Extending Deep Rhythm for Tempo and Genre Estimation Using Complex Convolutions, Multitask Learning and Multi-input Network

Hadrien Foroughmand\textsuperscript{1} and Geoffroy Peeters\textsuperscript{2} \textasteriskcentered

\textsuperscript{1} IRCAM - Sorbonne Université
hadrien.foroughmand@ircam.fr
\textsuperscript{2} LTCI - Télécom Paris - Institut Polytechnique
geoffroy.peeters@telecom-paris.fr

Abstract. Tempo and genre are two inter-leaved aspects of music, genres are often associated to rhythm patterns which are played in specific tempo ranges. In this paper, we focus on the recent Deep Rhythm system based on a harmonic representation of rhythm used as an input to a convolutional neural network. To consider the relationships between frequency bands, we process complex-valued inputs through complex-convolutions. We also study the joint estimation of tempo/genre using a multitask learning approach. Finally, we study the addition of a second input branch to the system based on a VGG-like architecture applied to a mel-spectrogram input. This multi-input approach allows to improve the performances for tempo and genre estimation.

Keywords: Tempo estimation, genre classification, deep-learning, complex network, multitask, multi-input.

1 Introduction

Tempo is usually defined as the rate at which a listener taps while listening to a piece of music (Fraisse, 1982). The large number of works dedicated to its automatic estimation somehow demonstrates how important this task is in the Music Information Retrieval (MIR) community, but also that there is still room for improving its estimation.

The work on tempo estimation has for a long time concentrated on the development of hand-crafted systems, often based on the perceptual process used in human tempo inference. As an example, one of the earliest system proposed by (Scheirer, 1998) used a bank of band-pass filters followed by resonant comb-filters and a peak-picking process. Nearly a decade later, (Klapuri, Eronen, & Astola, 2006) still used resonant comb-filters but as input to a process to track the rhythm at several metric levels. (Gainza & Coyle, 2011) developed a hybrid

\textasteriskcentered\textsuperscript{*} This work was partly supported by European Union’s Horizon 2020 research and innovation program under grant agreement No 761634 (FuturePulse project).
multi-band decomposition using autocorrelation of onset functions across multiple frequency bands. These works highlighted the strong relationship between tempo and beat tracking, since tempo can be estimated as the period between successive beats. Overviews of these systems can be found in (Gouyon et al., 2006; Zapata & Gómez, 2011; Peeters, 2011).

The appearance of large datasets annotated into tempo or beat/downbeat positions has favored the development of data-driven systems where the machine learns from the annotated data using machine-learning (ML) algorithms. The first ML algorithms used were K-Nearest-Neighbors (KNN) (Seyerlehner, Widmer, & Schnitzer, 2007), Gaussian Mixture Model (GMM) (Xiao, Tian, Li, & Zhou, 2008; Peeters & Flocon-Cholet, 2012), Support Vector Machine (SVM) (Chen, Cremer, Lee, DiMaria, & Wu, 2009; Gkiokas, Katsouros, & Carayannis, 2012; Percival & Tzanetakis, 2014), bags of classifiers (Levy, 2011), Random Forest (Schreiber & Müller, 2017). Then deep learning (DL) became the most used ML algorithms in MIR. One of the first DL systems proposed for beat-tracking is the one of (Böck, Krebs, & Widmer, 2015) which used resonant comb-filters applied to the output of a Reccurent Neural Network (Bi-LSTM) that predicts the beat position inside the raw audio and then estimates the periodicity as the predicted tempo. Later, (Schreiber & Müller, 2018) proposed the first end-to-end DL system (although starting from the mel-spectrogram) for tempo estimation.

The mel-spectrogramm is used as input to a convolutional architecture that simulates a resonant comb filters. Their system considers the tempo prediction task as a classification task into tempo classes.

Recently, (Foroughmand & Peeters, 2019) proposed to combine the two types of systems in the so called “Deep Rhythm” system for tempo estimation and rhythm pattern/genre classification. It relies on a new harmonic representation of rhythm (the Harmonic Constant-Q Modulation - HCQM) used as input to a Convolutional Neural Network.

The HCQM represents the rhythm content of an audio signal in the spectral domain (as proposed by (Peeters, 2011)) and extends this representation with an extra dimension representing the harmonic series related to each frequency (as proposed by (Bittner, McFee, Salamon, Li, & Bello, 2017)). More precisely, onset strength functions are extracted in various acoustical frequency bands \( b \) (denoted by \( o_b(t) \)). Their temporal evolution are then independently represented by the modulus of a Constant-Q-Transform with frequencies in the modulation range (0Hz - 240 Hz) which directly correspond to the tempo range (0 - 240 BPM). This leads to a 2D representation (acoustic frequency bands \( b \), modulation frequency \( \Phi \)) which is extended to a third dimension \( h \) representing the harmonic series of each modulation frequency \( \Phi \). This representation is computed over segments of 8s with a frame analysis of hop-size 8 s. We denote by \( \tau' \) the time of the frames. The result is a 4-dimensional representation of size \( (\tau' \times \Phi \times b \times h) \).

The HCQM is then used as input to a Deep Convolutional Neural Network which architecture is similar to that of (Bittner et al., 2017) except that the last layer and the loss are configured to perform single-label classification as proposed by (Schreiber & Müller, 2018). We consider each tempo between 30
2 Deep Rhythm extensions

2.1 Complex Deep Rhythm

As mentioned above, in the original Deep Rhythm network, the modulations $\phi$ of each frequency bands $b$ are modeled independently through convolutional layers, i.e. the network does not consider the inter-relationship between the various frequency bands $b$. For example, the network will not be able to distinguish between a rhythm having a kick and a snare alternating over time and one having both simultaneously. This is due to the fact that the modulation is represented taking only the modulus of the CQT of the onset-strength-function at frequency bands $b$. In the case of the scale-transform combined with a modulation spectrum, (Marchand & Peeters, 2016b) showed that using the correlation between frequency bands allows a better estimation of the rhythm pattern and so of tempo. Because, the modulus does not preserve the time-information (which is contained in the phase of the CQT) we propose to replace it by the use of the complex CQT represented as Real and Imaginary part. The complex HCQM $H$ then has a Real and an Imaginary part (it can also be presented by doubling the dimensions: $(r' \times \Phi \times b \times 2h)$). To deal with this complex input, the network is then adapted to process complex values using complex convolutions and batch normalizations as proposed by (Trabelsi et al., 2017).

Complex Convolution. The complex input to the layers is denoted by $H = H_{\Re} + iH_{\Im}$ (with $H_{\Re}$ and $H_{\Im}$ its real and imaginary parts). The complex kernel matrix of the layer (which is the trainable parameter) is denoted by $K = K_{\Re} + iK_{\Im}$ (with $K_{\Re}$ and $K_{\Im}$ its real and imaginary parts). The complex convolution is then expressed as $K \ast H = (K_{\Re} \ast H_{\Re} - K_{\Im} \ast H_{\Im}) + i(K_{\Im} \ast H_{\Re} + K_{\Re} \ast H_{\Im})$. The output of each complex convolution layer is itself complex and is then used as input to the next complex convolution layer. All convolution layers of the original Deep Rhythm network are therefore replaced by complex convolution layers. Also, each complex convolution layers is followed by a complex batch normalization (as described in (Trabelsi et al., 2017)). After the last complex convolution, the resulting feature maps are flattened, hence by concatenating the real and imaginary output.

We illustrate this in Figure 1 where we only detail the complex convolution for the first convolution layer (the one applied to the input complex HCQM $H$).

2.2 Multitask learning: joint tempo and genre estimation

In (Foroughmand & Peeters, 2019), it is shown that the same network architecture, but with two different trainings and hence set of parameters, can be used to
achieve two different tasks: tempo estimation and genre classification\(^3\). We want here to exploit this multitasking aspect through the implementation of a single network which jointly estimates the tempo and the genre class. Some recent works have shown the effectiveness of this type of joint learning for fundamental frequency estimation of multiple instrument (Bittner, McFee, & Bello, 2018) or joint beat tracking and tempo estimation (Böck, Davies, & Knees, 2019).

**Architecture.** We illustrate the architecture of our multitask network on the right part of Figure 2. The architecture is the same as the original Deep Rhythm network (Foroughmand & Peeters, 2019) but extended. The extension starts from the flatten layer that follows the last convolutional layer. This vector then feed two independent branches, each with 2 fully connected layers ending with a softmax. One branch is dedicated to genre classification, the other to tempo estimation. The output of the first (of the second) has the same size as the number of genre to be detected (as the number of tempo classes).

**Losses.** To train the system we then simultaneously minimize two categorical cross-entropy losses: one for the genre classes \(L_{\text{genre}}\) and one for the tempo classes \(L_{\text{tempo}}\). Both are applied to the output of the networks (the softmax). We then minimize \(\mathcal{L} = L_{\text{genre}} + L_{\text{tempo}}\) (i.e. both losses are equally weighted).

### 2.3 Multi-input Network

The Deep Rhythm network was designed to represent the rhythm content of an audio track. As showed in (Gouyon et al., 2006), the tempo range and possible rhythm patterns are strongly correlated to the music genre of the track. The Deep Rhythm network however was designed to represent only the aspects related to rhythm, not to timbre. We therefore test an extension of the Deep-Rhythm by joining it with a second input dedicated to the representation of timbre.

The second input is a network now-commonly-used for audio tagging, the so-called Choi network (Choi, Fazekas, Sandler, & Cho, 2017). This network uses

\(^3\) In some of the datasets used for evaluation, genres are considered as rhythmic styles/patterns. To simplify the taxonomy, we refer to this task as genre classification.
mel-spectrograms as input to a VGG-like network with five convolutional layers of $(3 \times 3)$ kernels each connected to a max pooling layer $(2 \times 4), (2 \times 4), (2 \times 4), (3 \times 5), (4 \times 4)$ in order to reduce the size without losing information during training. In the original network, the last layer then predicts the tags. We skip this last layer here.

We denote by multi-input network a network with two input branches: one is Deep Rhythm, the second is Choi network. This multi-input network is then supposed to represent both the rhythm content and the timbre. The outputs of the two are then flattened and concatenated. The resulting vector is then processed by a dense layer of size 256. This is illustrated in Figure 2.

The multi-input network can be then be used with the multitask (as described above) or without the multitask. In the latter case, there is only one output dedicated to genre classification and the last layer is a dense layer of size $C$ (the number of genre classes). The activation is a softmax activation and we train by minimizing the categorical cross-entropy loss.

3 Evaluation

3.1 Time-varying tempo

Within the datasets we considered, some of the tracks have time-varying tempo. However only a single global ground-truth annotated tempo $T$ is provided for each track, implying that this tempo is the ground-truth for all the temporal-segments of the track. The original Deep Rhythm network process each audio frame $\tau'$ independently. We denote by $x_{\tau}$ the segment of the audio signal centered on time $\tau$ and of 8 s duration. During training, each $x_{\tau}$ is considered as an instance of the single global ground-truth tempo $T$ and the network trained accordingly. For testing, the output of the network (the softmax output) provides for each $x_{\tau}$ a tempo likelihood vector $p(T|x_{\tau})$ which represents the likelihood of each tempo $T$. The average over frame $\tau'$ of this vector is then computed, $p(T) = \int_{\tau'} p(T|x_{\tau'}) d\tau'$, and used to estimate the global tempo: $\hat{T} = \arg\max_T p(T)$.

**Oracle frame predictor.** We would like to know what would be the upper bound achievable by Deep Rhythm to predict $T$ from the succession of $p(T|x_{\tau'})$. We define an Oracle Frame Predictor which knows which is the best frame $\tau'$
to be used to predict $T$ denoted by $\tau^*$. The oracle defines the best frame as $\tau^* = \arg\min_{\tau}(T - \arg\max_T p(T|x_\tau))^2$. The final prediction of the oracle still uses the tempo likelihood vector to estimate the tempo (but only using the best frame): $\hat{T}^* = \arg\max_T p(T|x_{\tau^*})$. Typically, if the track only contains a single frame corresponding to $T$ and if the network is performing well, the Oracle should be able to find $\tau^*$ and the corresponding $\hat{T}^*$ would be a good estimation. In the opposite the average value $p(T)$ will be blurred and $\hat{T} = \arg\max_T p(T)$ would provide a wrong prediction. Hence $\hat{T}^*$ is an upper bound.

3.2 Tempo-only estimation

Protocol. To evaluate the performances on tempo estimation, we follow the same protocol as described in (Foroughmand & Peeters, 2019), i.e. we train the network on 3 datasets and evaluate the performances on 7 independent datasets. We also summarized the overall performances by indicating the results on the Combined dataset (the union of the 7 datasets). We indicate the results in terms of Accuracy (Acc) which is the exact estimation of tempo, Accuracy1 (Acc1) which is the estimation within a 4% window and Accuracy2 (Acc2) which is the estimation taking into account the predicted tempo at the 2nd and the third octave above and below within a 4% window.


Considered systems. The results are indicated in Table 1. (DR): original Deep Rhythm network; (Oracle-DR): (DR) using the Oracle Frame Prediction; (Cplx-DR): complex version of (DR) (part 2.1); (Oracle-Cplx-DR): Oracle Frame Prediction of (Cplx-DR).

Results. First, we see that (Oracle-DR) performs actually much better than (DR), in other words if we know at which frame to look at we would have much better results. Then comparing (DR) with (Cplx-DR), we see that in terms of Acc, (Cplx-DR) is more efficient for 4/7 datasets. In terms of Acc1 and Acc2, (Cplx-DR) is more efficient for 3/7 datasets and performs same or slightly better on overall (Combined dataset). Finally we see that the results are better with (Oracle-Cplx-DR) than (Oracle-DR) for a majority of the test datasets in terms of Acc and Acc1 and for all of them in terms of Acc2. This shows the advantages of the (Cplx-DR) method over (DR).

3.3 Joint tempo-genre, genre-only estimation

Protocol. It is not possible to perform a cross-dataset validation since genre classes are specific to each dataset. We only consider the datasets which are
Table 1. Tempo-only estimation results (large-scale cross-dataset validation).

<table>
<thead>
<tr>
<th></th>
<th>ACM</th>
<th>ISMIR04</th>
<th>Ballroom</th>
<th>Hainsworth</th>
<th>GTZAN</th>
<th>SMC</th>
<th>Giantsteps</th>
<th>Combined</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DR</td>
<td>38.1</td>
<td>24.7</td>
<td>73.1</td>
<td>40.1</td>
<td>37.7</td>
<td>6.0</td>
<td>26.5</td>
<td>38.9</td>
</tr>
<tr>
<td>Cplx-DR</td>
<td>40.2</td>
<td>28.0</td>
<td>70.2</td>
<td>38.7</td>
<td>42.0</td>
<td>6.9</td>
<td>26.1</td>
<td>40.2</td>
</tr>
<tr>
<td>Oracle-DR</td>
<td>50.7</td>
<td>34.2</td>
<td>80.9</td>
<td>56.8</td>
<td>49.2</td>
<td>10.1</td>
<td>35.09</td>
<td>49.5</td>
</tr>
<tr>
<td>Oracle-Cplx-DR</td>
<td>55.7</td>
<td>38.7</td>
<td>79.5</td>
<td>51.4</td>
<td>47.8</td>
<td>10.1</td>
<td>51.7</td>
<td>53.0</td>
</tr>
<tr>
<td>Acc1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DR</td>
<td>75.0</td>
<td>52.7</td>
<td>90.4</td>
<td>73.0</td>
<td>70.2</td>
<td>21.7</td>
<td>84.0</td>
<td>72.8</td>
</tr>
<tr>
<td>Cplx-DR</td>
<td>74.7</td>
<td>55.9</td>
<td>88.1</td>
<td>69.8</td>
<td>71.2</td>
<td>20.3</td>
<td>84.2</td>
<td>72.7</td>
</tr>
<tr>
<td>Oracle-DR</td>
<td>81.6</td>
<td>58.9</td>
<td>94.3</td>
<td>82.0</td>
<td>76.0</td>
<td>34.1</td>
<td>92.8</td>
<td>79.4</td>
</tr>
<tr>
<td>Oracle-Cplx-DR</td>
<td>86.5</td>
<td>65.2</td>
<td>92.8</td>
<td>85.6</td>
<td>80.3</td>
<td>37.8</td>
<td>97.7</td>
<td>83.3</td>
</tr>
<tr>
<td>Acc2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DR</td>
<td>96.2</td>
<td>88.0</td>
<td>97.1</td>
<td>83.3</td>
<td>88.8</td>
<td>35.0</td>
<td>98.0</td>
<td>90.8</td>
</tr>
<tr>
<td>Cplx-DR</td>
<td>96.5</td>
<td>87.3</td>
<td>97.7</td>
<td>82.4</td>
<td>90.7</td>
<td>31.8</td>
<td>97.9</td>
<td>91.0</td>
</tr>
<tr>
<td>Oracle-DR</td>
<td>98.5</td>
<td>89.5</td>
<td>98.3</td>
<td>88.7</td>
<td>92.6</td>
<td>55.3</td>
<td>98.9</td>
<td>93.9</td>
</tr>
<tr>
<td>Oracle-Cplx-DR</td>
<td>99.0</td>
<td>90.3</td>
<td>98.9</td>
<td>92.8</td>
<td>93.9</td>
<td>58.1</td>
<td>99.4</td>
<td>95.9</td>
</tr>
</tbody>
</table>

both annotated into tempo and into genre and perform for each a ten-fold cross validation (splitting each dataset into ten folds). For the tempo estimation, we indicate the same metric Acc1 as above. For the genre classification, we indicate the mean over class Recall since it is independent of class distribution.

Datasets. For the experiments, we used the following datasets each in a 10-fold cross-validation scenario: Extended Ballroom (Marchand & Peeters, 2016a) (3992 tracks and 9 genres), Ballroom (Gouyon et al., 2006) (698 tracks and 8 genres), MTG (Knees et al., 2015) (1823 tracks and 23 electronic genres; GTZAN (Marchand et al., 2015) (1000 tracks and 10 genres), Greek Dance (Holzapfel & Stylianou, 2011) (180 tracks and 6 greek music genres not annotated in tempo).

Considered systems. Since the metrics are different we indicate the results in two tables: Table 2b for tempo estimation and Table 2a for genre classification. (DR): Deep Rhythm network; (Cplx-DR): complex version of (DR); (MTL): multitask learning (part 2.2);(Cplx-MTL): complex version of (MTL); (MI): multi-input network(part 2.3); (Cplx-MI): complex version of (MI); (MI-MTL): the multi-input multitask learning; (Cplx-MI-MTL): complex version of (MI-MTL); For comparison purposes, we also provides the results with Choi model using the same protocol.

Results. For genre classification (Table 2a), all the models outperform (Choi) (except for the GTZAN dataset which is mainly defined by the timbre). Since (DR) performs better than (Cplx-DR) it seems that the added phase information of (Cplx-DR) prevents the training to generalize. Since (MTL) does not outperform (DR) it seems that jointly training them does not bring any benefit. The only benefit is in having only one model instead of two. In the opposite, the Multi-Input system (MI, Cplx-MI and MI-MTL) outperforms all the other ones for all datasets. Especially, its simpler form (MI) provides the best results for 4/5 datasets. Its complex version is the best method for the MTG dataset.

For tempo estimation (Table 2b), The (MI-MTL) method shows best results (for the ExtBallroom) while the (MI) method performs better for Ballroom and GTZAN. The results of the (MTL) methods are directly linked to the results
Table 2. Joint estimation results (10-fold Cross-Validation).

<table>
<thead>
<tr>
<th></th>
<th>ExtBallroom</th>
<th>Ballroom</th>
<th>Greek Dance</th>
<th>MTG</th>
<th>GTZAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Choi</td>
<td>72.1</td>
<td>60.1</td>
<td>38.1</td>
<td>21.7</td>
<td>74.2</td>
</tr>
<tr>
<td>DR</td>
<td>95.2</td>
<td>93.0</td>
<td>68.9</td>
<td>37.6</td>
<td>59.1</td>
</tr>
<tr>
<td>Cplx-DR</td>
<td>92.1</td>
<td>86.5</td>
<td>40.0</td>
<td>36.4</td>
<td>43.5</td>
</tr>
<tr>
<td>MTL</td>
<td>94.8</td>
<td>92.1</td>
<td>X</td>
<td>37.1</td>
<td>57.1</td>
</tr>
<tr>
<td>Cplx-MTL</td>
<td>92.4</td>
<td>86.1</td>
<td>X</td>
<td>39.8</td>
<td>44.0</td>
</tr>
<tr>
<td>MI</td>
<td><strong>96.5</strong></td>
<td><strong>94.2</strong></td>
<td><strong>69.4</strong></td>
<td>37.3</td>
<td><strong>74.3</strong></td>
</tr>
<tr>
<td>Cplx-MI</td>
<td>93.9</td>
<td>92.3</td>
<td>47.2</td>
<td>40.6</td>
<td>74.1</td>
</tr>
<tr>
<td>MI-MTL</td>
<td>96.2</td>
<td>93.0</td>
<td>X</td>
<td>39.6</td>
<td>67.2</td>
</tr>
<tr>
<td>Cplx-MI-MTL</td>
<td>94.6</td>
<td>91.9</td>
<td>X</td>
<td>40.3</td>
<td>66.0</td>
</tr>
</tbody>
</table>

(a) Genre results in terms of mean-over-class recall.

<table>
<thead>
<tr>
<th></th>
<th>ExtBallroom</th>
<th>Ballroom</th>
<th>MTG</th>
<th>GTZAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>DR</td>
<td>95.4</td>
<td>92.8</td>
<td>91.3</td>
<td>72.4</td>
</tr>
<tr>
<td>Cplx-DR</td>
<td>95.6</td>
<td>88.2</td>
<td>90.4</td>
<td>68.9</td>
</tr>
<tr>
<td>MTL</td>
<td>95.6</td>
<td>92.0</td>
<td>91.1</td>
<td><strong>73.2</strong></td>
</tr>
<tr>
<td>Cplx-MTL</td>
<td>94.4</td>
<td>89.4</td>
<td><strong>92.0</strong></td>
<td>69.2</td>
</tr>
<tr>
<td>MI</td>
<td>95.6</td>
<td><strong>94.1</strong></td>
<td>90.8</td>
<td><strong>73.2</strong></td>
</tr>
<tr>
<td>Cplx-MI</td>
<td>94.6</td>
<td>92.7</td>
<td>90.1</td>
<td>69.5</td>
</tr>
<tr>
<td>MI-MTL</td>
<td><strong>96.0</strong></td>
<td>92.2</td>
<td>91.3</td>
<td>71.5</td>
</tr>
<tr>
<td>Cplx-MI-MTL</td>
<td>95.7</td>
<td>92.4</td>
<td>91.6</td>
<td>68.5</td>
</tr>
</tbody>
</table>

(b) Tempo results in terms of Accuracy1.

of genre classification presented in table 2a since we perform a joint learning of the two tasks. It is interesting to remark that for the MTG dataset, (Cplx-MTL) has very good results both for tempo estimation and genre classification. Phase information may be more important to respect the properties’ of electronic music tempo and rhythm pattern.

4 Conclusion

In this paper we presented three main extensions of the Deep Rhythm network for tempo estimation and genre classification. First, we proposed the use of a complex-HCQM representation as input of a complex convolution neural network. This allows an improvement in term of tempo Acc but surprisingly not in terms of tempo Acc1 and Acc2 neither in terms of genre classification. Second, in order to better take into account the interdependencies between tempo and genre we proposed a multi-input network where a VGG-like network with mel-spectrogram input is added to represent timbre information along Deep Rhythm. We showed that this allows a improvement for both tasks. Third, we proposed a multi-task output where both tempo and genre are estimated jointly. With the Oracle frame prediction, we showed that their is still room for improve the tempo estimation. One of the future works will be to apply an attention mechanism system on top of the Deep Rhythm network to select automatically the temporal segment corresponding to the global tempo ground-truth annotation. We showed encouraging results since the results, are good for both tempo and genre estimation.
References


References

11