# Pipe Organ Stop Identification via Non-Negative Matrix Factorization

Autumn Coe | ECE 477 Computer Audition

UNIVERSITY of ROCHESTER

## Abstract

- Most instrument identification methods (Cepstrum Coefficients, Source Filter Models) rely on the uniqueness of harmonic peak amplitudes.
- These methods are vulnerable to multiple instruments playing notes in unison or octaves due to the overlap of harmonic peaks.
- Since multiple stops are frequently turned on together, pipe organs frequently have different stops playing in unison or octaves.
- In this project, sparse NMF is used to take advantage of the additive linearity of the fourier transform.

## Objective

- Train a note dictionary with non-negative matrix factorization (NMF).
- Extract dictionary note activations in a sparse matrix using the LASSO algorithm.
- Identify stops by grouping activations by stop and thresholding the maximum activation value of each stop.
- Display results in an intuitive manner using a MatLab GUI

## Pipe Organ Overview

- Pipe organs contain multiple stops that imitate orchestral instruments. The main families are: Flutes, Principals, Strings and Reeds.
- Stops can be turned on individually or simultaneously so that multiple unison or octave notes will sound when a single key is pressed.
- Different stops can be selected for each keyboard.

## NMF

- NMF is used to decompose a spectrogram (*V*) into a dictionary of notes (*W*) and and activations (*H*) such that:

$$W \times H \approx V$$

- *W* and *H* are updated using euclidean distance multiplicative updates [1] such that:

$$W_{ia} \leftarrow W_{ia} \frac{(VH^T)_{ia}}{(WHH^T)_{ia}}$$

$$H_{a\mu} \leftarrow H_{a\mu} \frac{(W^T V)_{a\mu}}{(W^T WH)_{a\mu}}$$

## NMF Continued

- A note dictionary (*W*) is created for each stop with *a* equal to the number of notes in that stop. Dictionaries are trained on chromatic scales played on each stop individually.
- Individual stop dictionaries are concatenated to create a complete note dictionary.
- Just using NMF for stop identification produces an activation matrix that is too noisy as shown in fig. 1. To prevent this, a sparsity algorithm is employed.

## LASSO Algorithm

- To calculate the sparse activation matrix, the LASSO (least absolute shrinkage and selection operator) algorithm provided in MatLab [2] is used:

$$\min_{\beta_0,\beta}\left(\frac{1}{2N}\sum_{i=1}^{N}\left(y_i-\beta_0-x_i^T\beta\right)^2+\lambda\sum_{j=1}^{p}|\beta_j|\right)$$

  - $N$ is the number of observations
  - $y_i$ is the response at observation $i$
  - $x_i$ is a data vector of length p at observation i
  - $\lambda$ is a nonnegative regularization parameter
  - $\beta$ is a vector of length p

- The LASSO algorithm was applied to the spectrogram as follows:
  - $x = W$, the trained note dictionary
  - $y = V_\mu$, each frame of the spectrogram
  - $\beta = H_\mu$, each column of the activation matrix
- This resulted in a more accurate activation matrix as shown in fig. 1.

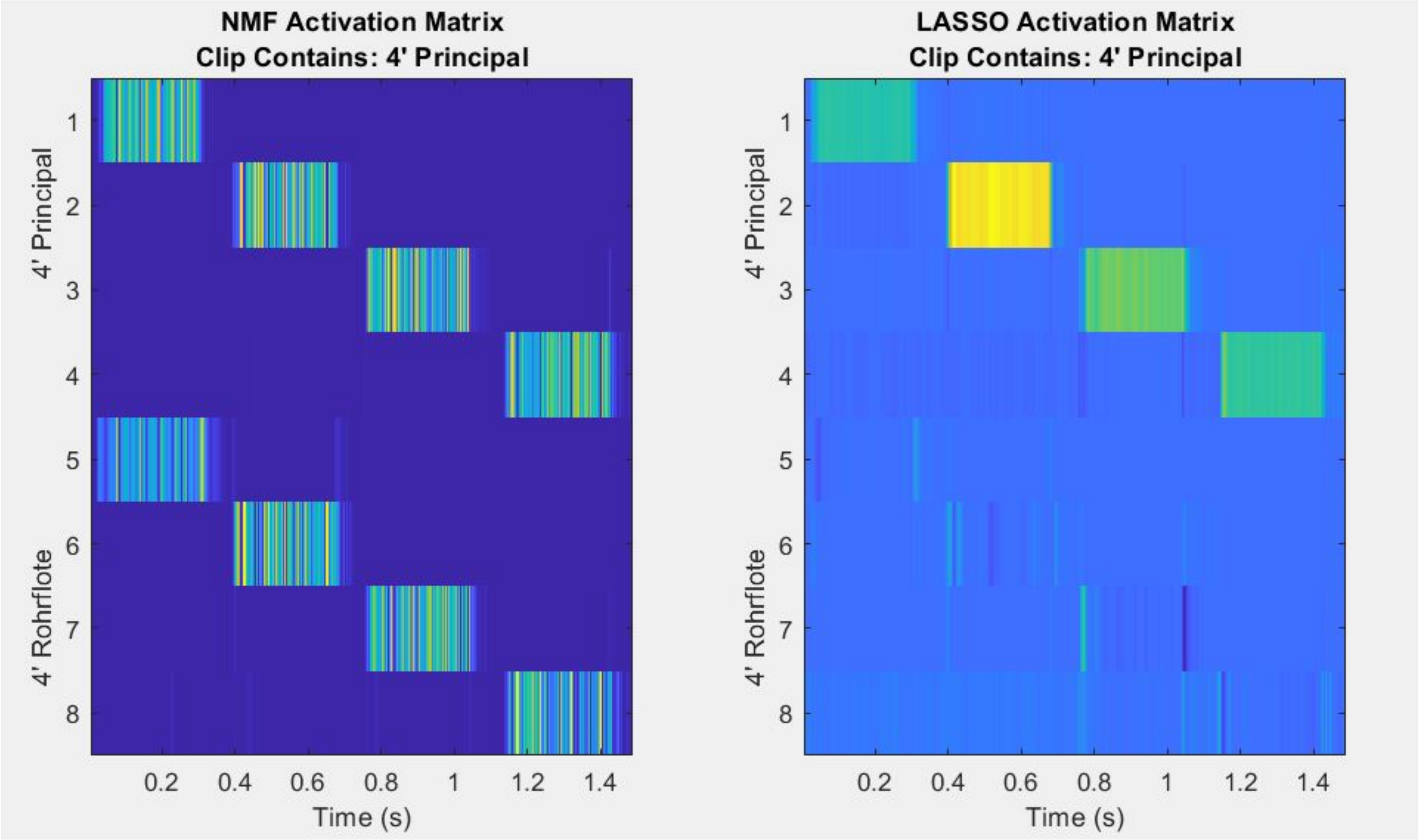


Figure 1 NMF vs LASSO activation matrix.

## Stop Identification

- The rows of the activation matrix are divided into sections representing each stop.
- To determine if a stop is being played, the maximum activation value is taken from each stop section and thresholded as shown in fig. 2.

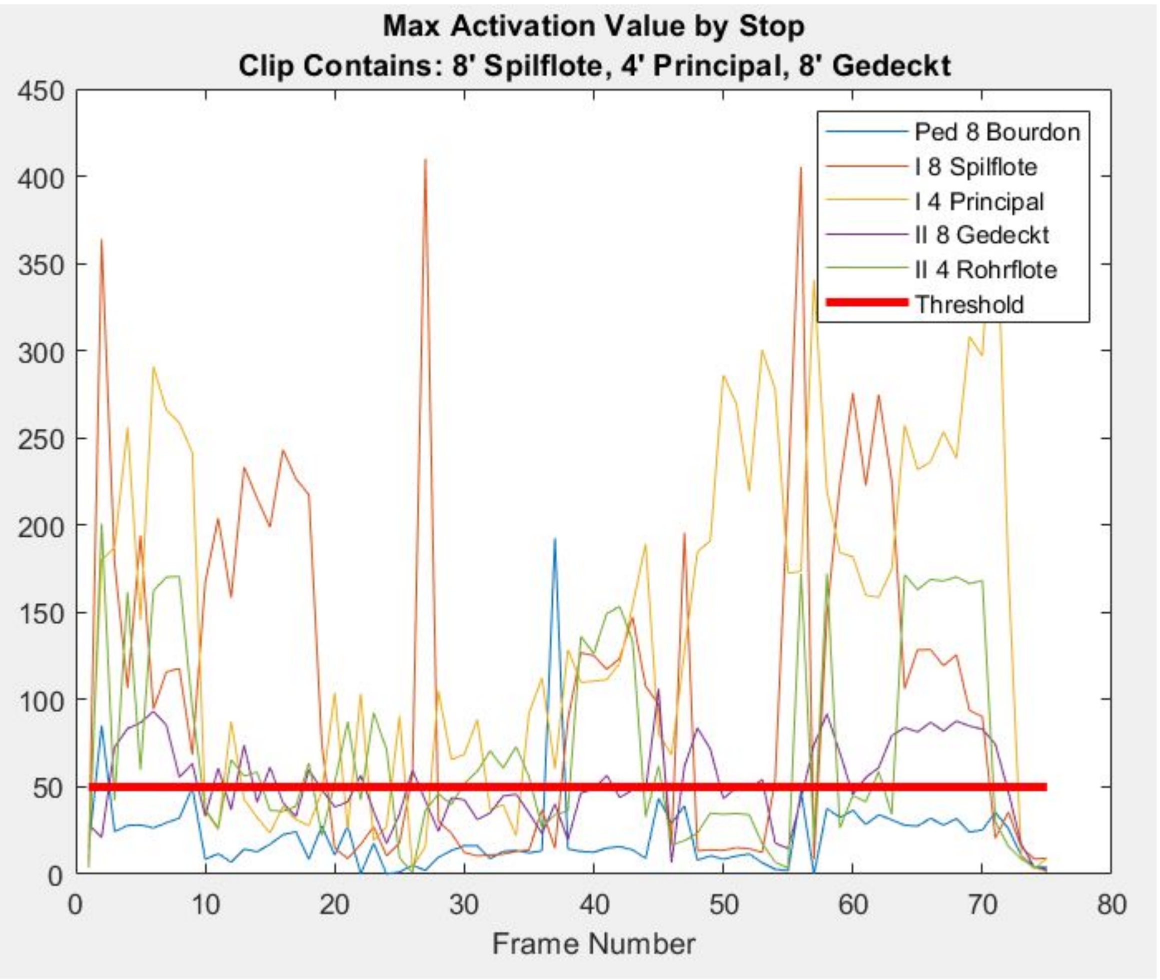


Figure 2: Maximum activations for each stop.

## Results and Future Work

- As shown in fig. 2, the accuracy is poor.This can be attributed to the difficulty of separating unison notes and the similarity of the stops on this organ.
- Even with the pretrained note dictionary, the LASSO algorithm is quite slow preventing the stop identification from running in real time.
- Future work includes improving the accuracy by refining the note dictionary and expanding the project to larger pipe organs.

## Citations

[1] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in Proceedings of the Neural Information Processing Systems (NIPS), Denver, CO, USA, 2000, pp. 556–562.

[2] Mathworks.com. (2018). Lasso or elastic net regularization for linear models - MATLAB lasso. [online] Available at: https://www.mathworks.com/help/stats/lasso.html [Accessed 4 Dec. 2018].