AUDIO MORPHING USING MATRIX DECOMPOSITION AND OPTIMAL TRANSPORT

Gerard Roma , Owen Green and Pierre Alexandre Tremblay

CeReNeM
University of Huddersfield
Huddersfield, UK
{g.roma, o.green, p.a.tremblay}@hud.ac.uk

ABSTRACT

This paper presents a system for morphing between audio recordings in a continuous parameter space. The proposed approach combines matrix decompositions used for audio source separation with displacement interpolation enabled by 1D optimal transport. By interpolating the spectral components obtained using non-negative matrix factorization of the source and target signals, the system allows varying the timbre of a sound in real time, while maintaining its temporal structure. Using harmonic / percussive source separation as a pre-processing step, the system affords more detailed control of the interpolation in perceptually meaningful dimensions.

1. INTRODUCTION

The term *morphing* is often used in sound synthesizers to refer to continuous interpolation in a timbre space, such as in wavetable synthesis. It was used in early audio spectral synthesis literature to refer to interpolation between sounds analyzed in the time-frequency domain [1, 2]. This often includes consideration of the temporal evolution of sounds, beyond real-time morphing of individual spectral frames. As noted in [3], there is little consensus about what morphing really means in the acoustic domain. The interpolation is typically an affordance of the computational models or signals, which does not guarantee an interpolation in the perceptual features of the resulting sound. Regardless, the concept is established in practice, and implemented in well-known commercial products such as Zynaptiq Morph [4] or MeldaProduction MMorph [5].

Existing audio recordings are generally used to guide user interaction in many spectral processing techniques. A broad distinction can be made between the use of large corpora (e.g. concatenative synthesis and mosaicing), and methods for hybridizing short sounds (for which the term *cross-synthesis* is often used).

In this paper we propose an extension of a cross-synthesis algorithm based on non-negative matrix factorization (NMF) to continuous morphing, using optimal transport. This allows morphing between two sounds independently in several dimensions corresponding to structural components of both sounds. We then explore the use of harmonic-percussive source separation (HPSS) as a pre-processing step, which allows applying the morphing algorithm independently to the harmonic and percussive parts of the sounds. This enables devising an interface for morphing using perceptually relevant interpolation parameters, independent of the number of NMF components.

Audio hybridization is generally useful in creative applications related to audio, allowing the creation of new sounds and the production of multiple variations based on perceived features of audio samples. Our algorithm offers more nuanced control of the process by providing a continuous parameter space. After a non-realtime analysis, the parameters affecting the morphing can be operated in real time.

In the next section we briefly review existing approaches to audio morphing and cross-synthesis. We then describe the NMF-based method as originally applied to cross-synthesis. In Section 4 we describe the extension to continuous morphing. We then introduce the HPSS pre-processing step in Section 5, and summarize the phase generation strategy in Section 6. In Section 7 we present two implementations of the proposed algorithm and discuss the effect of different parameters.

2. RELATED WORK

The idea of morphing between sounds has been investigated since the early days of audio spectral analysis / synthesis research. In [6] an application of the STFT was described where the spectrum of a modulator sound is smoothed and multiplied with a carrier sound, thus combining the timbre of the modulator and the pitch of the carrier. The system in [1] similarly involved separate matching of pitch and timbre components through the inversion of mel frequency cepstral coefficients (MFCC). In this case, however, the temporal dimension was taken into account by aligning the sounds using dynamic time warping (DTW). Several approaches were proposed focusing on sinusoidal models [7, 3, 8, 2]. This allows a more nuanced approach with respect to perceptual qualities of the interpolation, but typically requiring a specific focus on musical instrument sounds and human or animal voices.

More recently, with the popularization of the concatenative synthesis and mosaicing paradigms, most innovations have focused on dictionary-based methods, particularly matching pursuit (MP) [9]. Such approaches offer a promising theoretical framework for audio morphing, but it is still hard to obtain convincing sound quality. Non-negative matrix factorization can be seen as another dictionary-based method that can be naturally applied to audio by leveraging the 2D structure and positive nature of magnitude spectrograms.

NMF-based cross-synthesis was proposed in [10] using two separate decompositions. Another NMF method (more in line with dictionary-based methods) was proposed in [11], where the NMF optimization is used to reconstruct a target spectrogram using an existing recording as a dictionary. We discuss these methods in more depth in the next section. Our approach uses the same method described in [10], but extends it to continuous morphing.

Optimal transport has recently gained popularity in machine learning [12, 13]. It has been applied to unmixing within an NMF framework in [14], under strong harmonicity assumptions. In [15] optimal transport was used to obtain "automatic portamento" between audio streams, which is in effect a form of frame-level audio morphing. By applying this technique to the NMF decomposition, our system allows component-wise morphing, including the temporal dimension.

Finally, some recent work has explored the application of the popular style transfer mechanism developed for images using convolutional neural networks (CNN) [16, 17, 18]. The structure is similar to previous approaches in that one sound is used as the "content" while the other is used as "style". Some works have also explored the use of neural networks for morphing between musical instrument sounds using WaveNet [19] or variational autoencoders (VAE) [20]. Deep learning is generally promising for the task of audio morphing and hybridization but—given the costs of training and the limitations of supervised models—unsupervised techniques such as NMF are still appealing, considering the computing resources available in current music production setups.

3. NMF-BASED CROSS-SYNTHESIS

NMF is a very established technique in audio signal processing, since its introduction for transcription [21]. Using Wiener filtering, it can be used for basic audio source separation [22], which works well for simple signals such as drums or piano recordings. The decomposition is usually applied to a magnitude spectrogram V, which is approximated as the product of a matrix W—whose columns can be interpreted as spectral frame prototypes—and a matrix H where the rows represent the continuous activation of the prototypes in W:

$$\hat{V} = WH \tag{1}$$

The main parameter is the rank which corresponds to the number of columns of W and rows of H. The decomposition thus represents the sound as an additive combination of spectral components that can occur simultaneously.

The matrices W and H can be obtained via multiplicative updates rules, often using an extension of the the Kullback-Leibler (KL) divergence to positive matrices:

$$D(V, \hat{V}) = \sum_{kn}(V(k,n) log \frac{V(k,n)}{\hat{V}(k,n)} - V(k,n) + \hat{V}(k,n)), \tag{2}$$

where k is the frequency index and n is the time index. A straightforward way to hybridize sounds is thus to decompose each one with the same rank using NMF, and multiply the activations of one with the spectral bases of the other. This way two spectrograms can be obtained: $V_1 = W_1H_2$ and $V_2 = W_2H_1$. In this case, one of the sounds is used for the activations, but the corresponding spectral frames are substituted for those of the second sound. In this paper we will explore interpolating between the bases of the first decomposition and those of the second decomposition, while using the activations of the first one. Hence, from now on we will use V^s (approximated by W^sH^s) to denote the "source" spectrogram, and V^t (and correspondingly W^tH^t) for the "target" spectrogram.[1]

The activations, H, will typically capture rhythms and general structure, while the spectral bases, W, will represent pitches and timbre. This approach was described in [10], using euclidean distance instead of KL divergence as the NMF minimization cost. Such a procedure produces a new spectrogram that has not been generated from a waveform, and the phase is commonly synthesized using the Griffin-Lim algorithm (GL) [23]. In this paper we use the phase gradient heap integration (PGHI) algorithm, a more recent method that can run in real time [24].

A second approach, proposed in [11] is to use NMF to find the activations that will allow a reconstruction of the first spectrogram, using the frames of the second spectrogram as the W matrix. Here the multiplicative updates are only applied to the H matrix and only one NMF computation is required. The phase of the second spectrogram is used in the reconstruction by multiplying the obtained real H matrix with the complex spectrogram used for the bases. This algorithm follows the concatenative synthesis paradigm of reconstructing a sound with a dictionary of spectral frames. The work in [25] proposed some extensions and provided an open implementation. Another implementation has been included in the NMF Toolbox in [26].

In our experience with these implementations, it is quite hard to obtain good results for hybridizing short sounds. Since concrete spectral frames are used as spectral bases (as opposed to the prototype bases learnt by a separate NMF process), the reconstruction suffers from the superposition of spectral frames and the repetition of frames with the same phase. To avoid these problems, the authors in [11] introduced a number of additional constraints that result in a sparse H matrix. This matrix no longer represents the activation of the components of the target spectrogram but rather the selection of frames from the source spectrogram, given the constraints. In addition, in order to obtain reasonable results, large dictionaries are required (potentially on the order of thousands, corresponding to frames in the spectrogram used as W), which implies computing an NMF with a very large rank.

In this paper, we extend the original method in [10], based on separate decomposition of source and target. This has the problem that the decomposition $\hat{V} = W^sH^t$ no longer represents a valid NMF problem, and $\hat{V}$ no longer comes from an existing spectrogram. However, from a usability perspective, it has the advantage that the NMF framework is used to model the main components of each sound as a set of spectral templates and their activations, which are then combined. The temporal and spectral properties of each sound can thus be perceived in the result, and the process can be easily understood. Figure 1 shows an example of crossing the activations of a drum pattern with three piano chords using rank 3.

In the system proposed in [10], the rank is mostly chosen to capture musical notes or harmonics. However in the general case the optimal rank for the decomposition is not known. In order to estimate the rank, we use the approach proposed in [27], based on singular value decomposition (SVD). The rank is chosen using a parameter p that accounts for the proportion of the sum of singular values of the SVD decomposition of V^s with respect to the total. The number of singular values is the rank estimate.

Another issue is how to match the columns of W^t with the rows of H^s. Each assignment of basis to activation will lead to a different result. For low ranks (e.g. less than 10), it may be intuitive enough to do the assignment manually. However, since

[1] Note that this is different to the wording used in other works (e.g. [11]) where the W matrix used for its timbre is denoted as the "source", and the spectrogram used for activations is seen as the "target". Here the target is the "destination" of the morphing.

the algorithm is initialized with random matrices, the order of the result is not guaranteed to be the same each time, which requires some additional analysis for sorting the columns. For large ranks (e.g. in the order of tens of components, as commonly obtained using the SVD estimate), manual assignment is not viable. One way to automate this process is to assign to each activation the basis of the timbre spectrogram that is most similar to its original corresponding activation. However, there is the risk of the same column of W^s being assigned to more than one activation and vice versa. The distribution of spectral features is known to produce *hub* vectors that dominate similarity spaces [28]. Therefore given very different materials we could find that a few bases of one sound are similar to all the bases in the other sound. To avoid this, an injective constraint was proposed as a user option in [10]. In this paper we propose a more automated approach, which allows the use of larger ranks. We treat this mapping as an assignment problem, defined by the cost matrix

$$C_{ij} = d({w^s}_i, {w^t}_j), \tag{3}$$

where $d(x, y)$ represents the distance between two spectral bases (defined in the next section), and ${w^s}_n$ represents the nth column of the W^s matrix and analogously for W^t. The assignment is solved using the Hungarian algorithm, as recently proposed for matching spectral peaks to partial tracks in sinusoidal modeling [29]. The algorithm works for rectangular matrices, but the number of assignments always corresponds to the smaller side, so no element is used twice. Given the ranks k^s, k^t obtained via SVD for the two NMF decompositions, we restrict k^t to $\min(k^s, k^t)$, in order to avoid using the same base twice.

Finally, an issue with this approach is that, depending on the original amounts of energy represented by each of the components, the new combinations driven by the unrelated activations can create spectrograms that exceed the maximum magnitudes allowed by the STFT representation. To avoid this, we simply scale the resulting spectrogram based on the ratio of the RMS magnitude to the original spectrogram's as a gain factor:

$$g = \max(1, \frac{RMS(V^s)}{RMS(\hat{V})}) \tag{4}$$

4. DISPLACEMENT INTERPOLATION

The combination of the activations of one sound with the bases of another source provides a way to create new spectrograms that share some properties of both. The result will generally follow the temporal energy patterns of the first source, encoded in the activations, and the spectral energy patterns in the second source (notes and / or timbre components depending on the material and the NMF rank).

For each pair of matched spectral bases ${w^s}_i, {w^t}_j$, we morph between the original and new spectral patterns by interpolating between the two vectors. Linear interpolation would only give an impression of mixing, as the peaks in each spectrum would appear in the result scaled by the interpolation factor. A more interesting method was proposed in [15] based on optimal transport. This algorithm implements displacement interpolation [30], which has been applied previously to computer graphics [31]. The framework of optimal transport—originally dealing with probability distributions—applies naturally to NMF bases since they are typically normalized during the computation, so that $\sum w_n = 1$.

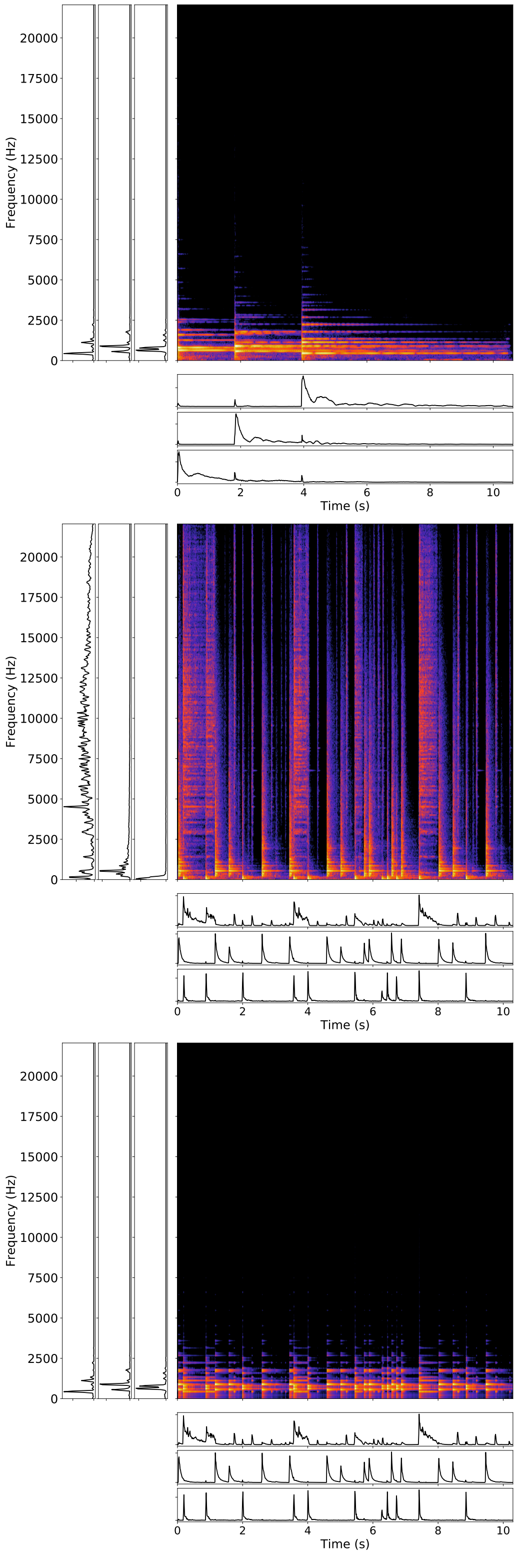

Figure 1: *NMF-based cross-syntnesis. Top: piano, middle: drum sound, bottom: result*

Optimal transport is commonly introduced as a problem of mapping between two probability distributions [13]. For 1D discrete distributions, the problem can be described as finding a plan γ that minimizes the cost of moving from a distribution A to a distribution B:

$$\min_{\gamma} \sum_{x} \sum_{y} c(x, y)\gamma(x, y). \tag{5}$$

Here x and y represent positions in the support of A and B respectively; $c(x, y)$ represents the cost of moving one particle of mass from location x to location y; and $\gamma(x, y)$ is a plan that specifies the connections between locations and the amount of mass to be carried for each path. The plan needs also to satisfy $\sum_y \gamma(x, y) = A$ and $\sum_x \gamma_(x, y) = B$, so that the mass of A is preserved and distributed into B. This can be seen as a linear program, which is relatively simple for 1D signals and closed cost. In this case, the domain is the frequency axis, and the cost is defined as $c(x, y) = ||x - y||^2$. This defines a metric space where a continuous path exists between both distributions.

In [15], the problem was defined for audio spectra by considering the fact that a spectral peak corresponding to a given frequency component in the original signal is represented in the STFT by a range of frequency bins. Thus, each peak is treated as a "spectral mass" and interpolated separately. In [15] this is done via spectrum reassignment [32], by cutting at the zero-crossings of the curve of reassigned frequencies of the bins. In this paper we interpolate between prototype spectral bases for which we don't have the original phase, but the displacement of the spectral peak as a unit is still of interest. Thus we simply segment the spectrum at the minima between peaks. In this setting, locations x and y in eq. 5 can be replaced by x_i and y_i, so that x_i represents the frequency associated to the center bin of mass i in the magnitude spectrum A. The corresponding mass is computed as the sum of the region of the magnitude spectrum corresponding to peak i.

The optimal plan is then represented by the matrix $\Gamma(i, j)$. Following [15], the matrix is constructed using the north-west corner rule, which is a commonly used heuristic for transportation problems [33]: starting from $(0, 0)$, the smaller of the two masses is placed at i, j and consumed from the other mass. The index of the depleted mass is incremented. The process continues until all mass has been consumed. The matrix is thus zero in most entries, and loosely follows a diagonal depending on the balance of masses across frequency.

The matrix $\Gamma(i, j)$ also provides a basis for the Wasserstein distance defined over the spectral masses at x_i and y_j:

$$d(A, B) = (\sum_{i} \sum_{j} ||x_i - y_j||^2 \Gamma(i, j))^{1/2} \tag{6}$$

This allows us to define the distance between the columns of W^s and the columns of W^t in eq. 3 as the minimum cost of transporting the mass of the peaks from the source to the target NMF basis across the frequency axis.

Displacement interpolation is then accomplished by sliding through the non-zero entries of the transport matrix: given an interpolation parameter λ, each pair of masses in the matrix are interpolated to $(1 - \lambda)x_i + \lambda y_i$ and added to the output spectrum. Figure 2 shows 10 interpolation steps between one base corresponding to a piano chord and one corresponding to a drum sound.

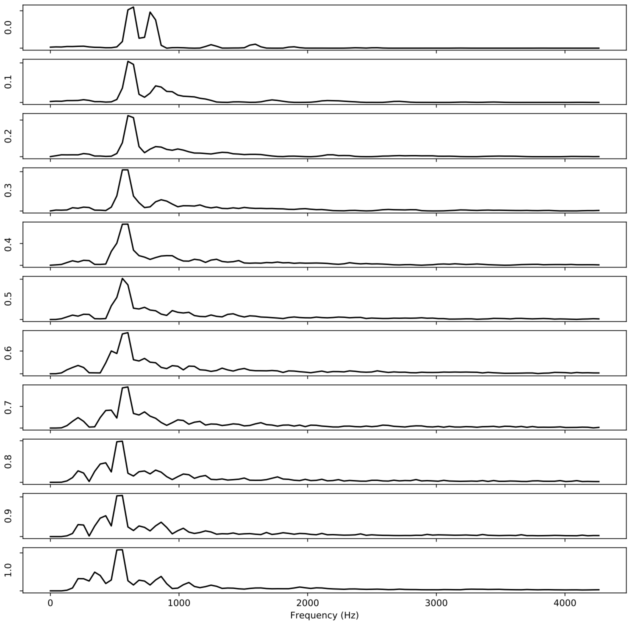

Figure 2: *Displacement interpolation between NMF bases of a piano (top) and a drum (bottom) ($\lambda = 0, 0.1...1$)*

5. HPSS PRE-PROCESSING

Interpolation between NMF bases offers an interesting possibility as long as the NMF decomposition represents a meaningful description of both spectrograms. In the above example, by having a separate interpolation parameter for each drum sound, we could choose to independently move each one towards the corresponding piano chord. However, in the general case we do not have such a clear mapping. Most often a large rank results in better sound quality, making it impractical to interpolate each component independently.

We thus introduce a pre-processing step to obtain signal components that are perceptually relevant by applying the popular harmonic / percussive source separation (HPSS) method by Fitzgerald [34] based on median filtering. Both V_s and V_t are decomposed into harmonic and percussive components, and the NMF process is performed independently for each component. This also helps the matching of similar components between V_s and V_t, at the expense of additional NMF computations (which can be run in parallel). As a result, our algorithm allows the use of one interpolation parameter for all the harmonic components, and another one for all the percussive components. This is similar to the perceptual interpolation space proposed in [1], but here using the median filter instead of MFCC-based smoothing. The basic morphing algorithm is summarized in Figure 3, while the HPSS version is shown in Figure 4.

6. PHASE GENERATION

Either through the basic or the HPSS variants, the algorithm ends up with the synthetic magnitude spectrogram of the morphed sound. In order to obtain a time domain waveform, the corresponding phase spectrogram needs to be synthesizeed as well. We accomplish this on a frame-by frame basis using Phase Gradient Heap

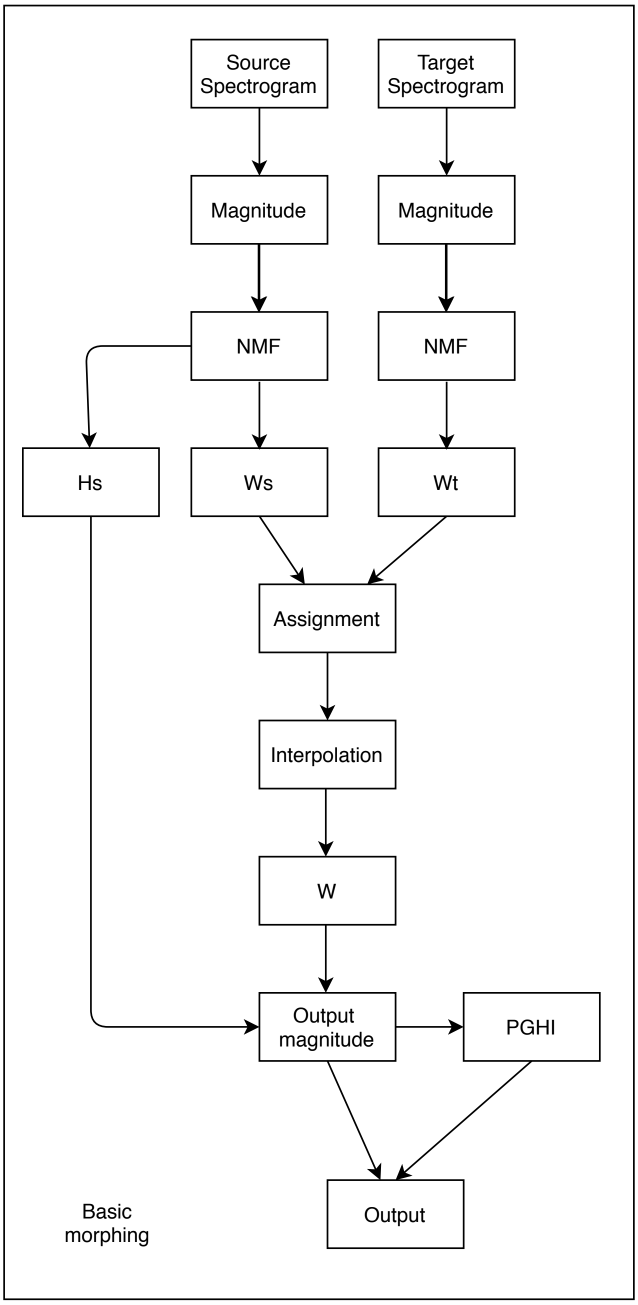

Figure 3: *Block diagram for the basic morphing algorithm*

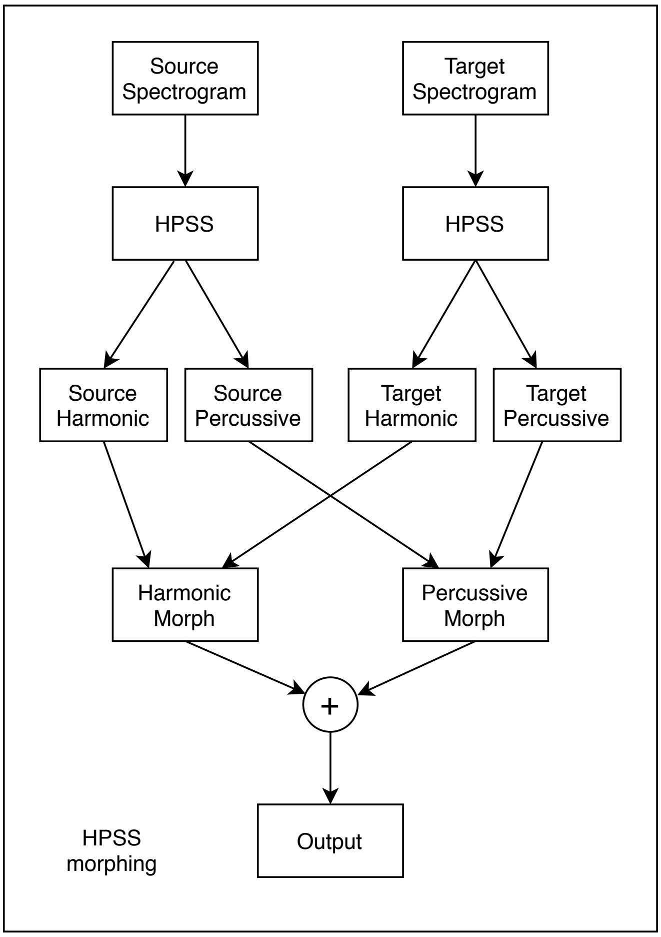

Figure 4: *Block diagram for the HPSS-based morphing algorithm*

Integration (PGHI) [35]. PGHI approximates the phase from a magnitude spectrogram by leveraging a theoretical result showing that the STFT phase has an algebraically expressible relationship to the derivative of the log magnitude spectrum. Even though this result only holds in the continuous domain using a Gaussian window with infinite support, the authors in [35] show that an acceptable approximation is possible in the discrete domain using non-Gaussian windows with finite support. The algorithm seeks to reconstruct the phase by focusing first on the ridges of the magnitude spectrogram (i.e. the points with the greatest energy) by applying the integration using a heap structure that places the bins with peaks at the top, before proceeding to estimate the phase for the surrounding bins. The real-time extension of this algorithm [24] makes it possible to estimate the phase from frame to frame, incurring only a single frame's worth of latency or, at the cost of some extra error, no additional latency at all. Since the displacement interpolation algorithm can run in real time, each frame of the morphed spectrogram is computed according to the interpolation parameter, and then the phase for that frame is computed accordingly in real time.

7. RESULTS AND DISCUSSION

In order to assess the proposed approach, we have implemented it in a Python library, which allows experimenting with the main variants in an offline fashion. In order to test the real-time morphing, we have implemented an object for the Max patching language[2], which we plan to introduce in the Fluid Corpus Manipulation Toolbox[3] [36]. Both the Python and the Max implementations can be downloaded from the companion website of this

[2] `https://blockding74.com`
[3] `https://www.flucoma.org`

paper[4]. The real-time implementation performs an offline analysis phase which computes the HPSS and NMF decompositions and the matching of bases. It is then possible to morph between W^t and W^s for both the harmonic and the percussive components of the target signal, while the sound is playing according to the activations in H^t.

With respect to the basic cross-synthesis algorithm in [10], our system introduces several novel contributions: the automatic rank; the automatic assignment with the Hungarian algorithm; the use of real-time PGHI for phase synthesis; the extension to morphing; and the HPSS pre-processing. We tested the system with a variety of signals, generally in the order of a few seconds, including tonal and percussive material and polyphonic mixtures, in order to assess these contributions. Some examples can be heard in the companion website.

With respect to the rank, we noted that in general it is useful to keep the ability to define it manually when the material contains clear melodies or patterns, so each pitch or spectral pattern can be captured for an NMF base. However, for more complex signals such as polyphonic mixtures, low ranks result in a characteristic grainy sound, arising from the reduction of information. Also, as with the original approach in [15], interpolation between bases representing pitches is affected by the discrete nature of the interpolation, so the displacement of pitches is quantized to frequency bins. This can be partially alleviated by zero-padding the spectrum by a sufficient amount. Using the automatic rank allows us to get good sounding results in the general case, where no assumptions are made about the content of the signal. The parameter p (controlling the rank) can be used as a tradeoff between sound quality and computational cost.

With respect to the automatic assignment, we observed that while interesting results could be obtained with arbitrary mappings, the proposed approach provided consistent results for larger ranks. In general, using a lower rank for W^t than W^s and thus repeating bases for different activations could lead to very poor results, which motivated the restriction proposed in Section 3. The automation also helps giving a predictable outcome given the random initialization of NMF. In addition to the Wasserstein distance in eq. 6, we obtained interesting results with the symmetrized KL divergence between spectral bases. In general, more parameters for controlling the assignment of bases are interesting for experimenting with and obtaining new sounds.

With respect to PGHI, we noted that the sound quality is generally improved with respect to using GL, especially in the harmonic component. Both algorithms provide good approximations for existing sounds, but inventing a phase for synthetic sounds is more of a challenge. Using a large rank tends to result in a good magnitude reconstruction, which also helps with obtaining a good phase estimate. This can be noted when $\lambda = 0$, i.e. when the morphing algorithm reconstructs the original spectrogram V^s with a synthetic phase estimate.

Real-time synthesis of the phase also enables real-time morphing, as the corresponding magnitude frame in $\hat{V}$ can be computed by multiplying the current frame in H^t with the interpolated spectral base. This feature showed great creative potential and allows, for example, the automation of dynamic timbre movements according to different rhythms. Since the NMF decomposition takes into account activations over time, the effect of the parameters may not be heard immediately in some cases. In practice, of course, real-time operation allows for a more intuitive control and facilitates quick experimentation, as well as automation and modulation of the interpolation parameters. The target sample can thus be applied to the source sample as if it was an effect. One limitation of the proposed approach is that the interpolation is linear in frequency. This results in a noticeable exponential effect of the interpolation parameter. In this sense it would be interesting to apply the proposed approach to an invertible constant-Q transform such as [37].

Finally, the use of HPSS as a preprocessing step further increases the possibilities of the morphing approach, while often resulting in more traditional sonorities with emphasized percussive and tonal components. Splitting the match of NMF bases into harmonic and percussive components generally results in more natural sounding morphs, particularly as it avoids the repetition of noisy patterns as if they were stationary. The introduction of the HPSS decomposition also introduces more parameters that can be used to improve the result, such as the size of the median filters used in the decomposition, as well as separate parameters of the morphing algorithm for each of the components. The results of each component can also be remixed with arbitrary gains. The HPSS variant of the algorithm also introduces a significant increase of the computational cost of the analysis stage. In our experience, the most valuable aspect of this variant is the ability to morph between the harmonic components of the source and target spectrograms. The percussive part is made of short transients which are often perceived similarly for both sounds. When the source spectrogram contains sharp transients (e.g. for rhythmic material) it is sometimes convenient to use the percussive part from the source spectrogram, in which case the NMF and morphing computations for the percussive part be switched off.

8. CONCLUSIONS

Beyond source separation and transcription, NMF decomposition of spectrograms provides a useful framework for creative applications based on transformation of sounds containing temporal variations of spectral patterns. In this paper we have presented an application to continuous morphing, leveraging the application of optimal transport for audio spectra. With respect to the system presented in [15], which works at a frame level, the application to the NMF cross-synthesis allows for flexible morphing of audio in multiple dimensions. In addition, we have shown how further decomposition of the spectrogram using HPSS can be used to provide a more intuitive interface.

Our approach can be seen as a framework into which other decompositions of the spectrogram could be plugged, allowing for further granularity. As such, the combination of matrix decompositions with displacement interpolation offers a promising environment for audio transformation that affords intuitive interfaces and relatively low computational requirements. As an example, our approach has been implemented on a popular computer music system, which allows its use in current creative workflows.

9. ACKNOWLEDGMENTS

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 725899).

[4] https://www.flucoma.org/DAFX-2020

10. REFERENCES

[1] M. Slaney, M. Covell, and B. Lassiter, "Automatic audio morphing," in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, 1996, vol. 2, pp. 1001–1004.

[2] Edwin Tellman, Lippold Haken, and Bryan Holloway, "Timbre morphing of sounds with unequal numbers of features," *Journal of the Audio Engineering Society*, vol. 43, no. 9, pp. 678–689, 1995.

[3] Marcelo Freitas Caetano and Xavier Rodet, "Automatic timbral morphing of musical instrument sounds by high-level descriptors," in *Proceedings of 2010 International Computer Music Conference. Computer Music Assosiation.*, 2010, pp. 11–21.

[4] Zynaptiq GmbH, "Zynaptiq Morph," Available at https://www.zynaptiq.com/morph/, accessed August 2020.

[5] MeldaProduction, "MeldaProduction MMorph," Available at https://www.meldaproduction.com/MMorph, accessed August 2020.

[6] Xavier Serra, *A System for Sound Analysis / Transformation / Synthesis based on a Deterministic plus Stochastic Decomposition*, Ph.D. thesis, Stanford University, 1989.

[7] Federico Boccardi and Carlo Drioli, "Sound morphing with gaussian mixture models," in *Proceedings of the 2001 International Conference on Digital Audio Effects (DAFx-01)*, 2001, pp. 44–48.

[8] Naotoshi Osaka, "Timbre interpolation of sounds using a sinusoidal model," in *Proceedings of 1995 International Computer Music Conference*, 1995.

[9] Nick Collins and Bob L Sturm, "Sound cross-synthesis and morphing using dictionary-based methods," in *Proceedings of the 2011 International Computer Music Conference*, 2011.

[10] Juan José Burred, "Cross-synthesis based on spectrogram factorization," in *Proceedings of the 2013 International Computer Music Conference*, 2013.

[11] Jonathan Driedger, Thomas Prätzlich, and Meinard Müller, "Let it bee — towards nmf-inspired audio mosaicing.," in *Proceedings of the 16th International Society for Music Information Retrieval Conference*, 2015, pp. 350–356.

[12] Martin Arjovsky, Soumith Chintala, and Léon Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning*, 2017, pp. 214–223.

[13] S. Kolouri, S. R. Park, M. Thorpe, D. Slepcev, and G. K. Rohde, "Optimal mass transport: Signal processing and machine-learning applications," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 43–59, 2017.

[14] Rémi Flamary, Cédric Févotte, Nicolas Courty, and Valentin Emiya, "Optimal spectral transportation with application to music transcription," in *Advances in Neural Information Processing Systems*, 2016, pp. 703–711.

[15] Trevor Henderson and Justin Solomon, "Audio transport: A generalized portamento via optimal transport," *Proceedings of the 2019 International Conference on Digital Audio Effects (DAFx-19)*, 2019.

[16] Eric Grinstein, Ngoc QK Duong, Alexey Ozerov, and Patrick Pérez, "Audio style transfer," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 586–590.

[17] Maciek Tomczak, Carl Southall, and Jason Hockman, "Audio style transfer with rhythmic constraints," in *Proceedings of the 2018 International Conference on Digital Audio Effects (DAFx-18)*, 2018, pp. 45–50.

[18] Hugo Caracalla and Axel Roebel, "Sound texture synthesis using convolutional neural networks," *Proceedings of the 2019 International Conference on Digital Audio Effects (DAFx-19)*, 2019.

[19] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan, "Neural audio synthesis of musical notes with wavenet autoencoders," in *International Conference on Machine Learning*, 2017, pp. 1068–1077.

[20] Philippe Esling, Adrien Bitton, et al., "Generative timbre spaces: regularizing variational auto-encoders with perceptual metrics," *Proceedings of the 2018 International Conference on Digital Audio Effects (DAFx-18)*, pp. 369–376, 2018.

[21] Paris Smaragdis and Judith C Brown, "Non-negative matrix factorization for polyphonic music transcription," in *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2003, pp. 177–180.

[22] Cédric Févotte, Emmanuel Vincent, and Alexey Ozerov, "Single-channel audio source separation with NMF: Divergences, constraints and algorithms," in *Audio Source Separation*, Shoji Makino, Ed., pp. 1–24. Springer International Publishing, 2018.

[23] Daniel Griffin and Jae Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[24] Zdeněk Průša and Peter L. Søndergaard, "Real-Time Spectrogram Inversion Using Phase Gradient Heap Integration," in *Proceedings of the 2016 International Conference on Digital Audio Effects (DAFx-16)*, 2016, pp. 17–21.

[25] E. Quinton M. Buch and B. L. Sturm, "Nicht-negativematrixfaktorisierungnutzendesklangsynthesensystem (nimfks): Extensions of nmf-based concatenative sound synthesis," in *Proceedings of the 2017 International Conference on Digital Audio Effects (DAFx-17)*, 2017.

[26] Patricio López-Serrano, Christian Dittmar, Yigitcan Özer, and Meinard Müller, "NMF toolbox: Music processing applications of nonnegative matrix factorization," in *Proceedings of the 2019 International Conference on Digital Audio Effects (DAFx-19)*, 2019.

[27] Hanli Qiao, "New SVD based initialization strategy for non-negative matrix factorization," *Pattern Recognition Letters*, vol. 63, pp. 71–77, 2015.

[28] Arthur Flexer, Dominik Schnitzer, Martin Gasser, and Tim Pohle, "Combining features reduces hubness in audio similarity.," in *Proceedings of the 2010 International Conference on Sound and Music Computing*, 01 2010, pp. 171–176.

[29] Julian Neri and Philippe Depalle, "Fast partial tracking of audio with real-time capability through linear programming," in *Proceedings of the 2018 International Conference on Digital Audio Effects (DAFx-18)*, 2018, pp. 326–333.

[30] Robert J McCann, "A convexity principle for interacting gases," *Advances in mathematics*, vol. 128, no. 1, pp. 153–179, 1997.

[31] Nicolas Bonneel, Michiel Van De Panne, Sylvain Paris, and Wolfgang Heidrich, "Displacement interpolation using Lagrangian mass transport," *ACM Transactions on Graphics*, vol. 30, no. 6, pp. 1–12, 2011.

[32] P. Flandrin, F. Auger, and E. Chassande-Mottin, "Time-Frequency Reassignment: From Principles to Algorithms," in *Applications in Time-Frequency Signal Processing*, Antonia Papandreou-Suppappola, Ed., pp. 179–204. CRC Press, 2002.

[33] Gabriel Peyré, Marco Cuturi, et al., "Computational optimal transport: With applications to data science," *Foundations and Trends in Machine Learning*, vol. 11, no. 5-6, pp. 355–607, 2019.

[34] Derry Fitzgerald, "Harmonic/percussive separation using median filtering," *Proceedings of the 2010 International Conference on Digital Audio Effects (DAFx-10)*, 2010.

[35] Zdeněk Průša, Peter Balazs, and Peter Lempel Søndergaard, "A noniterative method for reconstruction of phase from stft magnitude," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 1154–1164, 2017.

[36] Pierre Alexandre Tremblay, Owen Green, Gerard Roma, and Alexander Harker, "From collections to corpora: Exploring sounds through fluid decomposition," in *Proceedings of the 2019 International Computer Music Conference*, 2019.

[37] Nicki Holighaus, Monika Dörfler, Gino Angelo Velasco, and Thomas Grill, "A Framework for Invertible, Real-Time Constant-Q Transforms," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 775–785, 2013.