. We found that FAD was not affected by reducing the evaluation set but we did not evaluate the index of dispersion.

We also explored the behaviour of FAD by varying the sampling rate. In [16], the authors tested FAD on audio sampled at 16 kHz. We wished to evaluate the proposed method on full bandwidth audio. In each bar chart we use the same sampling rate in both the reference and evaluation set to avoid a sample rate mismatch. We see that a lower sampling rate increases FAD scores which means lower quality. This suggests that the FAD output is sensitive to the sampling rate.

VI. Conclusions

We have presented a self-similarity based inpainting algorithm designed to improve the subjective quality of any time-frequency based source separation algorithm. We performed subjective listening and objective FAD testing, both of which indicate a preference for the inpainted separations over the separated audio alone using a popular source separation method as a test case. The algorithm appears to work better when the separations contain drums and multiple sources over singular sources. As a result of this, we intend to investigate the use of the algorithm for the source removal problem such as that of removing instruments to create backing tracks or dialogue removal. Furthermore, subjects report that the main benefit of the inpainting process is the restoration of high frequency content leading to more natural sounding results. The strength of the algorithm is its simplicity and the fact that it requires no training. It is currently a proof of concept and no attempt was made to optimise parameters. Future work will focus on refining the self-similarity search space and inpainting thresholds which would further reduce artefacts. In this paper, we also conducted objective testing using the Fréchet Audio Distance metric. We have shown that the FAD metric is robust to variations in sample rate and evaluation set size. The FAD results support the findings of the subjective listening test preferences.

References

[1] Y. Bahat, Y. Y. Schechner, and M. Elad, "Self-content-based audio inpainting," *Signal Processing*, vol. 111, pp. 61–72, 2015.

[2] A. Marafioti, N. Holighaus, P. Majdak, N. Perraudin *et al.*, "Audio inpainting of music by means of neural networks," in *Audio Engineering Society Convention 146*. Audio Engineering Society, 2019.

[3] P. Peter, J. Contelly, and J. Weickert, "Compressing audio signals with inpainting-based sparsification," in *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer, 2019, pp. 92–103.

[4] V. Kuleshov, S. Z. Enam, and S. Ermon, "Audio super resolution using neural networks," *arXiv preprint arXiv:1708.00853*, 2017.

[5] M. Chinen, W. B. Kleijn, F. S. Lim, and J. Skoglund, "Generative speech enhancement based on cloned networks," in *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2019, pp. 214–218.

[6] (2018) Audio examples and results from the 2018 signal separation evaluation campaign. [Online]. Available: https://sisec18.unmix.app//results/vocals/SDR

[7] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, D. FitzGerald, and B. Pardo, "An overview of lead and accompaniment separation in music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 8, pp. 1307–1335, 2018.

[8] R. Hennequin, A. Khlif, F. Voituret, and M. Moussalam, "Spleeter: A fast and state-of-the art music source separation tool with pre-trained models," in *Proc. International Society for Music Information Retrieval Conference*, 2019.

[9] D. FitzGerald and D. Barry, "On inpainting the adress algorithm," in *IET Conference Proceedings*. The Institution of Engineering & Technology, 2012.

[10] S. Rickard, "The duet blind source separation algorithm," in *Blind speech separation*. Springer, 2007, pp. 217–241.

[11] C. Avendano and J.-M. Jot, "A frequency-domain approach to multi-channel upmix," *Journal of the Audio Engineering Society*, vol. 52, no. 7/8, pp. 740–749, 2004.

[12] D. Barry, B. Lawlor, and E. Coyle, "Sound source separation: Azimuth discrimination and resynthesis," in *7th International Conference on Digital Audio Effects, DAFX 04*, 2004.

[13] T. Thiede, W. C. Treurniet, R. Bitto, C. Schmidmer, T. Sporer, J. G. Beerends, and C. Colomes, "PEAQ - The ITU standard for objective measurement of perceived audio quality," *Journal of the Audio Engineering Society*, vol. 48, no. 1/2, pp. 3–29, 2000.

[14] C. Sloan, N. Harte, D. Kelly, A. C. Kokaram, and A. Hines, "Objective assessment of perceptual audio quality using ViSQOLAudio," *IEEE Transactions on Broadcasting*, vol. 63, no. 4, pp. 693–705, 2017.

[15] M. Chinen, F. S. Lim, J. Skoglund, N. Gureev, F. O'Gorman, and A. Hines, "Visqol v3: An open source production ready objective speech and audio metric," in *12th Intl. Conf. on Quality of Multimedia Experience (QoMEX)*. IEEE, 2020.

[16] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms," in *Proc. Interspeech 2019*, 09 2019, pp. 2350–2354.