Cover Song Synthesis By Analogy

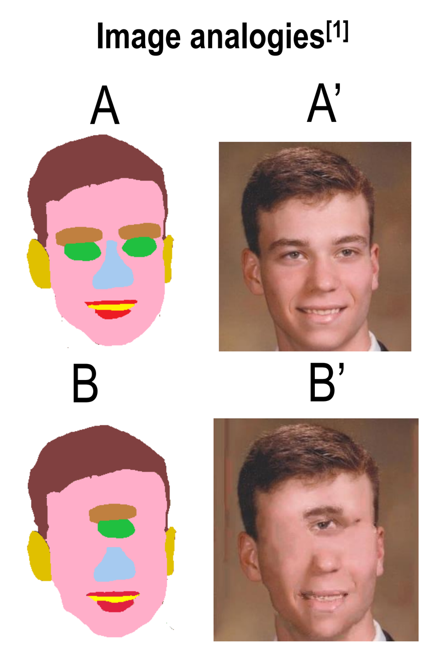
Christopher J. Tralie. ctralie@alumni.princeton.edu
Department of Mathematics, Duke University

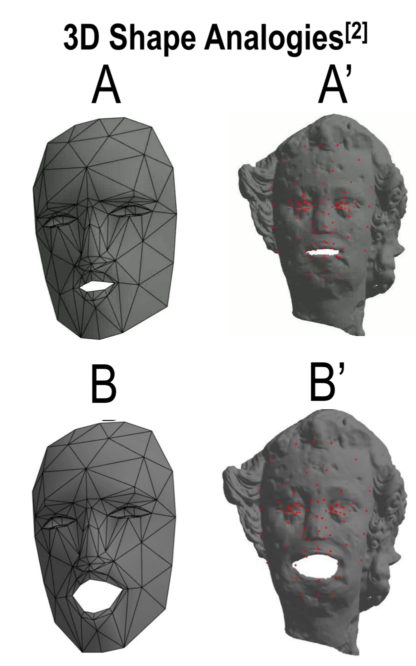
Problem Statement

GOAL: Given polyphonic audio by artist 1, re-synthesize it in the style of artist 2.

MAIN APPROACH: Use a cover song example pair of another song A by artist 1 with cover A' by artist 2 to constrain the problem. Learn instrument transformations from artist 1 to artist 2 and apply them to a new song B by artist 1.

INSPIRED BY: Image analogies, 3D shape analogies





Results And Code

SUPPLEMENTARY MATERIAL

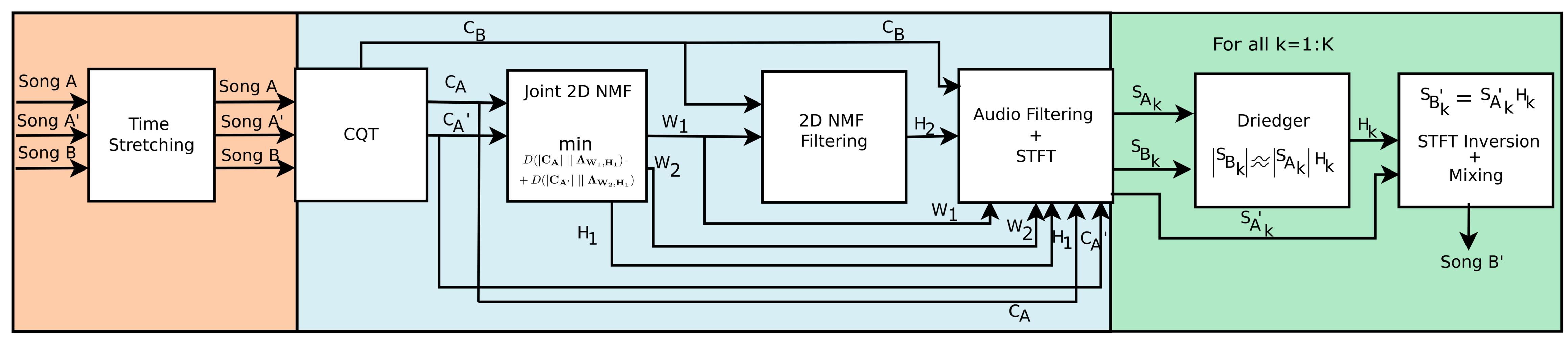
- **Synchronized** cover songs **A** and **B**
- **Synthesized** songs **B'**
- **Translation dictionary** elements **W** converted to audio with Griffin Lim
- **Filtered audio components for each track**

http://www.covers1000.net/analogies.html

CODE (Work in progress, but 2D NMF is solid)

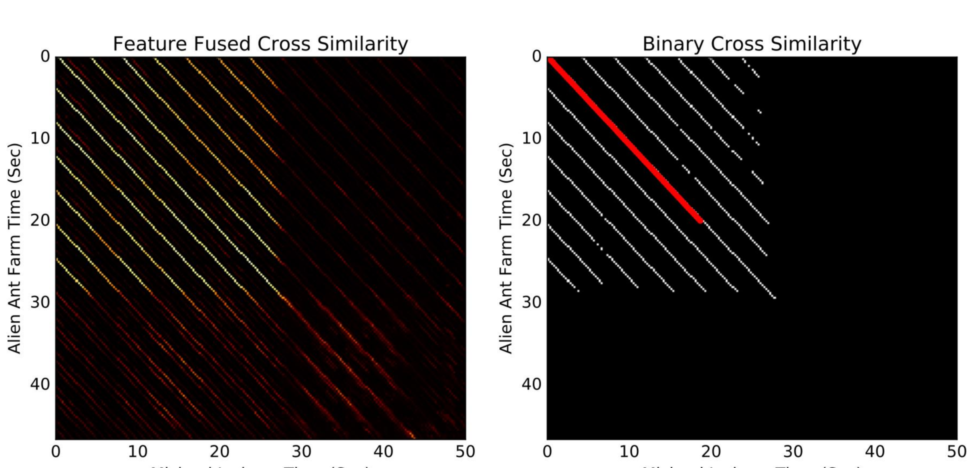
https://github.com/ctralie/CoverSongSynthesis

Full Pipeline

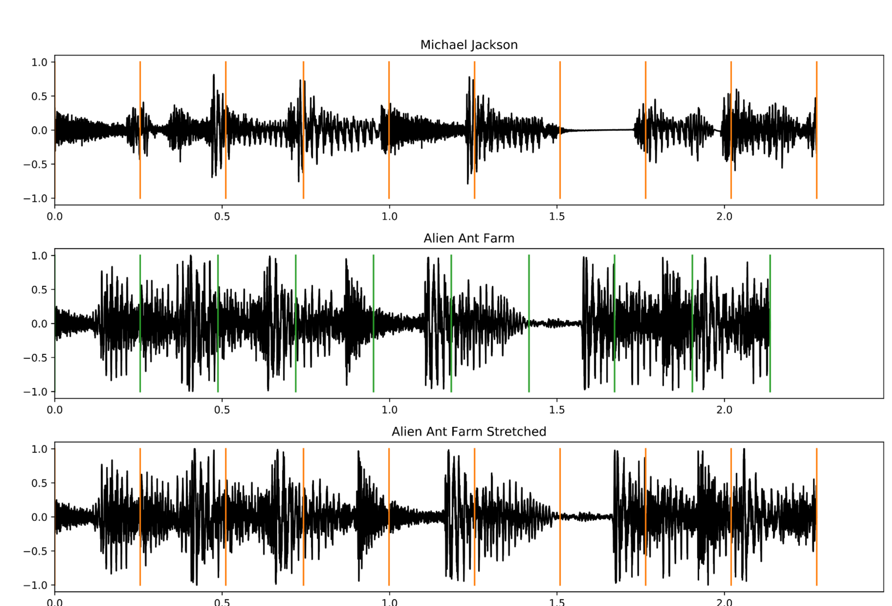


Aligning And Stretching

- Use our upstream feature fusion technique from [4] for accurate beat-level synchronization between cover song example pair



- Perform beat by beat uniform rescaling using the Rubberband library[9]
- The result is that A is aligned to A'



DICTIONARY LEARNING: Joint Filtering with 2D Convolutional NMF

- Main technique to learn transformations, based on work in [5]
- **Ws**: Time-frequency template snippets for different instruments
- $\mathbf{W_1}$: artist 1, $\mathbf{W_2}$: artist 2
- **Hs**: Activations/frequency shifts over time
- Learn different Ws for each song, but **share the same H**

$$\mathbf{X} \approx \mathbf{\Lambda}_{\mathbf{W},\mathbf{H}} = \sum_{\tau=1}^{T} \sum_{\phi=1}^{F} \overset{\downarrow\phi}{\mathbf{W}^{\tau}} \overset{\rightarrow\tau}{\mathbf{H}^{\phi}}$$

$$\mathbf{W}^{\tau} \in \mathbb{R}^{M \times K} \quad \mathbf{H}^{\phi} \in \mathbb{R}^{K \times N}$$

- Perform decomposition on **magnitude CQTs[8]** $\mathbf{C_A}, \mathbf{C_{A'}} \in \mathbb{C}^{M \times N_1}$
- *Minimize KL Divergence* $D(|\mathbf{C_A}| \;||\; \mathbf{\Lambda}_{\mathbf{W_1},\mathbf{H_1}}) + D(|\mathbf{C_{A'}}| \;||\; \mathbf{\Lambda}_{\mathbf{W_2},\mathbf{H_1}})$
- $\mathbf{W_1}$ artist 1 instrument templates, $\mathbf{W_2}$ artist 2 instrument templates
- $\mathbf{H_1}$ can be thought of as a *musical score that's shared between songs*
- Iterative update rules below given CQTs
- *Implemented on GPU in pyCUDA for speed*

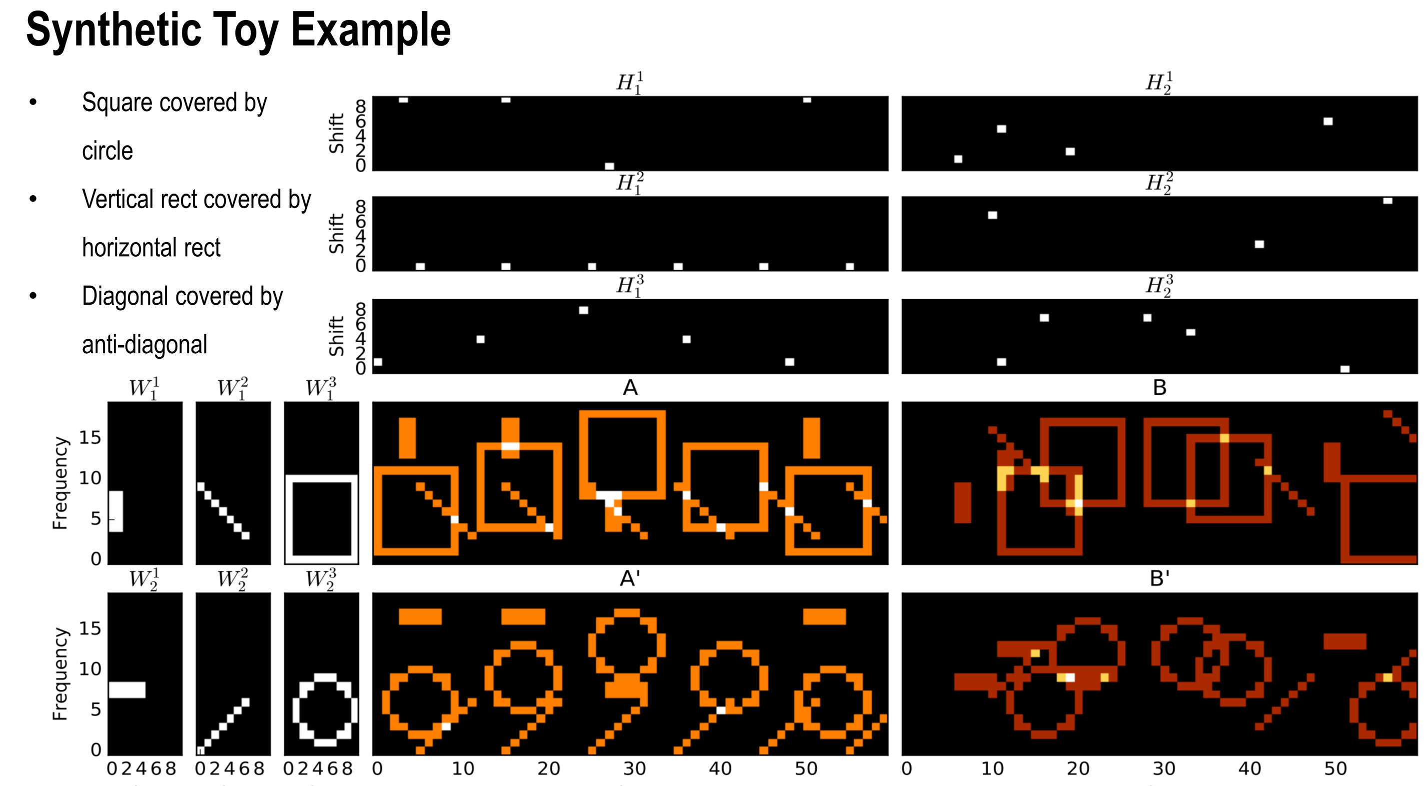
$$\mathbf{W_1^{\tau}} \leftarrow \mathbf{W_1^{\tau}} \odot \frac{\sum_{\phi=1}^{F} \overset{\uparrow\phi}{\left(\frac{|\mathbf{C_A}|}{\mathbf{\Lambda}_{\mathbf{W_1},\mathbf{H_1}}}\right)} \overset{\rightarrow\tau}{\mathbf{H_1^{\phi}}}^{T}}{\sum_{\phi=1}^{F} \mathbf{1} \cdot \overset{\rightarrow\tau}{\mathbf{H_1^{\phi}}}^{T}}$$

$$\mathbf{W_2^{\tau}} \leftarrow \mathbf{W_2^{\tau}} \odot \frac{\sum_{\phi=1}^{F} \overset{\uparrow\phi}{\left(\frac{|\mathbf{C_{A'}}|}{\mathbf{\Lambda}_{\mathbf{W_2},\mathbf{H_1}}}\right)} \overset{\rightarrow\tau}{\mathbf{H_1^{\phi}}}^{T}}{\sum_{\phi=1}^{F} \mathbf{1} \cdot \overset{\rightarrow\tau}{\mathbf{H_1^{\phi}}}^{T}}$$

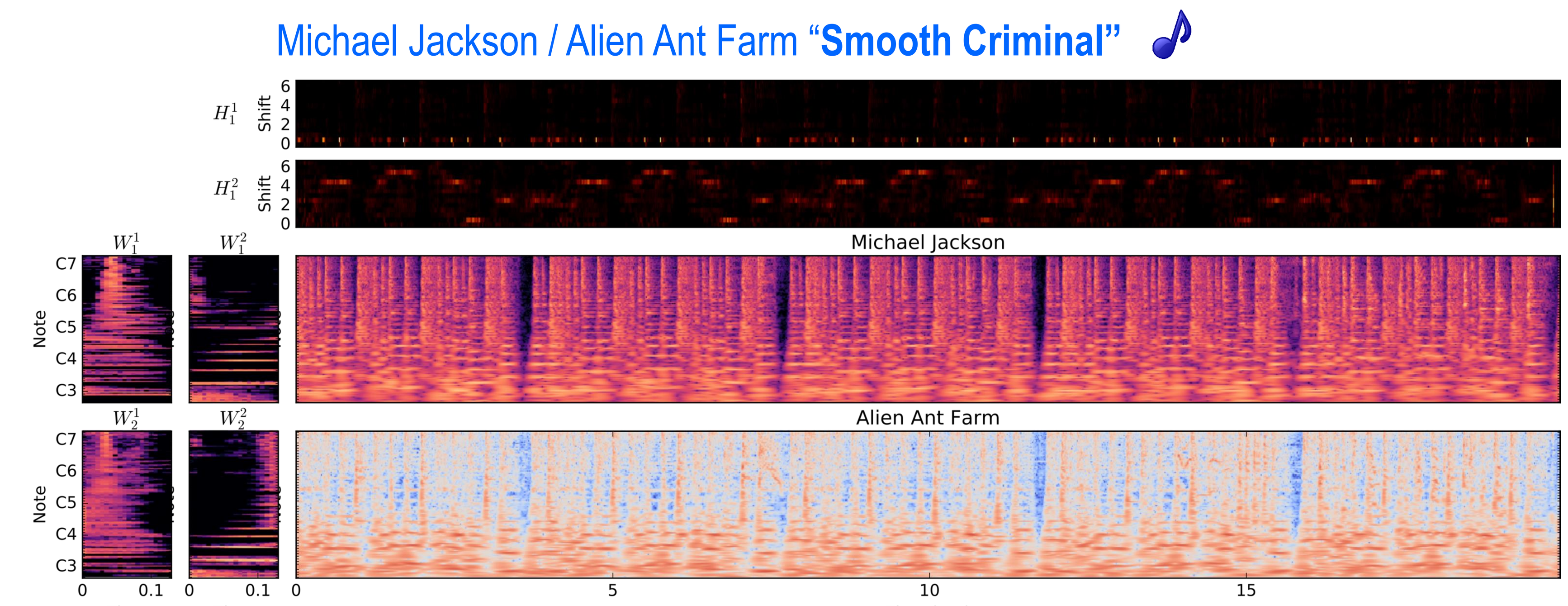
$$\mathbf{H_1^{\phi}} \leftarrow \mathbf{H_1^{\phi}} \odot \left(\frac{\sum_{\tau=1}^{T} \overset{\downarrow\phi}{\mathbf{W_1^{\tau}}}^{T} \overset{\leftarrow\tau}{\left(\frac{|\mathbf{C_A}|}{\mathbf{\Lambda}_{\mathbf{W_1},\mathbf{H_1}}}\right)} + \overset{\downarrow\phi}{\mathbf{W_2^{\tau}}}^{T} \overset{\leftarrow\tau}{\left(\frac{|\mathbf{C_{A'}}|}{\mathbf{\Lambda}_{\mathbf{W_2},\mathbf{H_1}}}\right)}}{\sum_{\tau=1}^{T} \overset{\downarrow\phi}{\mathbf{W_1^{\tau}}}^{T} \overset{\leftarrow\tau}{\mathbf{1}} + \overset{\downarrow\phi}{\mathbf{W_2^{\tau}}}^{T} \overset{\leftarrow\tau}{\mathbf{1}}} \right)$$

Synthetic Toy Example

- Square covered by circle
- Vertical rect covered by horizontal rect
- Diagonal covered by anti-diagonal



Michael Jackson / Alien Ant Farm "**Smooth Criminal**"



Applying Filters To B And Separating Audio

- Fix $\mathbf{W_1}$, learn activations $\mathbf{H_2}$ for **B**

$$\mathbf{H_2^{\phi}} \leftarrow \mathbf{H_2^{\phi}} \odot \left(\frac{\sum_{\tau=1}^{T} \overset{\downarrow\phi}{\mathbf{W_1^{\tau}}}^{T} \overset{\leftarrow\tau}{\left(\frac{|\mathbf{C_B}|}{\mathbf{\Lambda}_{\mathbf{W_1},\mathbf{H_2}}}\right)}}{\sum_{\tau=1}^{T} \overset{\downarrow\phi}{\mathbf{W_1^{\tau}}}^{T} \overset{\leftarrow\tau}{\mathbf{1}}} \right)$$

- Apply Wiener filters to complex CQTs to obtain **k** separate audio track CQTs
- Invert CQTs back to audio domain, end up with k pairs of tracks from A and A', each associated to one of k tracks from B

$$\mathbf{C_{A_k}} = \mathbf{C_A} \odot \left(\frac{\mathbf{\Lambda}^{\mathbf{p}}_{\mathbf{W_1},\mathbf{H_1},\mathbf{k}}}{\sum_{m=1}^{K} \mathbf{\Lambda}^{\mathbf{p}}_{\mathbf{W_1},\mathbf{H_1},\mathbf{m}}} \right)$$

$$\mathbf{C_{A'_k}} = \mathbf{C_{A'}} \odot \left(\frac{\mathbf{\Lambda}^{\mathbf{p}}_{\mathbf{W_2},\mathbf{H_1},\mathbf{k}}}{\sum_{m=1}^{K} \mathbf{\Lambda}^{\mathbf{p}}_{\mathbf{W_2},\mathbf{H_1},\mathbf{m}}} \right)$$

$$\mathbf{C_{B_k}} = \mathbf{C_B} \odot \left(\frac{\mathbf{\Lambda}^{\mathbf{p}}_{\mathbf{W_1},\mathbf{H_2},\mathbf{k}}}{\sum_{m=1}^{K} \mathbf{\Lambda}^{\mathbf{p}}_{\mathbf{W_1},\mathbf{H_2},\mathbf{m}}} \right)$$

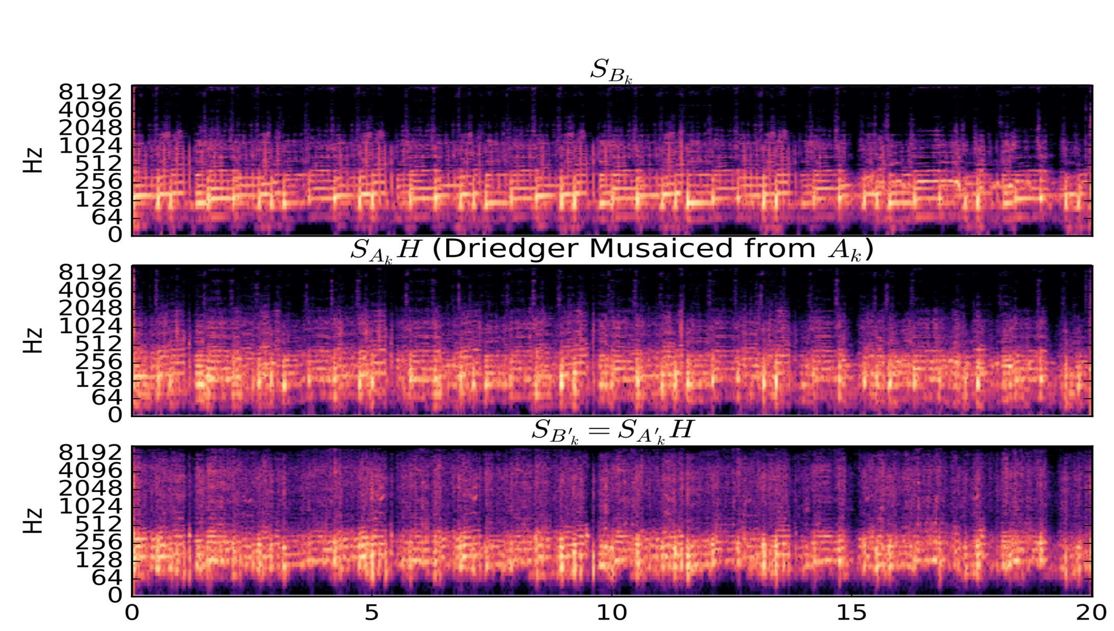
Musaicing And Mixing

- Use Driedger's technique[6] to mash up STFT grains from song A to form song A' (STFT instead of CQT for memory reasons)

$$|\mathbf{S_{B_k}}| \approx |\mathbf{S_{A_k}}| \mathbf{H}_k$$

- Use the activations with A' instead of A to create the synthesized cover song B'. This is the final step!

$$\mathbf{S_{B'_k}} = \mathbf{S_{A'_k}} \mathbf{H_k}$$



Qualitative Results

A	A'	B	B'
Michael Jackson "Smooth Criminal"	Alien Ant Farm "Smooth Criminal"	Michael Jackson "Bad"	**Alien Ant Farm "Bad"**
Michael Jackson "Smooth Criminal"	Alien Ant Farm "Smooth Criminal"	Michael Jackson "Wanna Be Startin Something"	**Alien Ant Farm "Wanna Be Startin Something"**
Eurythmics "Sweet Dreams"	Marilyn Manson "Sweet Dreams"	Eurythmics "Who's That Girl"	**Marilyn Manson "Who's That Girl"**

- MJ -> AAF, synth guitar -> electric guitar
- Eurythmics -> Marilyn Manson, synth keyboard -> electric guitar

References

[1] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. InProceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 327–340. ACM, 2001.
[2] Robert W Sumner and Jovan Popovi´c. Deformation transfer for triangle meshes. In ACM Transactions on Graphics (TOG), volume 23, pages 399–405. ACM, 2004
[3] Ian Simon, Sumit Basu, David Salesin, and Maneesh Agrawala. Audio analogies: Creating new music from an existing performance by concatenative synthesis. In ICMC. Citeseer, 2005.
[4] Christopher J Tralie. Early mfcc and hpcp fusion for robust cover song identification. In 18th International Society for Music Information Retrieval (ISMIR), 2017.
[5] Mikkel N Schmidt and Morten Mørup. Nonnegative matrix factor 2-d deconvolution for blind single channel source separation. In International Conference on Independent Component Analysis and Signal Separation, pages 700–707. Springer, 2006.
[6] Jonathan Driedger, Thomas Pr¨atzlich, and Meinard M¨uller. Let it bee-towards nmf-inspired audio mosaicing. In ISMIR, pages 350–356, 2015.
[7] Hadrien Foroughmand and Geoffroy Peeters. Multisource musaicing using non-negative matrix factor 2-d deconvolution. In 18th International Society for Music Information Retrieval (ISMIR), Late Breaking Session, 2017.
[8] Gino Angelo Velasco, Nicki Holighaus, Monika D¨orfler, and Thomas Grill. Constructing an invertible constant-qtransformwithnon-stationarygaborframes. Proceedings of DAFX11, Paris, pages 93–99, 2011.
[9] C Cannam. Rubber band library. Software released under GNU General Public License (version 1.8. 1), 2012.

Please see our paper for a more complete list of references