FREQUENCY-ANCHORED DEEP NETWORKS FOR POLYPHONIC MELODY EXTRACTION

1st Aman Kumar Sharma*
MIG Routing
Cisco Systems
Bangalore, India
amansh2@cisco.com

2nd Kavya Ranjan Saxena*
Department of Electrical Engineering
Indian Institute of Technology
Kanpur, India
kavyars@iitk.ac.in

3rd Vipul Arora
Department of Electrical Engineering
Indian Institute of Technology
Kanpur, India
vipular@iitk.ac.in

***Abstract*—Extraction of the predominant melodic line from polyphonic audio containing more than one source playing simultaneously is a challenging task in the field of music information retrieval. The proposed method aims at providing finer F0s, and not coarse notes while using deep classifiers. Frequency-anchored input features extracted from constant Q-transform allow the signatures of melody to be independent of F0. The proposed scheme also takes care of the data imbalance problem across classes, as it uses only two or three output classes as opposed to a large number of notes. Experimental evaluation shows the proposed method outperforms a state-of-the-art deep learning-based melody estimation method.**

***Index Terms*—Melody extraction, music information retrieval, pitch shifting, constant Q-transform(CQT), Deep neural network**

I. Introduction

The melody of music is formed by arranging notes into an organized and rhythmic sequence. It usually dominates the human sense of hearing. Singing melody extraction is a challenging and critical task in music information retrieval (MIR), to estimate the melody pitch contour of singing voice from polyphonic music [1]. It has become an active area of research in the field of music information retrieval, having many important applications such as music retrieval [2]–[6], music transcription [7], [8], source separation [9], [10] and cover song identification [11]. A lot more challenging task is to extract melody in real-time from the musical performances [12]. With the rapid development of deep learning, many algorithms now use neural networks for melody extraction. There are many works that include data-driven pitch classification methods [13] and deep salience-based methods [14]–[18]. The first category employs deep neural networks to learn a discriminative representation of the spectrum of the music and then classify the melody into different pitches. The second category employs deep salience-based methods, which employ deep neural networks to learn the mapping between the mixture spectrum that represents the audio and the matrix with the same size that represents the melody line. Melody extraction using deep learning can be viewed as a classification or a regression problem. In the classification problem, the candidate pitches are categorized into a discrete number of pitch categories [19], [20]. The limitation of this model is that it will not be able to capture the fine variations in F0. The discretization of the candidate pitches into pitch categories loses the contour features such as vibrato and pitch deviation. Hence, the precision of the model will be low. This limitation is overcome by using regression models, as they will provide a continuous output for candidate pitches [21], [22]. But, generally deep regression models are not as effective as deep classification models. In order to estimate the high precision of F0 from the classification model, we have proposed a method that shows how a deep neural network model for music analysis on the constant Q transform (CQT) can be used for melody extraction. This method uses the property of logarithmic frequency representation in CQT that results in the constant pattern in the spectral components which acts as the feature. These features are then given as an input to the neural network model for classifying the frequency as melodic or non-melodic. Many deep classifier-based methods [19] estimate the notes in the output. Generally, there are a large number of notes and the data is heavily imbalanced among these classes. Also, there are many deep models which uses unlabeled data along with the labeled data because labelling data is laborious and costly [23]. The proposed method classifies the data into two or three classes. Thus, handling data imbalance is easier. The proposed frequency anchored feature vector makes the patterns of different $F0$s look similar, and thereby, easy to classify.

Section II describes the major modules of our system starting from feature building to finally the vocal melody estimation. Section III gives the comparative evaluation of the performance of this work with the state of the art model using standard databases. The conclusion follows in Section IV.

II. Proposed Method

Fig. 1 shows the overall schematic of the proposed method. The audio waveform is transformed to magnitude spectrum, $\bar{V}(f,t)$ with constant Q transform (CQT) using Hanning window at 10 ms hop size (128 samples) having 36 bins per octave (B). Here, f and t represent the frequency bin and time

This work was supported by IMPRINT-2C grant from SERB, Government of India Project no. IMP/2019/400.

* Equal contributors

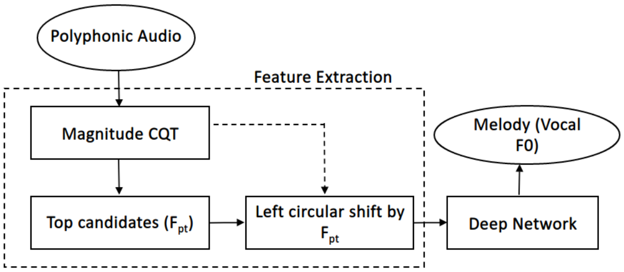

Fig. 1. Pipeline for vocal melody extraction from polyphonic audio

frame, respectively. The total number of frequency bins is N_f and the total time frames is N_t.

The obtained magnitude spectrum $\bar{V}(f,t)$ is then normalized with average sum per frame,

$$V(f,t) = \frac{\bar{V}(f,t)}{\sum_{f,t} \bar{V}(f,t)/N_t} \tag{1}$$

At each time frame t, a fixed number of K frequency bins, which includes top F frequency bins lying between the range of frequencies f_{min} and f_{max}, along with the adjacent spectral frequency bins F', are chosen. The remaining bins are discarded to avoid noisy peaks and also to save further processing time. All the bin candidates at time frame t are indexed with p and are denoted as F_{pt}. After narrowing down the potential F_{pt} bin candidates, a feature vector corresponding to each F_{pt} bin candidate is calculated. Now, we wish to find stationary patterns corresponding to the melody. In DFT, the harmonic frequencies are multiples of $F0$, so their spacing depends on $F0$. But if the frequencies are logarithmically scaled, as in CQT, the spacing between the adjacent harmonics is independent of $F0$. In CQT, we can visualize the signature of the melody source as a pattern of peaks shifted in frequency. We can use pattern matching techniques to identify the $F0$. A brute force way to achieve this could be to use convolutional neural networks. But a more efficient way could be to circular shift the CQT spectrum so that the bin candidate F_{pt} is anchored to the first bin as

$$V_{F_{pt}}(f) = V(\langle f + F_{pt} \rangle_{N_f}, t) \tag{2}$$

Here, $V_{F_{pt}}(f)$ is a vector that is given as an input to our model. This vector is calculated for each F_{pt} at each time frame t, such that the bin candidate F_{pt} is anchored to the first bin. We are interested in identifying the patterns associated with the peak F_{pt} corresponding to $F0$ (melody). The other peaks can be classified as non-melody. However, we know that the patterns of higher harmonics of $F0$s are somewhat distinct from the patterns of non-melodic peaks.

At a particular time frame t there will be at-most one melodic $F0$ candidate and multiple numbers of non-melodic sources. This leads to an imbalance in the number of samples in each class. For effective training of the model, we need to maintain uniform distribution of samples across all the classes.

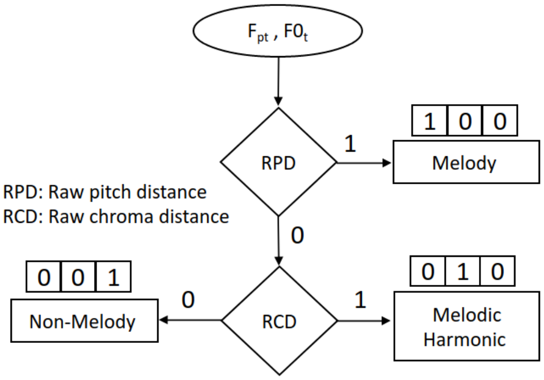

Fig. 2. Data annotation for proposed method - 1

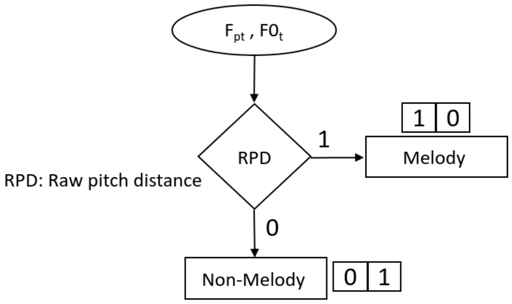

Fig. 3. Data annotation for proposed method - 2

So, the modified loss that is optimized during the training is the weighted categorical cross-entropy loss, given by

$$L_{\text{wBin}} = -\sum_{c=1}^{\text{M}} w_c y_{t,c} \log y_{i,c} \tag{3}$$

where $w_c \in \mathbb{R}$ is chosen to be inversely proportional to the number of samples of the class c, y_t and y_i are the ground truth and the predicted value for each class c, respectively. We prepare the feature matrix $V_{F_{pt}}(f)$ as an input to the deep neural network model. The architecture of the model consists of 3 dense layers and a classification layer. The first three layers of the model have 128, 128, and 32 nodes respectively with ReLu activation. The output of the third layer is fed into the classification layer, which classifies the candidate F_{pt} into one of the classes.

In proposed method-1, we are classifying the bin candidate F_{pt} into three classes, namely melodic, harmonic and non-melodic. We extract the feature matrix $V_{F_{pt}}(f)$ from the audio waveform. The annotations of F_{pt} into corresponding classes is done as mentioned in fig. 2. In the fig. 2, $F0_t$ is the bin corresponding to the ground truth frequency. Since we are classifying into three classes, the candidate F_{pt} is considered to be “melodic”(within 50 cents of ground truth), it is considered “harmonic” if it matches an integer multiple of the ground frequency $F0_t$, otherwise it is considered to be "non-melodic". The deep network model used now has a classifier layer having 3 nodes which classify the F_{pt} into one of the three classes. The loss function which is optimized is the same as given in (3).

In proposed method-2, we have merged the harmonic and non-melodic class as mentioned in proposed method-1 into one class, namely non-melodic. We extract the feature matrix $V_{F_{pt}}(f)$ from the audio waveform. Now, we classify candidate F_{pt} into two classes, namely melody, and non-melody. The annotations of F_{pt} into corresponding classes is done as mentioned in fig. 3. In the fig. 3, $F0_t$ is the bin corresponding to the ground truth frequency. Since we are classifying into two classes, the candidate F_{pt} is considered to be "melodic", if the pitch matches the ground frequency $F0_t$ (less than half semitone apart); otherwise, it is considered to be "non-melodic". The deep network model used now has a classifier layer having only 2 nodes which classify the F_{pt} into one of the two classes. The loss function which is optimized is the same as given in (3).

III. Experimental Setup

In this paper, we have used the first 800 audio clips of the MIR1K dataset which consists of 1000 Chinese songs to train our model. No data augmentation is performed. The testing data for evaluation are from three standard datasets for melody extraction: MIREX05, ADC2004, and MIR1K. Since the proposed model is designed solely for singing voice melody detection, we select those audios having melody sung by a human voice from MIREX05 and ADC2004. As a result, our test data consists of 11 clips from ADC2004, 9 clips from MIREX05, and the rest of 200 songs from MIR1K.

We use the patch-based CNN method [24] as the baseline method 1. It is based on extracting combined frequency and periodicity representation from the signal and extracts the vocal melody objects and then localizes a voice melody object while suppressing the harmonics. It is a deep classification model that selects patches as candidates of vocal melody objects in the representation, trains a CNN to determine whether a patch corresponds to a singing voice or not, and then localizes a voice melody object both in time and frequency. The output of the CNN model represents the likelihood of being a melodic contour. The pre-trained model of the baseline method 1 is trained on the first 800 songs of the MIR1K dataset and used for testing the test datasets.

Another method used as the baseline method 2 is the unweighted class method in which all the classes in the proposed method-2 were uniformly weighted, i.e, w_c = 1 for all classes c. So, the loss function that is optimized during the training is the categorical cross-entropy loss, given by

$$L_{\text{Bin}} = -\sum_{c=1}^{\text{M}} y_{t,c} \log y_{i,c} \tag{4}$$

where, y_t and y_i are the ground truth and the predicted value for each class c respectively.

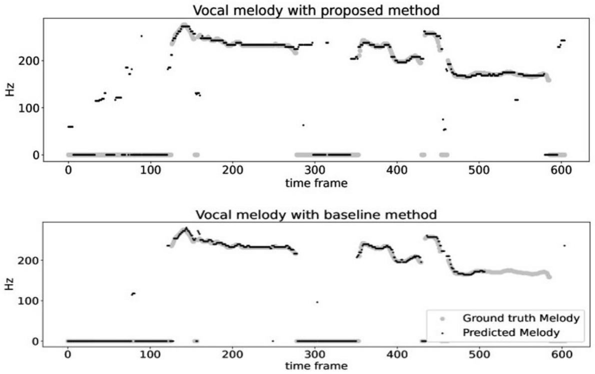

Fig. 4. Comparison of vocal melody extracted by proposed 2 and baseline method on a song titled 'stool_4_01'.

For our proposed methods , we have taken top K (=30) peaks between f_{min} (=50Hz) and f_{max} (=1500Hz), which includes top F (=10) peaks along with the adjacent F' neighbors at each time frame t. We have chosen the highest probabilistic score for multiple melodic predictions at a particular time frame. The proposed model is a dense model with 3 hidden layers having 128, 128 and 32 nodes respectively. The output layer is a classifier layer having three and two nodes corresponding to proposed-1 and proposed-2, respectively. The loss function used is the categorical loss function and the optimizer is the adam optimizer. The proposed models are trained end to end for 20 epochs with a batch size of 128. We performed the experiments by taking different values of bins per octave (B) as 12, 24, 36, and 48. B = 36 was chosen as the optimal value because it gave the best model accuracy.

A. *Results*

Table-I lists the raw pitch accuracy (RPA) and raw chroma accuracy (RCA) of the proposed methods and the baseline methods on all the three test datasets - MIREX05 and ADC2004 and MIR1K. All the metrics are computed by the mir-eval library [27] with the default setting, e.g a pitch estimate is considered correct if it is within 50 cents of the ground truth. From Table-I, we can see that proposed-2 works better than proposed-1 on all the three test datasets. To further probe into the performance of the two proposed methods, we analyze the classification ability of the two in the form of confusion matrices shown in Table-II. For detecting the melodic class, proposed-1 has got higher precision but lower recall as compared to proposed-2. Thus, proposed-1 is missing many true F0s, while proposed-2 is accepting many spurious F0s. In other words, as compared to proposed-1, proposed-2 is doing poorly in unvoiced regions but better in voiced regions. Since we are not considering unvoiced regions while computing RPA and RCA, proposed-2 gives better results. Another reason for the success of proposed-2 could be that it is difficult to separate non-melodic from harmonic candidates, as is apparent from Table-II(i). In proposed-1, the loss function gives equal weight to classify the three classes - hence, the

TABLE I
VOCAL MELODY EXTRACTION RESULTS OF THE PROPOSED AND BASELINE METHODS ON VARIOUS DATASETS

Method	MIREX05		ADC2004		MIR1K	
	RPA	RCA	RPA	RCA	RPA	RCA
Proposed 1	66.31	69.84	77.84	78.75	77.81	**79.80**
Proposed 2	73.01	76.01	**80.2**	**81.9**	**78.7**	79.1
Patch-based CNN (Baseline 1) [24]	73.98	74.47	67.81	67.86	57.31	69.23
Unweighted class (Baseline 2)	35.23	37.15	47.02	48.90	75.43	76.21
MCDNN (Baseline 3) [25]	70.10	71.60	45.38	49.28	69.74	72.46
Hybrid (Baseline 4) [26]	**74.36**	**76.22**	50.20	55.03	70.30	73.88

TABLE II
EXAMPLE OF CONFUSION MATRIX FOR A SONG TITLED POP1.WAV (I) WITH 3 CLASSES NAMELY MELODY:**0**, HARMONIC:**1**, NON-MELODY:**2** (II) WITH 2 CLASSES NAMELY MELODY:**0**, NON-MELODY:**1**

Predicted class

Actual Class

	0	**1**	**2**
0	4282	39	292
1	39	5979	384
2	957	1026	50069

(i)

	0	**1**
0	4445	168
1	1948	56470

(ii)

melodic class is not so well separated from the other two. In proposed-2, however, the goal is only to separate the melodic class from the rest, and hence, the performance for melody detection escalates. Since the proposed method-2 works better than proposed method-1, the comparison of vocal melody extracted by proposed method-2 and baseline method-1 is shown in the fig. 4.

Again, in Table-I, both the proposed methods outperform the baseline methods for ADC2004 and MIR1K dataset. For MIREX05, however, we see proposed-2 gives better RCA but slightly worse RPA as compared to those of the baseline. From Table-I, we can see that proposed-2 works better than proposed-1 on all the three test datasets. To further probe into the performance of the two proposed methods, we analyze the classification ability of the two in the form of confusion matrices shown in Table-II. For detecting the melodic class, proposed-1 has got higher precision but lower recall as compared to proposed-2. Thus, proposed-1 is missing many true F0s, while proposed-2 is accepting many spurious F0s. In other words, as compared to proposed-1, proposed-2 is doing poorly in unvoiced regions but better in voiced regions. Since we are not considering unvoiced regions while computing RPA and RCA, proposed-2 gives better results. Another reason for the success of proposed-2 could be that it is difficult to separate non-melodic from harmonic candidates, as is apparent from Table-II(i). In proposed-1, the loss function gives equal weight to classify the three classes - hence, the melodic class is not so well separated from the other two. In proposed-2, however, the goal is only to separate the melodic class from the rest, and hence, the performance for melody detection escalates.

Again, in Table-I, both the proposed methods outperform the baseline methods for ADC2004 and MIR1K dataset. For MIREX05, however, we see proposed-2 gives better RCA but slightly worse RPA as compared to those of the baseline. A possible reason for this could be that our proposed frequency-anchored feature vector uses the entire spectral information of a time frame, while the baseline 1 method uses it only partly.

IV. CONCLUSION

We have built a deep classification model that extracts vocal melody from polyphonic audio. Here, CQT is used because the frequencies are logarithmically scaled, which gives stationary patterns corresponding to the melody. This method reduces two degrees of freedom, which is the relative distance between the harmonics and absolute position by shifting F0, thus making the feature architecture independent of the pitch shift. Experiment results show that our model performs better than the state of the art method. We can use better imbalanced data handling techniques [28] for better classification.

REFERENCES

[1] J. Salamon, E. Gómez, D. P. Ellis, and G. Richard, “Melody extraction from polyphonic music signals: Approaches, applications, and challenges,” *IEEE Signal Processing Magazine*, vol. 31, no. 2, pp. 118–134, 2014.

[2] M. Marolt, “A mid-level representation for melody-based retrieval in audio collections,” *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1617–1625, 2008.

[3] Y. Yu, R. Zimmermann, Y. Wang, and V. Oria, “Scalable content-based music retrieval using chord progression histogram and tree-structure lsh,” *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 1969–1981, 2013.

[4] P. Knees and M. Schedl, “Music retrieval and recommendation: A tutorial overview,” in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2015, pp. 1133–1136.

[5] Z. Li, B. Zhang, Y. Yu, J. Shen, and Y. Wang, “Query-document-dependent fusion: A case study of multimodal music retrieval,” *IEEE transactions on multimedia*, vol. 15, no. 8, pp. 1830–1842, 2013.

[6] J. Salamon and E. Gómez, “Melody extraction from polyphonic music signals using pitch contour characteristics,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1759–1770, 2012.

[7] G. E. Poliner, D. P. Ellis, A. F. Ehmann, E. Gómez, S. Streich, and B. Ong, “Melody transcription from music audio: Approaches and evaluation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1247–1256, 2007.

[8] A. Rizzi, M. Antonelli, and M. Luzi, “Instrument learning and sparse nmd for automatic polyphonic music transcription,” *IEEE Transactions on Multimedia*, vol. 19, no. 7, pp. 1405–1415, 2017.

[9] V. Arora and L. Behera, “Multiple f0 estimation and source clustering of polyphonic music audio using plca and hmrfs,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 2, pp. 278–287, 2015.

[10] V. Arora and L. Behera, “Musical source clustering and identification in polyphonic audio,” *IEEE/ACM transactions on audio, speech, and language processing*, vol. 22, no. 6, pp. 1003–1012, 2014.

[11] J. Serra, E. Gómez, and P. Herrera, “Audio cover song identification and similarity: background, approaches, evaluation, and beyond,” in *Advances in Music Information Retrieval*. Springer, 2010, pp. 307–332.

[12] V. Arora and L. Behera, "On-line melody extraction from polyphonic audio using harmonic cluster tracking," *IEEE transactions on audio, speech, and language processing*, vol. 21, no. 3, pp. 520–530, 2012.

[13] S. Kum, C. Oh, and J. Nam, "Melody extraction on vocal segments using multi-column deep neural networks." in *ISMIR*, 2016, pp. 819–825.

[14] T.-H. Hsieh, L. Su, and Y.-H. Yang, "A streamlined encoder/decoder architecture for melody extraction," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 156–160.

[15] T.-H. Hsieh, L. Su, and Y.-H. Yang, "A streamlined encoder/decoder architecture for melody extraction," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 156–160.

[16] L. Su, "Vocal melody extraction using patch-based cnn," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 371–375.

[17] W. T. Lu, L. Su *et al.*, "Vocal melody extraction with semantic segmentation and audio-symbolic domain transfer learning." in *ISMIR*, 2018, pp. 521–528.

[18] R. M. Bittner, B. McFee, J. Salamon, P. Li, and J. P. Bello, "Deep salience representations for f0 estimation in polyphonic music." in *ISMIR*, 2017, pp. 63–70.

[19] H. Park and C. D. Yoo, "Melody extraction and detection through lstm-rnn with harmonic sum loss," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2766–2770.

[20] D. Basaran, S. Essid, and G. Peeters, "Main melody extraction with source-filter nmf and crnn," in *ISMIR*, 2018.

[21] A. Nagathil, J.-W. Schlattmann, K. Neumann, and R. Martin, "A feature-based linear regression model for predicting perceptual ratings of music by cochlear implant listeners," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 346–350.

[22] Y.-R. Chien, H.-M. Wang, S.-K. Jeng *et al.*, "An acoustic-phonetic approach to vocal melody extraction." in *ISMIR*, 2011, pp. 25–30.

[23] S. Kum, J.-H. Lin, L. Su, and J. Nam, "Semi-supervised learning using teacher-student models for vocal melody extraction," *arXiv preprint arXiv:2008.06358*, 2020.

[24] L. Su, "Vocal melody extraction using patch-based cnn," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 371–375.

[25] S. Kum, C. Oh, and J. Nam, "Melody extraction on vocal segments using multi-column deep neural networks." in *ISMIR*, 2016, pp. 819–825.

[26] H. Chou, M.-T. Chen, and T.-S. Chi, "A hybrid neural network based on the duplex model of pitch perception for singing melody extraction," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 381–385.

[27] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, D. P. Ellis, and C. C. Raffel, "mir_eval: A transparent implementation of common mir metrics," in *In Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR*. Citeseer, 2014.

[28] V. Arora, M. Sun, and C. Wang, "Deep embeddings for rare audio event detection with imbalanced data," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3297–3301.