Computational Machine Learning Homework4

Yun-shao Sung* and Hung-Ting Wen†

Abstract. This document served as the purpose of answering qestions from homework assignment, and also conclude the experiment observations.

1. Introduction.

2. Background.

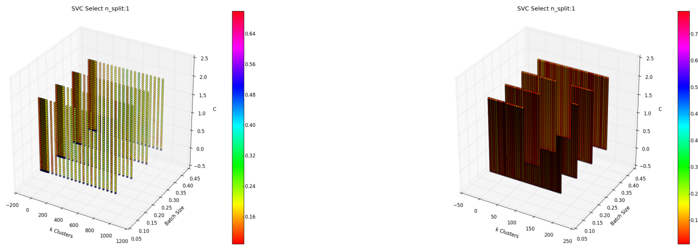
3. Baseline Approaches. To have our baseline approaches reaching similar accuracy as the previous finding, we have implemented some methods trying to reach the goal, and also trying to test our hypothesis. First of all is the interpretation of MFCC product. The MFCC product will be the shape of (n_mfcc, time), which n_mfcc is the number of mel-frequency and we set as 20, and time this time frame of the input clip signal. Then, as the instinct from previous experiment, we transpose MFCC product to the shape of (time, n_mfcc) which is transforming the meaning of feature representation of each mel-frequency over time to feature representation of each time moment in the clip over time mel-frequency. Althought this transpose will yield better accracucy, the further interprestation is yet unsure. We are not sure whether we should treat transposed-MFCC product as multi points in n_mfcc-dim space, or treat transposed-MFCC product as one point in $time \times n\_mfcc$-dim space.
The second question is about the VLAD. The idea of the VLAD descriptor is to accumulate, for each visual word c_i, the differences x-c_i of the vectors x assigned to c_i:

$$c_{i,j} = \sum_{x such that NN(x)=c_i} x_i - c_{i,j} \tag{3.1}$$

where x_i and $c_{i,j}$ respectively denote the jth component of the descriptor x considered and of its corresponding visual word c_i. We was wondering what if we sum up all the result into just d-dimension instead of the original VLAD representation $D = k \times d$, where k and d are the number of centroids and dimension of each centroid. Therefore, we named these two VLAD methods as Sum VLAD and Concatenate VLAD, and due to high dimension of Concatenate VLAD, we then perforemed PCA with whitening for dimension reduction.

As the results of the questions mentioned previously, we performed the folloing appraoches to reach the questions:
1. multi points MFCC + Kmeans + Sum VLAD + k nearest neighbor
2. multi points MFCC + Kmeans + Sum VLAD + SVC
3. one point MFCC + Kmeans + Sum VLAD + SVC
4. multi points MFCC + Kmeans + Concatenate VLAD + PCA + k nearest neighbor
5. multi points MFCC + Kmeans + Concatenate VLAD + PCA + SVC
6. one points MFCC + Kmeans + Concatenate VLAD + PCA + SVC

*yss265@nyu.edu
†htw230@nyu.edu



(a) Baseline method 5 (b) Feature extraction method 3

Figure 3.1: Best accuracy from baseline and new extraction method

As comparing method 1 and 2, we noticed generally SVC has better accuracy than KNN, but also at the cost of efficiency. This may due to the properity that SVC is using 1-vs-1 scheme for $n\_classes \times (n\_classes - 1)/2$ times, which maybe more delicate than KNN. We have also tested the LinearSVC method, which is 1-vs-rest scheme. Generally speaking, the accuracy of LinearSVC is still better than KNN and slightly lower than SVC.
By having the result from method 3 and compare to method 2, we noticed we cannot get any furthre improvement from method 3, and even the accuracy is decreased. This phenomenon is getting even worse when comparing the results from method 5 and 6. Therefore, we can clearly see treating the MFCC product as one point is definitely not a good method, and the possible explaination for this is probably due to we are fixing the sequencing meaning of feature representation into fixed order. For example a clip from Jazz genre may have the component of drum, guitar, and bass, and MFCC product gives us the features quantification at each time points. As we concatenate them with the fixed order, we are like telling the classifier the feature with this is order is belong to certain genre. However, the order of drump-guitar-bass or guitar-bass-drump should be equally considered as Jazz.
As comparing the method4 versus method1 and method5 versus method2, we can see the accuracy increased, especially in method5. Our centroids are obtained from the Kmeans of our implementation with Kmean++ for initialization and maxIter=200 as stopping criteria. Then our cancatenate VLAD method is the accumulation of measurement of each residuals, which defined as the vecter differences between each feature points and center, and then instead of sum up all vectors, here we concatenate each residuals. This method gave us a significant accuracy improvement, and is very likely due to concatenate each residuals at the same level will keep the strong feature while still save the minority feature representations. For example, if now this song which belong to Jazz has strong drum-related feature but also contains some guitar and bass feature that are not strong but yet representative enough, summing them all up like what we did in method 1-3 will easily loss those minority features. An abstraction metaphor for this idea, which hugely inspired our next new feature extraction method, is like a onion. During each step of our pipeline, we can either treat the onion as a whole, which we might not know this onion has multi-layers or probably we even don't know know this is a onion because it looks like a apple, or we can peeling each of the layer out and concatenate

each layer together. The peeling process will give us better understanding about this onion. In summary regarding to our current progress, which the pipeline described in method 5,our MFCC can reach to the accuracy of 70%, which is comparible to previous finding.

Method Ids	Accuracy (%)
Method 1	48
Method 2	58
Method 3	56
Method 4	50
Method 5	70
Method 6	33

Methods of each baseline approaches and its corresponding accuracy.

4. New Feature Extraction Approaches. As our result from baseline approaches (best 70%) yields is comparable to previouse finding using MFCC (best 71%), we also, based on the fundation of best baseline approach, experimentally invent new methods trying to further improve the best accuracy as well as the robustness of parameter searching, and here we mainly focus on the improvement of feature extraction. In additional to regular mfcc, librosa provides harmonic and percussive seperation (librosa.effects.hpss), which the underline mechanism is the STFT-HPSS-ISTFT pipeline, and it ensures that the output waveforms have equal length to the input signal. Secondly, we are trying to use librosa delta method (librosa.feature.delta) to capture the first and second derivative information from harmonic and percussive signal. As we mentioned from previous baseline method, concatenating MFCC product and treating it as one point is fixing the signal sequecial meaning which will limit the freedom the classification. However, the signal transition at every monent might hiding clear feature representation, and, therefore, taking the signal derivative and treat it as feature representation points might be a useful method. The final approach is the scattering of HPSS, and this idea is probably inspired by the switch of Sum VLAD to Concatenate VLAD mentioned previously. We noticed even signal being splited to harmonic and percussive parts, there are still plenty of residue signal in each splits. For example, although being removed significantly, there are still noticable percussive components in the harmonic split, and vise versa. Therefore, here we performed second order HPSS scattering as the new feature extraction method.

Below are our new approaches:
1. HPSS + MFCC + method 5 in baseline approach
2. HPSS w/ 1st delta w/ 2nd delta + MFCC + method 5 in baseline approach
3. 2nd HPSS scattering + MFCC + method 5 in baseline approach

From the original method mentioned in baseline approaches, now our method 1 can further reached to 77% of accuracy. As shown in figure 4.1, we noticed the harmonic and percussiv seperation is essential as we make the spectrogram plot of each parts. From the spectrogram plot of original clip, certain time period in original clip looks just like normal but these peroids actually existing noticable differences between harmonic and percussive parts. Just like the onion metaphor mentioned previously, we believe peeling off the original clip and contcate each of parts followed by MFCC will give us more feature representation to understand this song. As the result of this approach, in the improvement of this approach not only increase the max accuracy to 77%, but also improve the robustness of the search space. As we can see the row 2 in figure 4.3, the major peak of the search space now significantly shift from original accuracy of 50% to 65%. This implies this new feature extraction method can potentially release the computation time and guarantee a better accuracy.

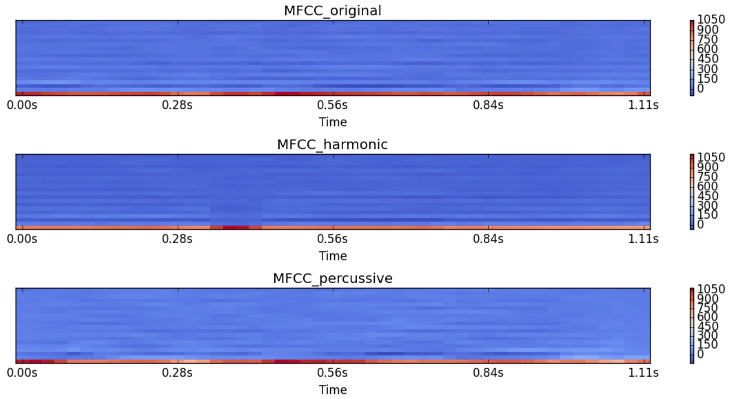


Figure 4.1: MFCC spectrogram of original clip, harmonic, and percussiv parts

In figure 4.2, not just the differences in harmonic and percussive parts of original song clip, we can also see the apparent difference when we take 1-order and 2-order of feature.delta. Instead of the original clip and its derivative here we keep only the harmonic and percussive parts and its 1-order and 2-order of derivative. Therefore, the feature point of each song clip will ne 6 times more than baseline method 5.

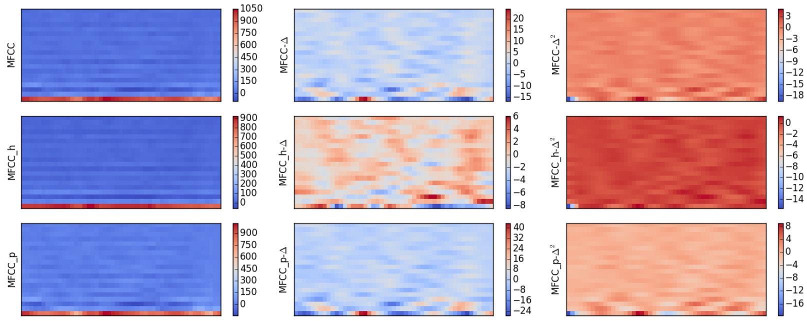


Figure 4.2: MFCC spectrogram of feature.delta from original clip, harmonic, and percussiv parts

TODOs
TODOs

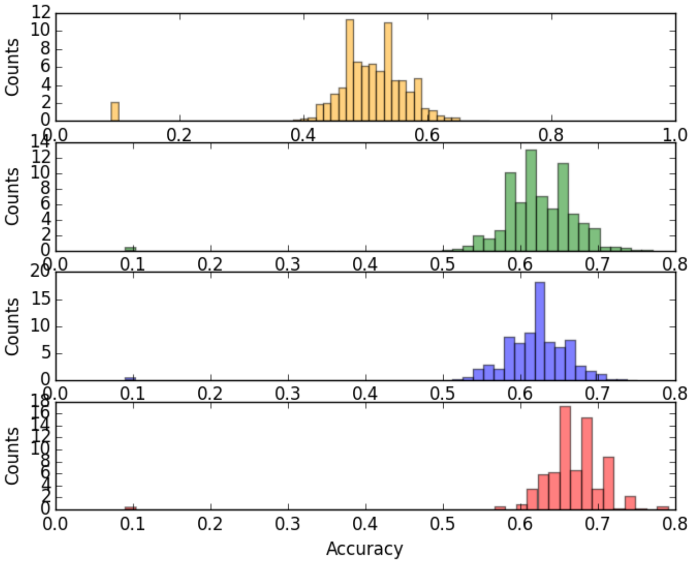


Figure 4.3: Histogram of accuracy result from search space. Top row is baseline method 5, and the remaining rows are new feature extraction method 1, 2, and 3 respectively

Method Ids	Accuracy (%)
Method 1	77
Method 2	76
Method 3	79

Methods of each new feature extraction approaches and its corresponding accuracy.

REFERENCES

[1] Low pass filter by FFT convolution, *http://www.dsprelated.com/freebooks/sasp/Example_1_Low_Pass_Filtering.html*

[2] Multiscale Scattering for Audio Classifications, *http://www.cmap.polytechnique.fr/scattering/ismir-final.pdf*

[3] Digital image processing: p067- Dictionary Learning, *https://www.youtube.com/watch?v=XLXSVLKZE7U*

[4] Interior-point methods, *https://web.stanford.edu/class/ee364a/lectures/barrier.pdf*

[5] Iterative Shrinkage/Thresholding Algorithms, *http://people.ee.duke.edu/ lcarin/figueiredo.pdf*

[6] Matching pursuits with time-frequency dictionaries, *http://www.cmap.polytechnique.fr/ mallat/papiers/MallatPursuit93.pdf*

[7] Efficient Implementation of the K-SVD Algorithm using Batch Orthogonal Matching Pursuit, *http://www.cs.technion.ac.il/ ronrubin/Publications/KSVD-OMP-v2.pdf*