An Analysis of Using Fuzzy Annotations in CRNN-Based Joint Beat and Downbeat Tracking

Tian Cheng and Masataka Goto
National Institute of Advanced Industrial Science and Technology (AIST), Japan
{*tian.cheng, m.goto*}*@aist.go.jp*

***Abstract*—This paper addresses joint beat and downbeat tracking by using a Convolutional Recurrent Neural Network (CRNN) trained on Mel-spectral and chroma features of musical audio signals. Since beats and downbeats occur sparsely, we extend the original beat/downbeat annotation ranges to create fuzzy annotations for a better-balanced training. We compare three fuzzy annotations generated by convolving the original annotations with triangular, Gaussian, and rectangular windows. In comparison to the baseline model trained with the original annotations, our model of using fuzzy annotations improves the beat and downbeat tracking performance on pieces with tempi slower than 80 bpm. We analyzed the activation functions from the CRNN outputs and found that the improvements are mainly due to enlarged activation functions and phase error correction. The results showed that the proposed models provide promising results on test datasets with various music styles.**

***Index Terms*—Convolutional Recurrent Neural Network, Fuzzy Annotations, Joint Beat and Downbeat Tracking**

I. Introduction

Beat and downbeat are two fundamental metrical concepts for music, defining a hierarchical beat structure in two rhythmic levels: beat-level and bar-level [1]. They appear quasi-periodically along music pieces, and hence beat and downbeat tracking has been applied as an intermediate processing step for time regulation in other music analysis tasks, such as music transcription [2], [3], chord estimation [4]–[6], structure analysis [7], [8], and so on. Beat and downbeat tracking is also important for music-synchronized applications [9]–[11]. The close relationship between beat and downbeat tracking and other musical features is evidenced by several joint estimation systems for drums [12], [13], onsets [2], [13], chords [14], and tempo [15], [16].

Beat and downbeat tracking not only depends on local musical features, such as onsets and harmonic changes, but also deals with more global consistency to obtain steady tracking results. Current beat and downbeat tracking methods employ deep learning techniques to learn beats and downbeats from labeled music annotations, and feed the deep networks with spectral features extracted from musical audio signals. Mel-spectrograms and their temporal differences have been used as input features in beat and downbeat tracking papers and show promising results [17]–[19]. Chroma features show close relations to downbeat tracking and have been used in several downbeat tracking systems [20]–[22]. Various deep models have been applied for such tasks, ranging from Recurrent Neural Network (RNN) [17]–[20], Convolutional Neural Network (CNN) [21], [23], Convolutional Recurrent Neural Network (CRNN) [22], [24] to the most latest Temporal Convolutional Network (TCN) [15], [16], [25]. In these models the convolution layers detect the local spectral events, such as onsets and harmonic changes, and sequence models (RNN or TCN) estimate beats and downbeats in a larger timescale.

In this paper, we also adapt the advantages from both signal processing and machine learning domains to tackle beat and downbeat tracking jointly. We extended a CRNN model in our previous paper [24] by including both Mel-spectral and chroma features. Since the appearance of those labels is imbalanced, the beat/downbeat annotation ranges are extended to create fuzzy annotations by convolving the original annotations with three different kinds of windows: triangular, Gaussian, and rectangular windows. The results showed that the proposed models outperform state-of-the-art methods on beat and downbeat tracking in two public available datasets (SMC [26] and Beatles [27]). In comparison to the baseline method trained with original annotations, using fuzzy annotations improves the beat tracking performance on the SMC dataset [26] and the downbeat tracking performance on the Songle dataset [24], [28]. We analyzed the performance improvements on individual pieces and found that using fuzzy annotations improves the beat and downbeat tracking performance on pieces with tempi slower than 80 bpm. Using fuzzy annotations can enlarge the activation functions, which makes the beat/downbeat detection easier, and suppress spurious inter-beat peaks, which helps the phase error correction. We also found that using fuzzy annotations is helpful when the annotations contain some deviation because the enlarged beat range is more likely to include the true beat times.

Related work: Fuzzy annotations have been used recently, including those created by convolving the original annotations with a triangular window [15], [16], [25], Gaussian window [23], [29], and rectangular window [24]. However, only one kind of window has been studied in individual papers. In this paper, we compare three different kinds of windows (triangular, Gaussian, and rectangular) and analyze the performance improvement of each individual piece. In our early study on using the Gaussian windows with different window sizes, we found that increasing window size did not improve the beat tracking performance [29]. Therefore, we focus on windows of small sizes in this paper.

This work was supported in part by JST CREST Grant Number JPMJCR20D4, Japan.

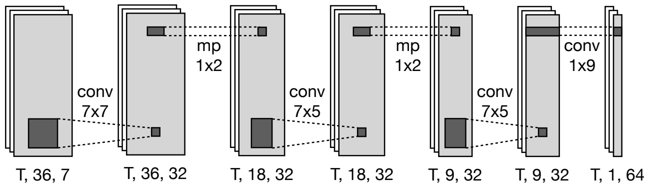


Fig. 1: Convolution (conv) and max pooling (mp) layers in the proposed model.

II. CRNN model for joint beat and downbeat tracking

A. Signal Pre-Processing

We compute Mel-spectral and chroma features in the pre-processing. To obtain Mel-spectrograms, a monaural waveform sequence is read from each audio file at a 44100 Hz sampling rate. Then the waveform is segmented into frames of window sizes of 1024, 2048, and 4096 samples with a hop size of 441 samples [19], resulting in 10 ms temporal resolution. For each windowed input frame we compute a magnitude Mel-spectrogram of 36 Mel bins within a frequency range from 30 Hz to 17000 Hz. Then we convert the Mel-spectrograms to the log-scale and calculate, along the time axis, their first-order difference with a positive half-wave rectifier. We concatenate the three Mel-spectrograms and their differences into 6 channels to obtain a Mel-spectral feature with a shape of [T, 36, 6], where T is the number of the input frames.

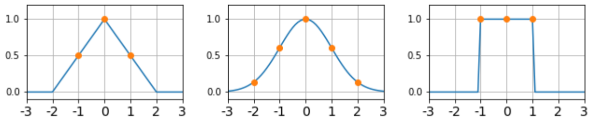
In addition to the Mel-spectrograms, we incorporate chroma features into the model input. We use three different kinds of chromagrams: a chromagram computed from a power spectrogram, a Constant-Q chromagram, and an energy normalized chromagram [30]. Since these chromagrams put different emphases on the frequency range, dynamics, and timbre, we concatenate them to obtain an informative and robust chroma feature.

The concatenated chroma feature has a dimension of 36 with 12 dimensions for each chromagram. It is stacked to the above Mel-spectral feature as another channel, resulting in an input feature of a shape of [T, 36, 7].

B. Network Architecture

We adapt the CRNN model from our previous work [24], which consists of a CNN block, an RNN block, and a fully connected layer. The CNN block consists of four convolutional layers, as shown in Figure 1. In the first three layers, we use 32 convolutional filters for each layer with the 'same' padding. The filter shapes are 7x7, 7x5, and 7x5, respectively. The fourth layer consists of 64 filters, with the shape of 1x9. There is a maxpooling layer stacked after each of the first two layers, with the shape of 1x2.

After the CNN layers, we reshape the output into a tensor of [T, 64] as the input of the RNN block. The RNN block consists of 4 bidirectional layers with 64 Gated Recurrent Units (GRUs) per layer in each direction. At the end of RNN layers we stack a dense layer with an output dimension of 3, which corresponds to the 'downbeat', 'beat', and 'no-beat' labels.



(a) Triangle (b) Gaussian (c) Rectangle

Fig. 2: Three windows used to create fuzzy annotations.

C. Training and Post-Processing

We train the CRNN model in 8-fold cross-validation with random splits. We apply the RMSprop optimizer [31] with a learning rate of 10^{-3} to minimize the cross-entropy error. We stop training if no improvement is found on the validation set in 15 epochs.

We adapt the post-processing method in [19]. First a threshold of 0.05 is applied on the beat/downbeat activation functions (from the CRNN model) to delete small peaks at the beginning and end of a music piece. Then a Dynamic Bayesian Network (DBN) is used to infer the metre, tempo, and beat phases jointly based on the observation distributions converted from the beat/downbeat activation functions. Readers are referred to [19], [32] for more details. In the experiment, we restrict the bar lengths to 2, 3, or 4 beats.

D. Fuzzy Annotations

In the original annotations, there are several 'beat' labels and a lot of 'no-beat' labels between adjacent 'downbeat' labels. For example, for a piece with a 4/4 time signature and a tempo of 120 bpm, in every 200 frames (2 sec) there are 1 downbeat and 3 beats, with the other 196 frames labeled as 'no-beat'. The annotations are even more imbalanced for a slower-tempo piece. To tackle this imbalance issue, we create fuzzy annotations by broadening the beat and downbeat labels, i.e., by convolving the original pulse-like beat and downbeat annotations with different windows (kernels). We compare three different windows: triangular, Gaussian, and rectangular windows, as shown in Figure 2.

We have studied the influence of different window sizes of the Gaussian window on beat tracking in our previous work [29], and the results showed that the best performance was achieved with a small window size, as shown in Figure 2(b). Based on the results, we choose the above compact windows, with a length of 3 or 5 frames.

III. Experiment

A. Datasets

We used 11 datasets with a variety of music genres in the experiment. Eight were used in cross-validation, Ballroom [33], [34], GTZAN [35], [36], Hainsworth [37], RWC classic, jazz, pop, royalty [38] and RWC genre [39]. There were also three held-out datasets for testing only. One was the Songle dataset, with 228 songs registered on [28] and with the annotations manually checked. The other two were publically

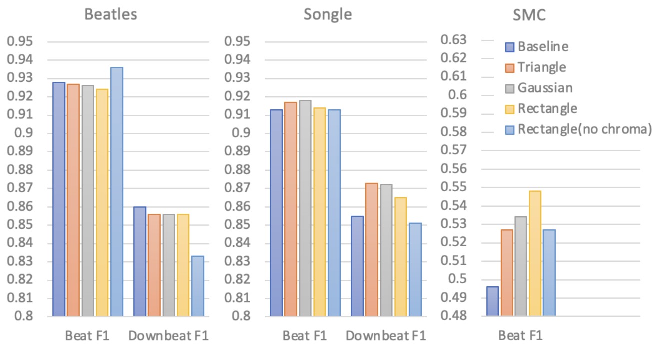


Fig. 3: Beat and downbeat tracking F-measures (*F1*) of the proposed models on the three test datasets. The baseline model was trained with the original annotations, and ‘Rectangle(no chroma)’ denotes the model trained with the rectangular window on Mel-spectrograms only [24].

SMC	*F1*	*CMLt*	*AMLt*
Proposed(Rectangle)	0.548	0.444	0.614
Böck et al. [16] †	0.544	0.443	0.635

(a) Beat tracking results

Beatles	*F1*	*CMLt*	*AMLt*
Proposed(Rectangle)	0.856	0.757	0.887
Durand et al. [21] †	0.847	0.722	0.875
Fuentes et al. [22] †	0.86		
Böck et al. [16] †	0.837	0.742	0.862

(b) Downbeat tracking results

TABLE I: A comparison to other state-of-the-art methods. † indicates cross-validation results reported in corresponding papers.

available datasets: the Beatles dataset [27] and the SMC dataset [26]. Note that the SMC dataset was built as a difficult dataset and only includes beat annotations, and hence there is no downbeat tracking result for this dataset.

B. Results

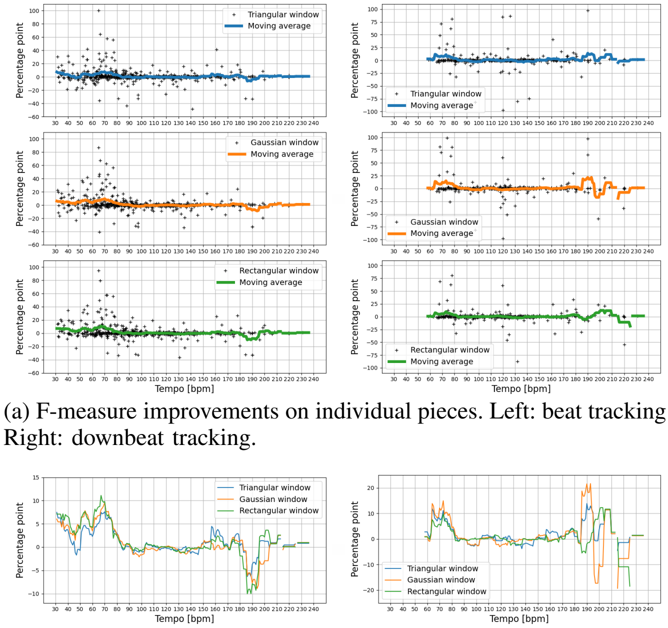
The beat and downbeat tracking results on the test datasets are shown in Figure 3. We first compared the proposed methods with the baseline method (trained with the original annotations) on individual datasets. For the Beatles dataset, the baseline method obtained the F-measure (*F1*) of 0.928 for the beat tracking task and obtained the F-measure of 0.86 for the downbeat tracking task. Using the fuzzy annotations did not bring significant differences in the performance of both tasks. For the Songle dataset, the baseline method obtained F-measures of 0.913 and 0.855 for beat tracking and downbeat tracking, respectively. Using the fuzzy annotations had no significant influence on beat tracking results but improved downbeat tracking results. There were improvements of 1.8, 1.7, and 1 percentage points brought by using triangular, Gaussian, and rectangular windows, respectively. For the SMC dataset, the beat tracking F-measure was 0.496 by using the baseline method. The performance was improved to 0.527, 0.534, and 0.548 by using the fuzzy annotations with the three windows, respectively. In general, using the fuzzy annotations can provide better or at least competitive results.

We also analyzed the effect of adding chroma features by comparing two methods trained using the rectangular window with and without chroma features (‘Rectangle’ vs. ‘Rectangle(no chroma)’ [24] in Figure 3). We found that adding chroma features brought a clear improvement of downbeat tracking, with improvements of 2.3 percentage points on the Beatles dataset and 1.4 percentage points on the Songle dataset. However, it had different effects on beat tracking for different datasets: it decreased the F-measure on the Beatles dataset for 1.2 percentage points, had no effect on the Songle dataset, and brought an improvement of 2.1 percentage points on the SMC dataset. This beat tracking performance could be interpreted by considering music types contained in the datasets. Music in the Beatles dataset has clear rhythmic clues, and the key problem is to find a constant tempo in ambiguous and noisy situations. Adding chroma features would make some situations more complicated, which would somehow decrease the beat tracking performance. On the other hand, music in the SMC dataset tends to have blurring onsets and there is much less information corresponding to the beats. Adding chroma features could be helpful in such situations.

Table I shows a comparison between the proposed method (trained with the rectangular window) and other state-of-the-art methods on two datasets. Noticeably, our testing-only results are comparable to the best existing cross-validation results. The testing-only evaluation is considered more difficult than typical cross-validation evaluation and shows its robustness on unseen music data. For the SMC dataset in Table I(a), we compared to a model built on CNN and TCN in [16] on the beat tracking performance. The proposed method provided a slightly better F-measure (*F1*). In the continuity-based metrics (*CMLt* and *AMLt* [27]), the model [16] (with an *AMLt* of 0.635) outperformed the proposed method (0.614) by 2.1 percentage points, showing better tolerances on errors like double/half tempo and off-beats. For downbeat tracking results of the Beatles dataset in Table I(b), the proposed method outperformed the methods of Durand et al. [21] and Böck et al. [16] on all three metrics. The F-measure of the proposed method was 0.856, slightly worse than the best F-measure of 0.86 in Fuentes et al. [22].

C. Performance Improvement Analysis

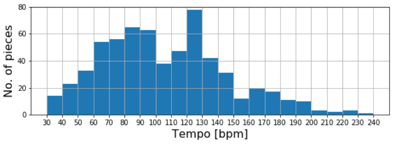
To understand how the fuzzy annotations work, we analyzed the performance improvements brought by using the fuzzy annotations for each individual piece in the three test datasets. In Figure 4(a), the y-axis of each point (‘+’) indicates the performance improvement for a musical piece brought by using the fuzzy annotations with each window; and the x-axis indicates the tempo of the piece computed from its ground-truth beat annotations. The lines represent the moving averages of the improvements along tempo (x-axis), with the average taken over pieces with similar tempi (in a range of 10 bpm). In Figure 4(b) with only the moving averages, it is clear that the



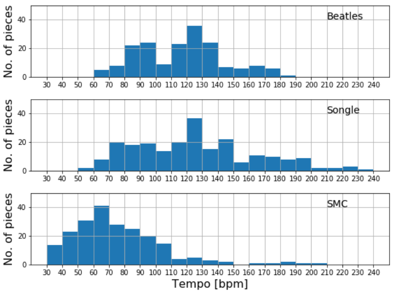
(a) F-measure improvements on individual pieces. Left: beat tracking; Right: downbeat tracking.

(b) Moving averages of F-measure improvements. Left: beat tracking; Right: downbeat tracking.

Fig. 4: F-measure improvements by the three windows for different tempi in the three test datasets.



(a) Tempo histogram of pieces in all three datasets.



(b) Tempo histograms of individual datasets.

Fig. 5: Tempo statistics for the three test datasets.

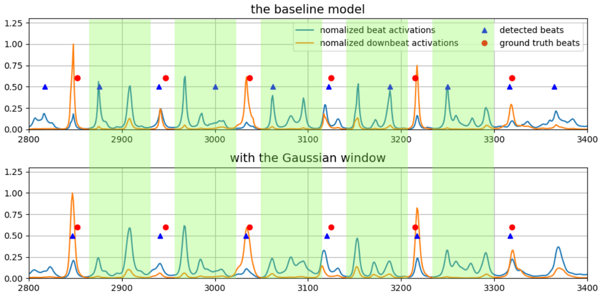


Fig. 6: Outputs of Piece SMC_236 (28s-34s).

fuzzy annotations worked for pieces with tempi slower than 80 bpm and brought subtle differences in the performance for pieces with tempi ranging from 80 to 150 bpm. For pieces with tempi faster than 150 bpm, the trend is not clear. This is because there are fewer pieces in this range, as shown in Figure 5(a); the average improvement was easily influenced by certain pieces.

Figure 5(b) showed the tempo histograms of the three test datasets. We found that the proportion of pieces with tempi slower than 80 bpm is related to the performance improvements in Figure 3. Since the proportion of slow tempi in the SMC dataset is much larger than those in the other datasets, there were larger improvements on beat tracking performance. The proportion of slow tempi in the Songle dataset is larger than that in the Beatles dataset, and hence there were improvements for the Songle dataset but no improvements for the Beatles dataset.

We further investigated the activation functions and found that using fuzzy annotations can enlarge activation functions and help correct phase errors. Firstly, the output (activation functions) of the model trained with fuzzy annotations is larger than that of the baseline model. We applied a threshold of 0.05 on the activation functions. If the activation functions are too small, then no beat is detected. With the fuzzy annotations, the activation functions become larger, and beats are detected. Secondly, we observed that even when we normalize the activation functions (which can help to solve the above threshold problem), there are some noticeable peaks between beats in the activation functions of the baseline model, as shown in Fig 6. With the fuzzy annotations, these peaks between beats are suppressed, making the phase errors easier to be recovered with the context information (in the post-processing with the DBN). We observed that the adjacent frames in the spectrogram are similar to each other due to the overlap. It is reasonable to put similar labels on adjacent frames around beats, so that the model can focus more on learning the latent pattern of beats and no-beats, rather than distinguishing beat and no-beat on similar inputs.

We looked at the cross-validation results and found that the largest improvement was on the Hainsworth dataset. In the experiment, we used the old annotations which are drifted from the beat times (the updated annotations can be found in [19]). When we enlarge the beat range, it is more likely to include the true beat times, which helps improve the results. This effect is also important because it is common that manual annotations are imprecise. This result suggests that it is worth trying large windows when the temporal accuracy of beat annotations is not high enough. This supplements our previous finding that small windows generally work better in [33].

D. Recommended choices

Since using the fuzzy annotations and adding the chroma features work differently on different music styles, the best choice depends on the target music data. For the fuzzy annotations, a rectangular window is recommended if the data tend to include slow-tempo pieces; otherwise, a triangular window is recommended. The Gaussian window could be used for the data with no prior information. Adding the chroma features is usually recommended unless the dataset has a lot of fast-tempo pieces or downbeat tracking is not necessary.

IV. Conclusions

In this paper we evaluated the idea of using fuzzy annotations in CRNN-based joint beat and downbeat tracking. In comparison to the baseline model with the original annotation, using the fuzzy annotations provides better performance on the test datasets: it improves results on relatively difficult datasets and brings no significant differences on relatively simple datasets. Our main contributions can be summarized as follows. (1) We evaluated the fuzzy annotations broadened by the three different kinds of windows, and provided state-of-the-art performance on beat and downbeat tracking. (2) We analyzed the performance improvement of each piece, and found that the main improvements are on pieces with tempi slower than 80 bpm. (3) We found that the improvements are mainly due to enlarged activation functions and phase error correction by investigating the activation functions from the CRNN outputs.

Future work will include tempo estimation with beat and downbeat tracking in a multi-task learning framework.

References

[1] M. Goto, "An Audio-based Real-time Beat Tracking System for Music With or Without Drum-sounds," *Journal of New Music Research*, vol. 30, no. 2, pp. 159–171, 2001.

[2] K. Ochiai, H. Kameoka, and S. Sagayama, "Explicit Beat Structure Modeling for Non-Negative Matrix Factorization-based Multipitch Analysis," in *Proc. IEEE ICASSP*, 2012, pp. 133–136.

[3] E. Nakamura, E. Benetos, K. Yoshii, and S. Dixon, "Towards Complete Polyphonic Music Transcription: Integrating Multi-Pitch Detection and Rhythm Quantization," in *Proc. IEEE ICASSP*, 2018, pp. 101–105.

[4] M. Mauch and S. Dixon, "Simultaneous Estimation of Chords and Musical Context from Audio," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 6, pp. 1280–1289, 2010.

[5] M. McVicar, R. Santos-Rodriguez, Y. Ni, and T. D. Bie, "Automatic Chord Estimation from Audio: A Review of the State of the Art," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 2, pp. 556–575, 2014.

[6] J. Pauwels, K. O'Hanlon, E. Gómez, and M. B. Sandler, "20 Years of Automatic Chord Recognition from Audio," in *Proc. ISMIR*, 2019.

[7] M. Levy and M. B. Sandler, "Structural Segmentation of Musical Audio by Constrained Clustering," *IEEE Trans. Audio, Speech, Language Process.*, vol. 16, no. 2, pp. 318–326, 2008.

[8] O. Nieto, G. J. Mysore, C. i Wang, J. B. L. Smith, J. Schlüter, T. Grill, and B. McFee, "Audio-Based Music Structure Analysis: Current Trends, Open Challenges, and Applications," *Trans. Int. Society Music Information Retrieval*, vol. 3, no. 1, pp. 246–263, 2020.

[9] M. Goto and Y. Muraoka, "A Beat Tracking System for Acoustic Signals of Music," in *Proc. ACM Multimedia*, 1994, pp. 365–372.

[10] J. L. Oliveira, M. E. P. Davies, F. Gouyon, and L. P. Reis, "Beat Tracking for Multiple Applications: A Multi-Agent System Architecture With State Recovery," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 10, pp. 2696–2706, 2012.

[11] J. Kato, M. Ogata, T. Inoue, and M. Goto, "Songle Sync: A Large-Scale Web-based Platform for Controlling Various Devices in Synchronization with Music," in *Proc. ACM Multimedia*, 2018, pp. 1697–1705.

[12] R. Vogl, M. Dorfer, G. Widmer, and P. Knees, "Drum Transcription via Joint Beat and Drum Modeling Using Convolutional Recurrent Neural Networks," in *Proc. ISMIR*, 2017.

[13] M. Cartwright and J. P. Bello, "Increasing Drum Transcription Vocabulary Using Data Synthesis," in *Proc. DAFx*, 2018.

[14] H. Papadopoulos and G. Peeters., "Joint Estimation of Chords and Downbeats from An Audio Signal," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 1, pp. 138–152, 2011.

[15] S. Böck, M. E. P. Davies, and P. Knees, "Multi-Task Learning of Tempo and Beat: Learning One to Improve the Other," in *Proc. ISMIR*, 2019, pp. 486–493.

[16] S. Böck and M. E. Davies, "Deconstruct, Analyse, Reconstruct: How to Improve Tempo, Beat, and Downbeat Estimation," in *Proc. ISMIR*, 2020.

[17] S. Böck and M. Schedl, "Enhanced Beat Tracking with Context-Aware Neural Networks," in *Proc. DAFx*, 2011.

[18] S. Böck, F. Krebs, and G. Widmer, "A Multi-Model Approach to Beat Tracking Considering Heterogeneous Music Styles," in *Proc. ISMIR*, 2014.

[19] ——, "Joint Beat and Downbeat Tracking with Recurrent Neural Networks," in *Proc. ISMIR*, 2016.

[20] F. Krebs, S. Böck, and G. Widmer, "Downbeat Tracking Using Beat-Synchronous Features and Recurrent Networks," in *Proc. ISMIR*, 2016.

[21] S. Durand, J. P. Bello, B. David, and G. Richard, "Robust Downbeat Tracking Using an Ensemble of Convolutional Networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 25, no. 1, pp. 72–85, 2017.

[22] M. Fuentes, B. Mcfee, H. C. Crayencour, S. Essid, and J. P. Bello, "Analysis of Common Design Choices in Deep Learning Systems for Downbeat Tracking," in *Proc. ISMIR*, 2018.

[23] A. Gkiokas and V. Katsouros, "Convolutional Neural Networks for Real-Time Beat Tracking: A Dancing Robot Application," in *Proc. ISMIR*, 2017, pp. 286–293.

[24] T. Cheng, S. Fukayama, and M. Goto, "Joint Beat and Downbeat Tracking Based on CRNN Models and a Comparison of Using Different Context Ranges in Convolutional Layers," in *Proc. ICMC*, 2020.

[25] M. E. P. Davies and S. Böck, "Temporal Convolutional Networks for Musical Audio Beat Tracking," in *Proc. EUSIPCO*, 2019.

[26] A. Holzapfel, M. E. P. Davies, J. R. Zapata, J. a. L. Oliveira, and F. Gouyon, "Selective Sampling for Beat Tracking Evaluation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 9, pp. 2539–2548, 2012.

[27] M. E. P. Davies, N. Degara, and M. D. Plumbley, "Evaluation Methods for Musical Audio Beat Tracking Algorithms," Queen Mary University of London, London, United Kingdom, Tech. Rep. C4DM-TR-09-06, 2009.

[28] M. Goto, K. Yoshii, H. Fujihara, M. Mauch, and T. Nakano, "Songle: A Web Service for Active Music Listening Improved by User Contributions," in *Proc. ISMIR*, 2011, pp. 311–316.

[29] T. Cheng, S. Fukayama, and M. Goto, "Convolving Gaussian Kernels for RNN-based Beat Tracking," in *Proc. EUSIPCO*, 2018, pp. 1919–1923.

[30] M. Müller and S. Ewert, "Chroma Toolbox: MATLAB Implementations for Extracting Variants of Chroma-Based Audio Features," in *Proc. ISMIR*, 2011.

[31] T. Tieleman and G. Hinton, "Lecture 6.5—RMSProp: Divide the Gradient by a Running Average of its Recent Magnitude," COURSERA: Neural Networks for Machine Learning, 2012.

[32] F. Krebs, S. Böck, and G. Widmer, "An Efficient State-Space Model for Joint Tempo and Meter Tracking," in *Proc. ISMIR*, 2015, pp. 72–78.

[33] F. Gouyon, A. Klapuri, S. Dixon, M. Alonso, G. Tzanetakis, C. Uhle, and P. Cano, "An Experimental Comparison of Audio Tempo Induction Algorithms," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 5, pp. 1832–1844, 2006.

[34] F. Krebs, S. Böck, and G. Widmer, "Rhythmic Pattern Modeling for Beat and Downbeat Tracking in Musical Audio," in *Proc. ISMIR*, 2013, pp. 227–232.

[35] G. Tzanetakis and P. Cook, "Musical Genre Classification of Audio Signals," *IEEE Speech Audio Process.*, vol. 10, no. 5, 2002.

[36] U. Marchand and G. Peeters, "Swing Ratio Estimation," in *Proc. DAFx*, 2015.

[37] S. W. Hainsworth and M. D. Macleod, "Particle Filtering Applied to Musical Tempo Tracking," *EURASIP Journal on Applied Signal Process.*, vol. 15, 2004.

[38] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC Music Database: Popular, Classical, and Jazz Music Databases," in *Proc. ISMIR*, 2002, pp. 287–288.

[39] ——, "RWC Music Database: Music Genre Database and Musical Instrument Sound Database," in *Proc. ISMIR*, 2003, pp. 229–230.