AUTOMATIC MUSIC TRANSCRIPTION LEVERAGING GENERALIZED CEPSTRAL FEATURES AND DEEP LEARNING

Yu-Te Wu[1,2], *Berlin Chen*[1], *Li Su*[2]

[1]Dept. CSIE, National Taiwan Normal University, Taiwan
[2]Institute of Information Science, Academia Sinica, Taiwan

ABSTRACT

Spectral features are limited in modeling musical signals with multiple concurrent pitches due to the challenge to suppress the interference of the harmonic peaks from one pitch to another. In this paper, we show that using multiple features represented in both the frequency and time domains with deep learning modeling can reduce such interference. These features are derived systematically from conventional pitch detection functions that relate to one another through the discrete Fourier transform and a nonlinear scaling function. Neural networks modeled with these features outperform state-of-the-art methods while using less training data.

Index Terms— Automatic music transcription, cepstrum, deep learning, convolutional neural networks.

1. INTRODUCTION

Automatic music transcription (AMT) is one of the most important tasks in music information retrieval (MIR) [1]. Recently, increasing number of studies has incorporated deep learning to this direction, such as vocal melody extraction [2, 3] and multipitch estimation (MPE) [4–7], the latter one being the main focus of this work. However, utilizing deep learning in MPE is still a relatively new, and even a hard topic, comparing to other widely-used techniques such as the non-negative matrix factorization (NMF), sparse coding (SC) and convolutional sparse coding (CSC) [8–10]. Those techniques seek to decompose a spectrogram over a dictionary with note-specific templates. Such a modeling strategy has been prove successful, with recorded high performance on the well-known MAPS dataset [11] under constraints of sparsity, attack-decay pattern, and instrument type [9].

The challenges of using deep learning techniques in MPE include: limited annotated data, heavy computation loading for searching network hyperparameters, and the choice of *data representation* (i.e., feature) [12]. The latest one, as pointed out in [4], is an additional complication found only in the audio domain, in contrast to other tasks in which only raw input data suffice to give competitive performance. A systematic investigation in the same paper further demonstrated that the performance of MPE is not only sensitive to the spectrogram type (i.e. linear-frequency scale, log-frequency-scale, or constant-Q transform), but even sensitive to very basic signal parameters such as sampling rate [4].

One way to mitigate this issue is to leverage multiple data representations as the model input. For example, Böck *et al.* employed multi-resolution short-time Fourier transforms (STFTs) computed with different window sizes as the input of a recurrent neural network [6]. Nam *et al.* also takes a multi-resolution approach in the vocal melody extraction task, while different resolution is in the model prediction rather than the input representation [2]. Recently, Bittner *et al.* further proposed the harmonic constant-Q transform (HCQT), which combines multiple CQTs with different minimal frequencies (f_{min}) to make the harmonic components of each pitch being aligned across the input channels of a convolutional neural network (CNN) [5]. These studies indicate that multiple data representations are favorable in deep learning modeling, which is highly flexible in combining data inputs and in learning correlations among them.

It is worth noting that all of the above-mentioned studies use only spectral representations, such as spectrogram and CQT, as the data representation; it is probably because such representations are just what being utilized in other state-of-the-art methods such as NMF and SC. However, a deep learning model should not be restricted in this way since it does not perform signal decomposition. To explore more possibility in deep learning, we may relax this restriction and revisit other pitch detection functions reported in the literature. For example, the *generalized cepstrum* (GC), a lag-domain[1] representation widely used in early feature-based MPE algorithms [14–16], and the recently proposed *generalized cepstrum of spectrum* (GCoS) [17, 18], as a generalization of the autocorrelation of spectrum [19], can both be potential candidates for being the input data representation.

This paper for the first time employs both frequency- and time-domain data representations, including spectrum, GC and GCoS, as the multi-channel input to a CNN. Incorporating the "combined frequency and periodicity" (CFP)

[1]The cepstrum is also referred to as a feature in the *quefrency* domain [13]. In practice, 'quefrency,' 'lag' and 'time' are with the same unit. Therefore, these terms are used interchangeably in this paper.

approach [20], one recently proposed feature based MPE approach into our discussion, we show that the above-mentioned representations linked with a Fourier transform encompass most of the conventional pitch salience function, and such a multi-channel data representation can better enhance the fundamental frequencies than using only spectral representations. Evaluation on a dataset with a comparable baseline demonstrates the advantage of the proposed method.

2. DATA REPRESENTATION

2.1. The CFP approach

The CFP approach was firstly proposed in [20] for MPE of polyphonic music. It is then extended to the de-shaped time-frequency analysis, a general and rigorous theory representing multi-component signals with oscillatory signals [21]. The CFP approach holds the view that a pitch, as an object in audio signals, cannot be described solely by the frequency spectrum of the signal. Rather, it is described as a composite of frequency, periodicity, and harmonicity, implying that both the time-domain and frequency-domain representations are equally important [17, 18, 20, 22].

The basic assumption of the CFP approach is that the information of pitch tends to be the *fast-varying* part in every of its data representation, while the *slow-varying* parts are irrelevant to pitch. For example, when a cepstrum is employed as a pitch detector, we analyze only its high-quefrency counterparts while discard its low-quefrency ones, since the low-quefrency counterparts represents the spectral envelope rather than pitch [23]. The CFP approach then argue that a pitch object at frequency f_0 and time t_0 of a signal can be detected as true by means of a time-frequency representation $V(f,t)$, a time-quefrency representation $U(q,t)$, and the following *constraints of harmonicity* stating that

1. A sequence of prominent peaks are found at $V(t_0, f_0)$, $V(t_0, 2f_0)$, ..., $V(t_0, M_v f_0)$.
2. A sequence of prominent peaks are found at $U(t_0, q_0)$, $U(t_0, 2q_0)$, ..., $U(t_0, M_u q_0)$.
3. $f_0 = 1/q_0$.

The constraints of harmonicity have been implemented with a number of hand-crafted rules [20]. However, setting the parameters such as M_v, M_u and the threshold parameters for identifying peaks is rather ad-hoc. Therefore, in this paper we consider a data-driven modeling approach that represents the constraints of harmonicity in a neural network, by employing $V(f,t)$ and $U(q,t)$ as input.

2.2. Data representation

Given an input signal $\mathbf{x}$, a window function $\mathbf{h}$, $\mathbf{x}, \mathbf{h} \in \mathbb{R}^N$, $\mathbf{x} := \mathbf{x}[n]$ where n represents the time index, the amplitude part of the short-time Fourier transform (STFT) of $\mathbf{x}$ is defined as follows:

$$\mathbf{X}[k,n] := \left| \sum_{m=0}^{N-1} \mathbf{x}[m+nH]\mathbf{h}[m]e^{-\frac{j2\pi km}{N}} \right| , \qquad (1)$$

This is also known as the square root of a *spectrogram*. Given an N-point DFT matrix $\mathbf{F}$, high-pass filters $\mathbf{W}_f$ and $\mathbf{W}_t$, and nonlinear activation functions σ, consider the followings:

$$\mathbf{Z}_0[k,n] := \sigma_0 \left(\mathbf{W}_f \mathbf{X} \right) , \qquad (2)$$

$$\mathbf{Z}_1[q,n] := \sigma_1 \left(\mathbf{W}_t \mathbf{F}^{-1} \mathbf{Z}_0 \right) , \qquad (3)$$

$$\mathbf{Z}_2[k,n] := \sigma_2 \left(\mathbf{W}_f \mathbf{F} \mathbf{Z}_1 \right) . \qquad (4)$$

According to [17], $\mathbf{Z}_0$ is a power-scaled spectrogram, $\mathbf{Z}_1$ is a GC, and $\mathbf{Z}_2$ is a GCoS. $\mathbf{Z}_0$ and $\mathbf{Z}_2$ are time-frequency representations indexed by frequency k and time n, while $\mathbf{Z}_1$ has a time and a *quefrency* dimension q since it is the inverse DFT of a frequency-domain signal. The nonlinear function is then a rectified and root-power function:

$$\sigma_i \left(\mathbf{Z} \right) = |\text{relu}(\mathbf{Z})|^{\gamma_i}, \quad i = 0, 1, 2 , \qquad (5)$$

where $0 < \gamma_i \leq 1$, relu$(\cdot)$ represents a rectified linear unit, and $|\cdot|^{\gamma_0}$ is an element-wise root function. $\mathbf{W}_f$ and $\mathbf{W}_t$ are two high-pass filters whose main purpose is to remove the slow-varying part, i.e. the components in the low-frequency or low-quefrency range. This can be done by setting $\mathbf{W}_f$ and $\mathbf{W}_t$ to be diagonal matrices and defining the cutoff frequency and quefrency, k_c and q_c, respectively:

$$\mathbf{W}_{f \text{ or } t}[l,l] = \begin{cases} 1 , & l > k_c \text{ or } q_c ; \\ 0 , & \text{otherwise} . \end{cases} \qquad (6)$$

The discussion in [17] indicated that (2)-(4) parametrized by γ_i encompass most of the pitch detection functions in the literature. For example, when $(\gamma_0, \gamma_1) = (2, 1)$, $\mathbf{Z}_1$ is known as the autocorrelation function (ACF); when $(\gamma_0, \gamma_1, \gamma_2) = (1, 2, 1)$, $\mathbf{Z}_2$ represents the ACF of spectrum, which has been reported useful in resolving the missing fundamental effect in a spectrum and reduce pitch detection errors [19]. Also notice that many of the feature-based MPE methods are based on the general form of $\mathbf{Z}_1$, the GC with γ_1 between 0 and 1 [14–16, 24]. When $\gamma \to 0$, $\mathbf{Z}_1$ also approaches the conventional cepstrum, a classic pitch salience function using the logarithm function for nonlinearity [13, 23]. The general form of $\mathbf{Z}_2$ has also been investigated in the MPE task [17]. In summary, our generalized representation makes sense since most of the the well-known pitch detection approach are themselves related to each other through Fourier duality and filtering.

To fit the perception scale of pitch, we map the above-mentioned representations to the log-frequency scale. This is done with a filterbank: $\mathbf{Z}_0$, $\mathbf{Z}_1$ and $\mathbf{Z}_2$ are all processed by a filterbank with 275 triangular filters ranging from 20 Hz to 4 kHz with an interval of 36 bands per octave. The resulting log-frequency features are called $\hat{\mathbf{Z}}_0$, $\hat{\mathbf{Z}}_1$ and $\hat{\mathbf{Z}}_2$ hereafter.

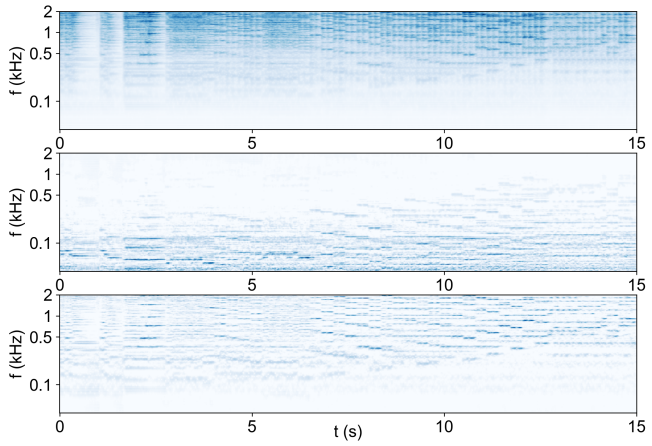

Fig. 1. Data representations of Edvard Grieg's *Kobold*, op.71, no.3, from the 10th to the 25th second, selected from the 'ENSTDkAm' subset in the MAPS dataset. From top to bottom: power-scale spectrogram ($\hat{\mathbf{Z}}_0$), generalized cepstrum ($\hat{\mathbf{Z}}_1$), and generalized cepstrum of spectrum ($\hat{\mathbf{Z}}_2$).

Fig. 1 illustrates the examples of $\hat{\mathbf{Z}}_0$, $\hat{\mathbf{Z}}_1$ and $\hat{\mathbf{Z}}_2$ of a segment of piano solo. As seen, $\hat{\mathbf{Z}}_0$ has rich harmonic components that make its energy almost concentrated in the high-frequency range. On the contrary, the sub-harmonic components in $\hat{\mathbf{Z}}_1$ make its energy concentrated in the low-frequency range. In both cases, the true fundamental frequencies are mostly of weak salience comparing to other components, making the results sensitive to the interference from the harmonic/sub-harmonic components and noise. Such an issue is mitigated in $\hat{\mathbf{Z}}_2$ through the high-pass filter; the high-frequency parts in $\hat{\mathbf{Z}}_2$ are suppressed so as to enhance those weak fundamental frequencies in the low-frequency range.

3. MODEL

We consider a CNN and a DNN model in this paper. The CNN model has two convolutional (CONV) layers followed by three fully-connected (FC) layers. The CONV layers have 32 (5×3) filters and 32 (1×3) filters respectively. It means that only the first layer covers multiple time steps, and the second CONV layer only covers one time step. In other words, for CNN we extend 2 more frames at both sides from current frame, summing to 5 frames in total. Both CONV layers are of varying dimensionality without padding. The number of units of the four FC layers is $[512, 512, 88]$. Detail architectures is shown in Table 1.

In the CNN architecture, each signal representation occupies one channel; that is, for the setting where $\mathbf{Z}_0$ and $\mathbf{Z}_1$ are combined, the input has two channels. In the DNN architecture, multiple input representations are concatenated.

For both the CNN and DNN, we employ the recently proposed scaled exponential linear units (SELU) as the activation function such that the network can be trained without batch normalization [25]. To avoid over-fitting, we use dropout and early-stopping. The initial network parameters are in Gaussian distribution with zero mean and 0.05 std. The output is an 88×1 binary-valued piano roll, where value 1 represents pitch activation. Modeled by the sigmoid function, the output layer is optimized by minimizing the binary cross-entropy of the output and the ground truth. The output of the network is a vector with the same dimension; the value of each bin is the likelihood of the activation of the pitch, represented in the range $[0, 1]$, and a binary prediction is obtained from a threshold at 0.5: $\overline{\mathbf{y}}_t = \mathbf{y}_t|_{\mathbf{y}_t[i]>0.5}$. We fit the model through Adam optimizer with the initial learning rate set to 0.001. Detail of the network is listed in Table 1.

CNN	DNN
Input 5 x 275 x # of channels	Input 1 x 275 x # of channels
Conv 32 x 5 x 3	Dropout 0.25
Conv 32 x 1 x 3	Dense 512
MaxPool 1 x 2	Dropout 0.5
Dropout 0.5	Dense 512
Dense 512	Dropout 0.5
Dropout 0.5	Dense 88
Dense 512	
Dropout 0.5	
Dense 88	
552408 parameters / channel	449112 parameters / channel

Table 1. Model architectures used in this paper.

4. EXPERIMENTS

The input signal is sampled at 44.1 kHz. The STFT is computed with the Blackman-Harris window with 0.18-second window size and 0.01-second hop size. The parameters of the nonlinear functions are $(\gamma_0, \gamma_1, \gamma_2) = (0.24, 0.6, 1)$. The feature extraction algorithm is implemented with MATLAB 2015b, and the deep learning algorithm is implemented with Python 3.5.2. The deep learning architecture is based on Keras with Tensorflow backend. The experiments is performed on Ubuntu 16.04 with i7-6700K CPU, and there are total 64GB of RAM. We use a GTX 1080 GPU for high-speed processing. The companion source code of this paper can be found at `https://github.com/BreezeWhite/CFP_NeuralNetwork` for reproducibility.

4.1. Data and evaluation metrics

We evaluate the performance of the proposed methods on the MAPS dataset, one of the most popular datasets in AMT [11]. Since the dataset provides isolated notes for every piano source, most of the previous studies focus on the transcription of only one piano, given the single notes of the same piano as training data [9]. On the contrary, only a few deep-learning-based algorithms are evaluated on the full dataset [4, 6, 7]. In

Data representation	60-sec training		full training	
	CNN	DNN	CNN	DNN
$\hat{\mathbf{Z}}_0$ (spectrum)	68.62	61.39	71.87	65.03
$\hat{\mathbf{Z}}_1$ (GC)	71.10	51.38	73.34	47.79
$\hat{\mathbf{Z}}_2$ (GCoS)	69.56	63.63	72.50	64.44
$[\hat{\mathbf{Z}}_0, \hat{\mathbf{Z}}_1]$	70.36	63.57	73.31	67.61
$[\hat{\mathbf{Z}}_0, \hat{\mathbf{Z}}_2]$	70.19	65.39	72.08	69.62
$[\hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$	**71.31**	66.14	**73.90**	70.03
$[\hat{\mathbf{Z}}_0, \hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$	70.26	**66.72**	72.63	**70.08**
[4]	–	–	70.60	65.15

Table 2. Results of Configuration II.

this work, we therefor adopt a *cross-instrument* train-test fold partition used in [4,7], namely Configuration II, in which only the 60 real piano recordings are used for testing, and other 210 synthetic piano recordings are used for training (180 recordings) and validation (30 recordings).

Apart from the cross-instrument evaluation, we are also interested in how the proposed method behaves when the training dataset is reduced. This is important for assessing the potential of an AMT method being applied to the transcription of low-resource data with limited labels. To this end, we again adopt Configuration II, but only use the first 60 sec of each recording in the training set for model training.

For the evaluation process, we count the number of true positives (TP), false positives (FP) and false negatives (FN) over all the frames in a test fold and then calculate the micro-average frame-level Precision (P), Recall (R), and F-score (F): $P = \text{TP}/(\text{TP} + \text{FP})$, $R = \text{TP}/(\text{TP} + \text{FN})$, and $F = 2PR/(P + R)$. A detected pitch is assumed to be a TP if it is within a half semitone of the ground-truth pitch of that frame. The average F-scores over all testing folds of different experimental settings are then reported in Table 2.

4.2. Experimental Results

Table 2 lists the resulting F-scores of the proposed CNN and DNN models. The left two columns list the results using the first 60-sec segment of each clip in the train folds for training, and the right two columns are those using full clips for training. At the outset, consider the first three rows showing the results using mono-channel inputs. In the full-training case, using $\hat{\mathbf{Z}}_0$ as the only input yields F-scores of 71.87% with CNN, and 65.03% with DNN, both are on par with the baseline results reported in [4], which also uses spectrum as the input. For both training schemes, $\hat{\mathbf{Z}}_1$ and $\hat{\mathbf{Z}}_2$ both outperform $\hat{\mathbf{Z}}_0$ with CNN, but $\hat{\mathbf{Z}}_1$ seems to be unsuitable with DNN; it reaches only 51.38% and 47.79% F-scores, respectively. This phenomenon is probably caused by the inharmonicity of piano. As the harmonic peaks of piano is not at exact integer multiples of the fundamental frequency, in GC there are additional speckles around the true peaks representing the pitches.

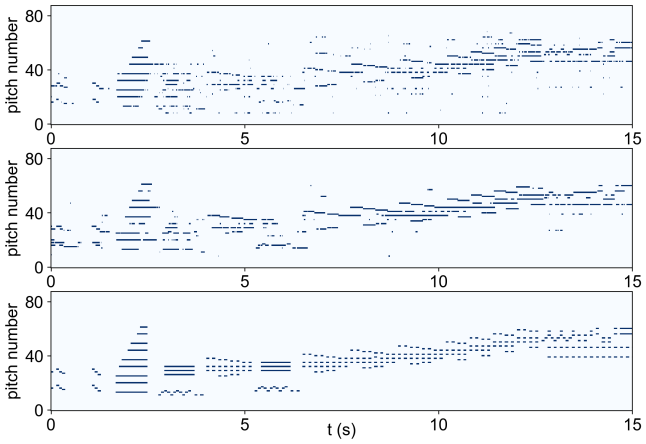

Fig. 2. Transcribed piano rolls of the same segment in Fig. 1. From top to bottom: result using $\hat{\mathbf{Z}}_0$, result using $[\hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$, and ground truth.

A CONV layer can better model such features than a FC layer does since it can smooth the speckles nearby a true peak.

For the multi-channel data representations, the best result appears to be $[\hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$ with CNN. It achieves an F-score of 73.90%, better than the baseline method in [4] by 3.3%. When using 60-sec training, it still outperforms [4], with only 25% of the training time in the full-training case (i.e., 153s/ep vs. 547s/ep). This implies that using the proposed multi-channel features does improve the accuracy and computing efficiency of the model. Also note that when using all the three data representations, the result is not always better than those using $[\hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$. Specifically, using all features results in the best performance for DNN, while using $[\hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$ is better than using all features for CNN. This is also an evidence that spectral features may not be the optimal data representation.

Fig. 2 shows examples of transcribed piano rolls using on a challenging piano solo with a wide pitch range and fast note groups, using $\hat{\mathbf{Z}}_0$ and $[\hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$ with CNN. Compared to the ground-truth piano roll, the result using $\hat{\mathbf{Z}}_0$ has a large number of upper-octave and lower-octave errors appearing as false alarms. When employing $[\hat{\mathbf{Z}}_1, \hat{\mathbf{Z}}_2]$ as input, these false alarms are mostly eliminated.

5. CONCLUSION AND ACKNOWLEDGEMENT

In this paper, we advocate to leverage new data representations other than raw spectrum in MPE. The proposed multi-channel CNN with CFP-based representations outperform state-of-the-art methods on the piano transcription task. Such an architecture will be scaled up for the transcription of other instruments. By utilizing Fourier duality, features 'deeper' than the GCoS are also worth investigating in the future.

This work is partially supported by MOST Taiwan, under the contract MOST 106-2218-E-001-003-MY3.

6. REFERENCES

[1] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: challenges and future directions," *J. Intelligent Information Systems*, vol. 41, no. 3, pp. 407–434, 2013.

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

[3] F. Rigaud and M. Radenen, "Singing voice melody transcription using deep neural networks.," in *Proc. ISMIR*, 2016, pp. 737–743.

[4] R. Kelz, M. Dorfer, F. Korzeniowski, S. Böck, A. Arzt, and G. Widmer, "On the potential of simple framewise approaches to piano transcription," *.arXiv:1612.05153*, 2016.

[5] R. M. Bittner, B. McFee, J. Salamon, P. Li, and J.P. Bello, "Deep salience representations for f_0 estimation in polyphonic music," in *Proc. ISMIR*, Oct. 2017.

[6] S. Böck and M. Schedl, "Polyphonic piano note transcription with recurrent neural networks," in *Proc. ICASSP*, 2012, pp. 121–124.

[7] S. Sigtia, E. Benetos, and S. Dixon, "An end-to-end neural network for polyphonic piano music transcription," *IEEE/ACM Trans. on Audio, Speech and Language Processing*, vol. 24, no. 5, pp. 927–939, 2016.

[8] K. O'Hanlon, H. Nagano, N. Keriven, and M. D. Plumbley, "Non-negative group sparsity with subspace note modelling for polyphonic transcription," *IEEE/ACM Trans. on Audio, Speech and Language Proc.*, vol. 24, no. 3, pp. 530–542, 2016.

[9] S. Ewert and M. Sandler, "Piano transcription in the studio using an extensible alternating directions framework," *IEEE/ACM Trans. on Audio, Speech, and Language Proc.*, vol. 24, no. 11, pp. 1983–1997, 2016.

[10] A. Cogliati, Z. Duan, and B. Wohlberg, "Context-dependent piano music transcription with convolutional sparse coding," *IEEE/ACM Trans. on Audio, Speech, and Language Proc.*, vol. 24, no. 12, pp. 2218–2230, 2016.

[11] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Trans. Audio, Speech, Lang. Proc.*, vol. 18, no. 6, pp. 1643–1654, 2010.

[12] K. Choi, G. Fazekas, K. Cho, and M. Sandler, "A tutorial on deep learning for music information retrieval," *.arXiv:1709.04396*, 2017.

[13] A. V. Oppenheim and R. W. Schafer, "From frequency to quefrency: A history of the cepstrum," *IEEE Signal Processing Magazine*, vol. 21, no. 5, pp. 95–106, 2004.

[14] A. Klapuri, "Multipitch analysis of polyphonic music and speech signals using an auditory model," *IEEE Trans. Audio, Speech, Lang. Proc.*, vol. 16, no. 2, pp. 255–266, 2008.

[15] H. Indefrey, W. Hess, and G. Seeser, "Design and evaluation of double-transform pitch determination algorithms with nonlinear distortion in the frequency domain-preliminary results," in *Proc. ICASSP*, 1985, pp. 415–418.

[16] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Processing*, vol. 8, no. 6, pp. 708–716, 2000.

[17] L. Su, "Between homomorphic signal processing and deep neural networks: Constructing deep algorithms for polyphonic music transcription," in *APSIPA ASC*, 2017.

[18] L. Su, "Vocal melody extraction using patch-based CNN," in *Proc. ICASSP*, 2018.

[19] G. Peeters, "Music pitch representation by periodicity measures based on combined temporal and spectral representations," in *Proc. ICASSP*, 2006.

[20] L. Su and Y.-H. Yang, "Combining spectral and temporal representations for multipitch estimation of polyphonic music," *IEEE/ACM Trans. Audio, Speech, Language Proc.*, vol. 23, no. 10, pp. 1600–1612, 2015.

[21] C.-Y. Lin, L. Su, and H.-T. Wu, "Wave-shape function analysis–when cepstrum meets time-frequency analysis," *Journal of Fourier Analysis and Applications*, pp. 1–55, 2017.

[22] L. Su, T.-Y. Chuang, and Y.-H. Yang, "Exploiting frequency, periodicity and harmonicity using advanced time-frequency concentration techniques for multipitch estimation of choir and symphony," in *Proc. ISMIR*, 2016, pp. 393–399.

[23] A.V. Oppenheim and R.W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall, 3rd edition, 2009.

[24] T. Kobayashi and S. Imai, "Spectral analysis using generalized cepstrum," *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. 32, no. 5, pp. 1087–1089, 1984.

[25] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks.," in *Advances in Neural Information Processing Systems*, 2017.