DANCE BEAT TRACKING FROM VISUAL INFORMATION ALONE

Fabrizio Pedersoli **Masataka Goto**
National Institute of Advanced Industrial Science and Technology (AIST), Japan
`{fabrizio.pedersoli, m.goto}@aist.go.jp`

ABSTRACT

We propose and explore the novel task of *dance beat tracking*, which can be regarded as a fundamental topic in the *Dance Information Retrieval* (DIR) research field. Dance beat tracking aims at detecting musical beats from a dance video by using its visual information without using its audio information (i.e., dance music). The visual analysis of dances is important to achieve general machine understanding of dances, not limited to dance music. As a subarea of Music Information Retrieval (MIR) research, DIR also shares similar goals with MIR and needs to extract various high-level semantics from dance videos. While audio-based beat tracking has been thoroughly studied in MIR, there has not been visual-based beat tracking for dance videos.

We approach dance beat tracking as a time series classification problem and conduct several experiments using a Temporal Convolutional Neural Network (TCN) using the AIST Dance Video Database. We evaluate the proposed solution considering different data splits based on either "dancer" or "music". Moreover, we propose a periodicity-based loss that considerably improves the overall beat tracking performance according to several evaluation metrics.

1. INTRODUCTION

One of core tasks of *Dance Information Retrieval (DIR)* [1] is to extract high-level semantics from dance videos, which could be similar to what Music Information Retrieval (MIR) tasks attempt to detect from music. For instance, some common tasks among the two research fields are: beat tracking, structure analysis, genre recognition, and automatic tagging. Although DIR shares similar objectives with MIR, DIR tasks are typically solved by analyzing video frames of dance motions (visual information). Of course, those tasks could also be solved by analyzing audio signals of dance music (audio information) when such signals are given or by analyzing both visual and audio information (multimodal information). As research on dance motions has not yet received much attention in the MIR community [1], the goal of this paper is to initiate *dance beat tracking* as a novel DIR task. Dance beat tracking is named to differentiate it from a standard task of music beat tracking and is defined as the task of detecting musical beats by using only visual analysis of video frames. Figure 1 shows an overview of our approach for dance beat tracking.

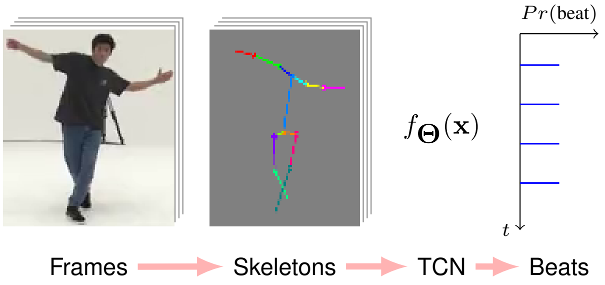

Figure 1. Our approach for dance beat tracking.

Since dance motions are usually related to the accompanying dance music, several dance characteristics can be inferred by joint analysis of motion and music. In fact, various researchers have already worked on multimodal aspects of dance music and motions [2–5]. In the MIR community, dance music such as traditional dances [6–9], electronic dance music [10–14], and ballroom dance music [15–17] has been a popular target of research. The literature on analysis of dance motions by using only video frames, however, is rather limited [1, 18, 19]. To the best of our knowledge, no work has focused on dance beat tracking using visual information of dance videos and evaluated its performance.

As music beat tracking is one of the most fundamental MIR tasks, dance beat tracking is also one of the most fundamental DIR tasks. *Beat* is the basic unit of time and can be used as a basis for further processing. For example, beat-synchronous analysis is effective and frequently used in the MIR community: music audio signals and dance videos could be divided into temporal sections associated with beats, which are then used to obtain beat-synchronous or beat-wise representations for various higher-level tasks [20–24]. Some direct applications of dance beat tracking systems would include automatic synchronization of dancing with music. Although dance videos usually have video frames synchronized with mu-

[1] Dance Information Retrieval is almost the same as *Dance Information Processing* [1] as Music Information Retrieval often means Music Information Processing/Research, but in this paper we use the term Dance Information Retrieval to focus on tasks analyzing dance information, which could be either audio or video.

Attribution: Fabrizio Pedersoli, Masataka Goto, "Dance Beat Tracking from Visual Information Alone", in *Proc. of the 21st Int. Society for Music Information Retrieval Conf.*, Montréal, Canada, 2020.

sic audio signals, there are irregular video files such as those in which the timing of video frames is out-of-sync with audio signals, and those in which a dancer is dancing without music or at different tempi. Dance beat tracking is useful for synchronizing and temporally-aligning (time-stretching) such video frames, or even identifying such out-of-sync videos.

Whether it is possible to automatically track beats of a dance video using only video frames is an open question [25]. To answer this question, we developed a dance beat tracking system that extracts skeletal body keypoints of a dancer from each video frame and uses Temporal Convolutional Neural Network (TCN) architectures to classify each frame as either a "beat" frame or a "non-beat" frame. In our experiments with a shared large-scale dance database, the AIST Dance Video Database [1], we found that it is possible to achieve dance beat tracking with the best F-measure performance of 61.20% and there is still large room for improvement. We also found that TCN architectures are more effective than architectures based on bidirectional Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNNs), and that the use of an additional loss term based on periodicity, which we propose in this paper, considerably improves beat tracking performances.

2. RELATED WORK

2.1 Audio-based music beat tracking

Initial work on music beat tracking for audio signals was based on spectral features, such as onset strength. By relying on these features, previous studies proposed multiple beat tracking agents [26, 27]. Further research on beat tracking was based on a dynamic programming framework [28–30]. Moreover, solutions based on the Kalman filter for detecting the beat locations were studied as well [31, 32]. Another popular way of approaching beat tracking was through the bar pointer model, originally proposed by Whiteley et al. [33], and improved by others [34, 35].

More recently, beat tracking research has largely adopted deep learning models to predict the beat positions, mainly by means of RNNs. The core idea of these deep learning solutions is to feed spectrogram frames, or features extracted from them, into an LSTM RNN. The network outputs the beat activation function, which is post-processed, usually by HMM decoding, to obtain the final beat locations. Previous work which adopted this processing setup is described in the papers of Durand et al. [36], Krebs et al. [37], Böck and Schedl [34], Böck et al. [38], and Cheng et al. [39].

RNN architectures have recently started to be replaced by more computationally efficient deep learning models such as Temporal Convolutional Networks (TCNs) [40, 41]. TCNs have also been used for beat tracking, as in the papers of Davies and Böck [42] and Böck et al. [43].

2.2 Visual or multimodal dance analysis related to beats

Guedes et al. [44] developed Max/MSP objects to control the tempo and rhythm of a music performance by using dance movements instead of a conducting baton. Although those objects extracted musically relevant rhythmic information from the video frames, they did not detect any dancer and their purpose was different from dance beat tracking. Ho et al. [45] developed a multimodal system that evaluated a street dance performance by estimating how well dance motions from a Kinect device correlated with musical beats. The system detected motion velocity drops as candidate motion beats, which did not necessarily have regular intervals, and then evaluated the synchronization between their candidates and musical beats.

Automatic dance motion generation for artificial characters, such as robot dancers and computer-graphics animation dancers, often needs beat-related audio-visual dance analysis. Ohkita et al. [46] presented a multimodal audio-visual beat tracking algorithm that enabled a robot to dance in synchronization with music and a human dancer. Audio-visual features were also used in the work of Shiratori et al. [47]. In their work, the authors proposed a method that synthesized a robot dance performance imitating the performance of a human dancer listening to the same music.

In addition, Davis et al. [18] presented a method that extracts visual rhythm from motions in video and aligns it with its musical counterpart. Visual rhythm was also at the base of a Xie et al. [48] article. In their work, the authors extracted several features from video frames, and proposed the use of an attention network to align them to the correspondent audio onsets through sequence labeling layers.

Although the above references do not directly deal with the task of dance beat tracking, they show its potential applications.

2.3 Orchestra conductor analysis

In the work of Huang et al. [49] body movements of the conductor were analyzed with the goal of inferring musical expressiveness. The authors proposed a multi-task learning model based on a bidirectional RNN to jointly identify dynamic, articulation, and phrasing cues of music from conducting movements.

A similar study on orchestral director movement was described in Schmidt et al. [50]. Musical beats were detected in correspondence of a director's hand's velocity peaks, where the movements were recorded by using a wrist e-watch. The paper of Schramm et al. [51] also focused on a director's hand gestures acquired by Kinect with the purpose of detecting duple, triple, and quadruple patterns by using a probabilistic Dynamic Time Warping framework.

Although motion analysis of dancers and conductors could have some technical similarities, conductor motions tend to give more explicit cues for musical beats. On the other hand, since dancer motions do not necessarily correlate with musical beats, dance beat tracking is harder than conductor analysis in general.

3. PROPOSED SYSTEM

Inspired by the success attained in audio-based music beat tracking, we address dance beat tracking as a sequence classification problem. According to this framework, a classifier takes as input a sequence of observations $x_{1:T} = \{x_1, \ldots, x_T\}$, and produces an output of the same length $y_{1:T} = \{y_1, \ldots, y_T\}$ where each observation is classified into "beat" ($y = 1$) or "non-beat" ($y = 0$), by taking into account past observations. In our application each observation consists of a video frame.

Two main technical challenges of dance beat tracking are modeling long time sequences and extracting meaningful descriptors from video frames. The former challenge has been successfully tackled in recent years initially by using RNNs and then by using TCNs [41]. TCNs, as explained in Section 3.1, are deep learning architectures based on stacks of causal dilated convolutions [40] which serve the same purpose as an RNNs while offering several advantages over them. TCNs are more computationally efficient since convolutions can be easily parallelized. In addition, TCNs do not suffer from exploding/vanishing of gradients, which is a major drawback when dealing with long time sequences such as the one used in this application. The latter challenge can be dealt with by using standard computer vision convolutional networks to extract meaningful features from the video frames.

We thus chose to use a TCN as a sequence classifier. From each video frame (60 fps) we extract the (x, y) position of dancer body keypoints by using the OpenPose framework [52]. Thus, we represent a video as a sequence of keypoints (Section 4.2). Although extracting the body keypoints requires preprocessing of the dance video, we found this description to be at the same time powerful for effectively modeling dance movements. We do not directly use video frame pixels since it is difficult to prepare a training dataset with sufficient variations of dancer clothes, colors, and backgrounds, and using video frame pixels limits generalization of the model.

3.1 Temporal convolutional networks

TCNs process the input x_n by taking into account only the past information, and produces an output y_n of the same length as the input. To achieve this goal, 1D causal[2] convolutional layers with the "same" padding[3] are used. In order to model long time sequences, the network must have a large receptive field. We therefore use a stack of *dilated* convolutional layers so that we can increase the receptive field while maintaining the same (small) kernel size of each layer. Each layer of the stack has the same number of features. The dilation factor increases in an exponential way at each convolution stack. More precisely, at a particular network level i, the dilation factor d is 2^i.

Stacking more dilated convolutional layers to model longer time sequences results in a deeper network, which is harder to train. It has been shown that for very deep networks, training on residual connections ensures a better gradient flow which allows more effective training [53].

In our work we use a TCN in the configuration proposed by Bai et al. [41]. The authors proposed a deep learning architecture which is composed of several TCN residual blocks. Each TCN residual block is composed of two dilated causal convolutional layers (of same dilation factor) and Rectified Linear Units activation (ReLU [54]). In order to accelerate the training of the model, a weight normalization layer [55] is placed after each dilated causal convolutional layer. In addition, a spatial dropout layer [56] is utilized after activations so that overfitting is mitigated. Finally, an optional 1×1 convolution is used on the identity path to match the feature map size of the input to the output when these two differ.

3.2 Network training

We train our model by using the Adam optimizer [57] with default PyTorch parameters, a learning rate of 0.5×10^{-3}, and batch size of 32. Training is stopped when the loss on a validation dataset does not improve for subsequent 30 epochs. The best model is then selected according to the best performance achieved on the validation dataset. For data augmentation, Gaussian noise $\mathcal{N}(0, 1)$ is added to the keypoint delta values before feeding them to the network.

3.2.1 Loss function

The basic loss criterion used to train the network is the cross-entropy, L_{ce}. Giving an input sequence of M observations, for each observation $m = 0, 1, \ldots, M-1$, the model outputs a softmax distribution $\hat{\mathbf{y}}_m$ over two classes: "beat" and "non-beat". Since probabilities of "beat" and "non-beat" in the ground truth are largely unbalanced, we weight the cross-entropy loss with a weight vector $\mathbf{w}$ of empirically chosen values: 1 and 0.1 respectively for "beat" and "non-beat". Thus, given a particular sample pair n of true sequence $\mathbf{y}^{(n)}$ and predicted sequence $\hat{\mathbf{y}}^{(n)}$, the weighted cross-entropy loss is defined as follows:

$$L_{ce}^{(n)} = -\mathbf{w} \sum_{m=0}^{M-1} \mathbf{y}_m^{(n)} \log\left(\hat{\mathbf{y}}_m^{(n)}\right). \qquad (1)$$

We also propose an additional loss term that takes account of *periodicity*, L_p. It is reasonable to assume that a dance is characterized by repeating patterns which are correlated to the music beats. By using the periodicity loss we inform the model that predictions made in correspondence of multiples of the music tempo T should be considered similar. In a training dataset the music tempo is known a priori and constant throughout the audio clips. Note that this additional loss term is used only during training. The periodicity loss is simply the summation of the absolute difference of predictions made at multiples of the music

[2] The result of the convolution at time $t = T$ is obtained using inputs at and before $t \leq T$.

[3] For a kernel $h_1, \ldots, h_M$ the "same" convolution padding length is $M - 1$.

BPM	Street Dance Genre									
	BR	PO	LO	WA	MH	LH	HO	KR	JS	JB
T_0	21	23	23	23	24	25	23	23	21	22
T_1	23	23	23	23	24	23	23	23	23	24
T_2	23	24	23	24	24	23	24	23	23	22
T_3	23	24	23	24	23	24	24	25	21	23
T_4	24	23	25	23	23	23	23	23	23	23
T_5	24	24	24	24	23	23	22	24	23	24

Table 1. Video counts of the AIST Dance Video Database subset used in our work.

tempo kT:

$$L_p^{(n)} = \left\| \sum_{\substack{k=0 \\ k'=k+1 \\ k''=k'+1}}^{N_b-3} \hat{\mathrm{y}}_{kT:k'T}^{(n)} - \hat{\mathrm{y}}_{k'T:k''T}^{(n)} \right\|_1 , \tag{2}$$

where N_b is the number of beats contained in the ground truth sequence. The output sequence is zero-padded to a multiple of ground truth tempo.

We get the total loss by adding together the weighted cross-entropy and a scaled version of the periodicity loss. The scale parameter α is empirically chosen by grid-search. Finally, we take the average among the training batch of N samples:

$$L = \frac{1}{N} \sum_{n=0}^{N-1} L_{ce}^{(n)} + \alpha L_p^{(n)}. \tag{3}$$

4. DATASET AND DATA PROCESSING

The AIST Dance Video Database [1] is a large-scale collection of dance videos. This database includes 10 street dance genres: “Break” (BR), “Pop” (PO), “Lock” (LO), “Waack” (WA), “Middle Hip-Hop” (MH), “LA-style Hip-Hop”(LH), “House” (HO), “Krump” (KR), “Street Jazz” (JS), and “Ballet Jazz” (JB). For each dance genre, 6 musical pieces of different tempi are used. In particular, the music tempi are: $T_0 = 80$, $T_1 = 90$, $T_2 = 100$, $T_3 = 110$, $T_4 = 120$, and $T_5 = 130$ beats per minute (bpm) for all the genres, except the “House” genre whose tempi are : $T_0 = 110$, $T_1 = 115$, $T_2 = 120$, $T_3 = 125$, $T_4 = 130$, and $T_5 = 135$ bpm.

In this work we consider dance videos where a single dancer is performing (“Basic Dance” and “Advanced Dance” in the database), and we use the frontal camera as the source of information [4] . The total number of dance videos considered in our experimental evaluation is 1396. The resolution of the videos is 1920×1080 pixels. Table 1 shows a detailed breakdown of our dataset.

4.1 Data splits

We split the data according to “music” and “dancer”. For each of the split configurations, we randomly split the data samples in training, validation, and test datasets with percentages of $70\,\%$, $20\,\%$, and $10\,\%$, respectively. In order to make balanced splits, we adopt the following strategy. In the case of the “music” data split, for each of the 60 music clips, we select the correspondent videos, and randomly split them according to the aforementioned train/validation/test percentages. We then concatenate the individual micro splits to obtain the final “music” train/validation/test splits. The same process is executed according to the individual dancer for compiling the “dancer” data splits.

4.2 Data preprocessing

All the dance videos are preprocessed by extracting the body keypoints by using the OpenPose framework [52]. We use the `BODY_25` pose model which represents the human body by 25 skeletal keypoints. However, in our application a subset of the `BODY_25` keypoints was problematic to detect with high reliability, and therefore it was discarded from the body pose. These problematic keypoints correspond to “eyes”, “ears”, “nose”, “heels”, and “big/small toe”. After removing these keypoints we end up with a total of 13 keypoints: “neck”, “shoulders”, “elbows”, “wrists”, “mid hip”, “hips”, “knees”, and “ankles”. However, missed detections of body keypoints can still happen. The missed detections are recovered by spline interpolation.

The body keypoints are defined by their pixel position (x, y) within a video frame on the basis of the OpenPose output. However, this representation has the drawback of not being invariant to the dancer’s position and of being dependent on the dancer’s body size. To overcome these issues, we convert the absolute $(x_n, y_n)^{(i)}$ position of the i-th keypoint at time n into its displacement (delta values) in time:

$$(\Delta x_n, \Delta y_n)^{(i)} = (x_n - x_{n-1},\ y_n - y_{n-1})^{(i)}. \tag{4}$$

4.3 Beat activation processing

The output of network is the beat activation function; i.e., for each video frame, the model predicts its probability of being a “beat” frame. To obtain the final beat positions, a postprocessing of the beat activation function is required. In our work we use the algorithm proposed by Krebs et al. [35], which is based on HMM decoding.

5. EXPERIMENTAL SETUP

We report performance results by using several beat tracking metrics typical of music and by following the practice described in Davies et al. [58]. For our experimental evaluation, we make use of the `mir_eval` [59] software package [5] .

For the evaluation we consider the first 420 frames ($7\,\mathrm{s}$) of each video. In addition, the first $1\,\mathrm{s}$ of the predicted beat sequence is discarded when computing the performance results.

[4] The frontal view of the dancer is the most reliable for detecting the body keypoints because body part occlusions are minimized by this visual perspective

[5] `https://github.com/craffel/mir_eval`

5.1 Model selection

In order to select the best-performing configuration of the model, we conduct hyperparameter grid-search on the number of stacks $N_{\text{stack}} \in \{3, \ldots, 12\}$ and the number of convolutional features $N_{\text{feat}} \in \{32, 64, 128, 256\}$. Since no pooling is involved in the TCN architecture, the number of convolutional features is kept constant among the entire stack. The extreme values (minimum and maximum) of these hyperparameters are chosen in a way that the model would respectively under-fit and over-fit the validation dataset.

The training is stopped when the loss on the validation dataset does not decrease for 30 successive epochs. According to the considered performance metric, the best-performing model on the validation dataset is selected. This model is then evaluated on the test dataset.

The hyperparameter grid-search reported that $N_{\text{stack}} = 7$ and $N_{\text{feat}} = 128$ yields, in the majority of the cases, the best TCN. The evaluation is done for both "dancer" and "music" data splits. We denote this configuration as TCN^{7}_{128} and we use it as our baseline.

5.2 Periodicity loss ablation study

With the purpose of assessing the usefulness of the proposed periodicity loss term (L_p), we conduct an ablation study using TCN^{7}_{128} as the baseline.

Additional hyperparameter grid-search is done for finding the best value of α, see equation (3). This parameter weights the contribution of L_p to the overall loss and must be carefully chosen by empirical evaluation. The tested values are: $\alpha \in \{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1\}$. The grid-search is done for both "dancer" and "music" data splits by training the TCN^{7}_{128} according to the procedure previously described.

6. EXPERIMENTAL RESULTS

In the first part of this section we report the baseline performance of TCN^{7}_{128}. We also compare the baseline performance between TCN and LSTM architectures and show that the TCN architecture is the best in our application. Finally, we elaborate on the results of the ablation study of the periodicity loss term by showing its effectiveness.

We report the performance in terms of different metrics. More specifically we consider: F-measure (F), Cemgil's score (Cem), and continuity base scores [58]. The continuity scores are: Correct Metrical Level continuity required/not required (CML$_{c/t}$), and Allowed Metrical Level continuity required/not required (AML$_{c/t}$).

6.1 Baseline loss

Table 2 reports the performance results of TCN^{7}_{128} model trained with L_{ce}. Results are subdivided according to the "dancer" and "music" data splits and are reported as percentages correct.

Examining the overall results, we acknowledge at first look the difficulty of this new task. A similar conclusion about music beat tracking is drawn in when percussion is not present in the analyzed audio signal. We find that in a dance, the body movements cannot stress the tempo as efficiently as percussive sounds do. Therefore, a lower performance is reasonably expected in dance beat tracking than in music beat tracking (Section 6.3).

Split	CML$_c$	CML$_t$	AML$_c$	AML$_t$	Cem	F
Dancer	44.28	46.93	47.27	49.04	52.92	55.02
Music	40.14	39.71	44.84	47.53	47.43	53.02

Table 2. TCN^{7}_{128} results trained with the baseline loss.

Split	CML$_c$	CML$_t$	AML$_c$	AML$_t$	Cem	F
Dancer	38.66	41.49	56.23	53.03	32.17	46.56
Music	27.62	32.45	43.13	46.32	28.35	39.18

Table 3. $b\mathrm{LSTM}^{4}_{128}$ results trained with the baseline loss.

In addition, we notice that the performance on the dancer data split is $\approx 4\,\%$ higher (for all the metrics) than the performance on the music data split. The dancing style that characterizes each dancer seems to be easier to capture by the network, while the difference in choreographies for the same music piece is less effectively captured by the model.

In more detail, for both of the data splits, we see that the continuity scores obtain lower performance. Specifically, CML is the least performing metric and is followed by AML. The performance gap between CML and AML ($\approx 3\,\%$ for dancer and $\approx 6\,\%$ for music) suggests that the model tends to detect beats at half or double the correct tempo. An improvement of CML/AML scores is achieved when continuity is not required (CML$_t$ and AML$_t$). The F-measure attains the highest performance for both of the data splits, while the Cemgil's score shows a slight decrease, which is more evident for the music data split. This performance decrease indicates that the model tends to detect beats with a slight offset with respect to the ground truth position.

The main conclusions of this experimental section are as follows. (1) Detecting beats is more difficult for the "music" split. (2) Detected beats are prone to errors such as beat positions with less continuity, with a half or double tempo, and with a small deviation from the correct beats.

6.1.1 Comparison with LSTM

We provide a baseline comparison with a bidirectional LSTM RNN, whose performance results are reported in Table 3. Also in this case, we conduct similar hyperparameter grid-search as done for the TCN. In particular, we tested $N_{\text{stack}} \in \{1, 2, \ldots, 6\}$ and $N_{\text{units}} \in \{32, 64, 128, 256\}$, and found that in average for the different performance metrics, the best performing configuration is $N_{\text{stack}} = 4$ and $N_{\text{units}} = 128$. We refer to this model specification as $b\mathrm{LSTM}^{4}_{128}$.

From Table 3 we notice that for both "dancer" and "music" data splits the $b\mathrm{LSTM}^{4}_{128}$ performs worse than the

Loss	CML$_c$	CML$_t$	AML$_c$	AML$_t$	Cem	F
L_{ce}	44.28	46.93	47.27	49.04	52.92	55.02
$L_{ce}+\alpha L_p$	**53.05**	**54.30**	**55.23**	**57.64**	**59.02**	**61.20**

Table 4. Performance results for the "dancer" data split using the proposed loss with $\alpha = 0.05$.

Loss	CML$_c$	CML$_t$	AML$_c$	AML$_t$	Cem	F
L_{ce}	40.14	39.71	44.84	47.53	47.43	53.02
$L_{ce}+\alpha L_p$	**46.50**	**48.33**	**48.27**	**50.87**	**54.27**	**58.25**

Table 5. Performance results for the "music" data split using the proposed loss with $\alpha = 0.1$.

TCN $^{7}_{128}$ in terms of almost all the considered metrics. The performance gap is quite significant for Cemgil's score and the F-measure. Notably, the AML scores are comparable, or even better ("dancer" split), than the TCN. In this particular instance, the recurrent nature of the tested model is helpful in detecting beat locations that are regularly spaced according to musical metric (half/double).

6.2 Proposed loss

From Tables 4 and 5 we assess the benefit introduced by the proposed periodicity loss term. Indeed, the proposed loss considerably improves each of the performance metrics and does so for both of the data splits. The improvement averages $\approx 7.5\,\%$ points for the dancer split and about $\approx 5.5\,\%$ points for the music split. We found by means of grid-search that the best value for the hyperparameter α is: 0.05 for the "dancer" split and 0.1 for the "music" split.

The performance gap between "dancer" and "music" data splits is also present in this case. Moreover, a similar trend of the performance scores also occurs in the combined loss experiments. In fact, sorted in ascending order of the achieved performances, the metrics are: CML, AML, Cemgil's score, and F-measure.

In the case of the "dancer" data split (see Table 4), the continuity metrics are the most improved metrics. With an improvement of $\approx 8\,\%$ points, the periodicity loss term is beneficial for detecting beats at the correct time spacing. Although less improved, Cemgil's score and the F-measure still indicate an important boost in performance by $\approx 6\,\%$ points. This means that the proposed loss is helpful for obtaining more accurate detection of beats.

In the case of the "music" data split (see Table 5), the performance improvement, although slightly less evident than in the case of the dancer split, is still consistent. For the music data split, the proposed loss improves all the performance metrics by an average of $\approx 5.5\,\%$ points. The behavior similar to the results of the dancer data split is also observed in this case. However, for the music data split we see that improvement for AML metrics is relatively low: $\approx 3\,\%$ points. In this case CML and AML results are more aligned, with the latter being $\approx 2\,\%$ points better.

To summarize, the main conclusions of this experimental section are as follows. (1) The proposed loss based on periodicity improves performance considerably. (2) The proposed loss helps in detecting beat locations which are more aligned with the correct music metric (or half/double of it). (3) The performance gap between the two data splits is still present, although slightly mitigated.

CML$_c$	CML$_t$	AML$_c$	AML$_t$	Cem	F
81.34	81.34	94.27	94.89	73.78	88.98

Table 6. Average performance results on the audio clips.

6.3 Audio-based music beat tracking

In Table 6, we report the beat tracking performance achieved on the audio clips of the AIST Dance Video Database [1] for comparison. We use the model of Böck et al. [34] in combination with the same HMM postprocessing module used for dance beat tracking.

Since the results are much better than those in Tables 4 and 5, music beat tracking is an easier task than dance beat tracking in our experiments. This is expected since the audio clips chosen for the dancing purpose have usually distinctive beats.

7. CONCLUSION

Our main contributions are as follow. (1) We propose the task of dance beat tracking which is characterized by the novelty of using visual information, in the form of motion patterns, for detecting musical beats. (2) We provide a baseline evaluation on the AIST Dance Video Database [1] considering data splits based on "music" and "dancer". By comparing the results based on those data splits, we gain deeper insights about the dance beat tracking task. In addition, we also provide a performance comparison of deep learning architectures commonly used for time series classification. In this regard, we show that TCNs outperform bidirectional LSTMs for dance beat tracking. (3) We propose the periodicity loss term, which is scaled and added to the baseline cross-entropy loss. This novel loss term takes into account motion repetitions in relationship to beats and considerably improves the beat tracking performance.

Detecting musical beats from video frames revealed to be a challenging task, and it encourages further research in this direction in order to improve the performance results. In fact, the relationship between dancer body movements and musical beats is difficult to capture due to high variability of motion patterns among different dancers and different choreographies. This challenge is similarly faced in MIR when trying to detect beats from non-percussive music. In future work we plan to investigate deep learning architectures that can directly process video frames without needing to extract the body keypoints ahead of time. Future work will also include investigation of whether it is possible for human beings to visually track beats of a dance video without listening to the accompanying sounds, and will compare machine and human performances.

8. ACKNOWLEDGMENTS

This work was supported in part by JST ACCEL Grant Number JPMJAC1602, Japan.

9. REFERENCES

[1] S. Tsuchida, S. Fukayama, M. Hamasaki, and M. Goto, “AIST Dance Video Database: Multi-genre, multi-dancer, and multi-camera database for dance information processing,” in *Proceedings of the 20th International Society for Music Information Retrieval Conference (ISMIR)*, 2019, pp. 501–510.

[2] J. Li, H. Peng, H. Hu, Z. Luo, and C. Tang, “Multimodal information fusion for automatic aesthetics evaluation of robotic dance poses,” *International Journal of Social Robotics*, pp. 1–16, 2019.

[3] F. Ofli, E. Erzin, Y. Yemez, and A. M. Tekalp, “Multimodal analysis of dance performances for music-driven choreography synthesis,” in *Proceedings of the 34th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 2466–2469.

[4] A. Camurri, B. Mazzarino, M. Ricchetti, R. Timmers, and G. Volpe, “Multimodal analysis of expressive gesture in music and dance performances,” in *Proceedings of the 4th International Gesture Workshop*. Springer, 2003, pp. 20–39.

[5] G. Qian, F. Guo, T. Ingalls, L. Olson, J. James, and T. Rikakis, “A gesture-driven multimodal interactive dance system,” in *Proceedings of the 5th IEEE International Conference on Multimedia and Expo (ICME)*, vol. 3, 2004, pp. 1579–1582.

[6] P. Beauguitte, B. Duggan, and J. D. Kelleher, “A corpus of annotated Irish traditional dance music recordings: Design and benchmark evaluations,” in *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 2016, pp. 53–59.

[7] B. Duggan, B. O’Shea, M. Gainza, and P. Cunningham, “Machine annotation of sets of traditional Irish dance tunes,” in *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, 2008, pp. 401–406.

[8] A. Holzapfel and E. Benetos, “The sousta corpus: Beat-informed automatic transcription of traditional dance tunes,” in *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 2016, pp. 531–537.

[9] L. Risk, L. Mok, A. Hankinson, and J. Cumming, “Melodic similarity in traditional french-canadian instrumental dance tunes.” in *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, 2015, pp. 93–99.

[10] N. Collins, “Influence in early electronic dance music: An audio content analysis investigation,” in *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp. 1–6.

[11] T. Kell and G. Tzanetakis, “Empirical analysis of track selection and ordering in electronic dance music using audio feature extraction,” in *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, 2013, pp. 505–510.

[12] M. Panteli, N. Bogaards, and A. K. Honingh, “Modeling rhythm similarity for electronic dance music.” in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, 2014, pp. 537–542.

[13] H. Schreiber and M. Müller, “A crowdsourced experiment for tempo estimation of electronic dance music.” in *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, 2018, pp. 409–415.

[14] K. Yadati, M. Larson, C. C. Liem, and A. Hanjalic, “Detecting drops in electronic dance music: Content based approaches to a socially significant music event,” in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, 2014, pp. 143–148.

[15] S. Dixon, E. Pampalk, and G. Widmer, “Classification of dance music by periodicity patterns,” in *Proceedings of the 4th International Conference of Music Information Retrieval (ISMIR)*, 2003.

[16] F. Gouyon and S. Dixon, “Dance music classification: A tempo-based approach,” in *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR)*, 2004.

[17] P. Knees, A. Faraldo, P. Herrera, R. Vogl, S. Böck, F. Hörschläger, and M. Le Goff, “Two data sets for tempo estimation and key detection in electronic dance music annotated from user corrections.” in *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, 2015, pp. 364–370.

[18] A. Davis and M. Agrawala, “Visual rhythm and beat,” *ACM Transaction on Graphics*, vol. 37, no. 4, pp. 122–1, 2018.

[19] Y. Xie, H. Wang, Y. Hao, and Z. Xu, “Visual rhythm prediction with feature-aligning network,” in *Proceedings of the 11th International Conference on Machine Vision Applications (MVA)*, 2019, pp. 1–6.

[20] G. Griffin, Y. E. Kim, and D. Turnbull, “Beat-sync-mash-coder: A web application for real-time creation of beat-synchronous music mashups,” in *Proceedings of the 35th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 437–440.

[21] R. Nishikimi, E. Nakamura, K. Itoyama, and K. Yoshii, "Musical note estimation for f0 trajectories of singing voices based on a bayesian semi-beat-synchronous hmm." in *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 2016, pp. 461–467.

[22] S.-C. Pei and N.-T. Hsu, "Instrumentation analysis and identification of polyphonic music using beat-synchronous feature integration and fuzzy clustering," in *Proceedings of the 34th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 169–172.

[23] Y. Panagakis and C. Kotropoulos, "Music structure analysis by ridge regression of beat-synchronous audio features." in *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp. 271–276.

[24] D. P. Ellis, C. V. Cotton, and M. I. Mandel, "Cross-correlation of beat-synchronous representations for music similarity," in *Proceedings of the 33rd IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008, pp. 57–60.

[25] C. C. Stauffer, J. Haldemann, S. J. Troche, and T. H. Rammsayer, "Auditory and visual temporal sensitivity: evidence for a hierarchical structure of modality-specific and modality-independent levels of temporal information processing," *Psychological research*, vol. 76, no. 1, pp. 20–31, 2012.

[26] M. Goto and Y. Muraoka, "A beat tracking system for acoustic signals of music," in *Proceedings of the 2nd ACM International Conference on Multimedia*, 1994, pp. 365–372.

[27] 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.

[28] D. P. Ellis, "Beat tracking by dynamic programming," *Journal of New Music Research*, vol. 36, no. 1, pp. 51–60, 2007.

[29] A. M. Stark, M. E. Davies, and M. D. Plumbley, "Real-time beat-synchronous analysis of musical audio," in *Proceedings of the 12th International Conference on Digital Audio Effects (DAFx)*, 2009, pp. 299–304.

[30] F.-H. F. Wu, T.-C. Lee, J.-S. R. Jang, K. K. Chang, C.-H. Lu, and W.-N. Wang, "A two-fold dynamic programming approach to beat tracking for audio music with time-varying tempo," in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 191–196.

[31] A. T. Cemgil, B. Kappen, P. Desain, and H. Honing, "On tempo tracking: Tempogram representation and Kalman filtering," *Journal of New Music Research*, vol. 29, no. 4, pp. 259–273, 2000.

[32] Y. Shiu, N. Cho, P.-C. Chang, and C.-C. J. Kuo, "Robust on-line beat tracking with Kalman filtering and probabilistic data association (kf-pda)," *IEEE transactions on consumer electronics*, vol. 54, no. 3, pp. 1369–1377, 2008.

[33] N. Whiteley, A. T. Cemgil, and S. J. Godsill, "Bayesian modelling of temporal structure in musical audio." in *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR)*, 2006, pp. 29–34.

[34] S. Böck and M. Schedl, "Enhanced beat tracking with context-aware neural networks," in *Proceedings of the 14th International Conference on Digital Audio Effects (DAFx)*, 2011, pp. 135–139.

[35] F. Krebs, S. Böck, and G. Widmer, "An efficient state-space model for joint tempo and meter tracking." in *Proceedings of the 16th International Society for Music Information Retrieval Conference (ISMIR)*, 2015, pp. 72–78.

[36] S. Durand, J. P. Bello, B. David, and G. Richard, "Downbeat tracking with multiple features and deep neural networks," in *Proceedings of the 40th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 409–413.

[37] F. Krebs, S. Böck, M. Dorfer, and G. Widmer, "Downbeat tracking using beat synchronous features with recurrent neural networks." in *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 2016, pp. 129–135.

[38] S. Böck, F. Krebs, and G. Widmer, "Joint beat and downbeat tracking with recurrent neural networks," in *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*, 2016, pp. 255–261.

[39] T. Cheng, S. Fukayama, and M. Goto, "Convolving Gaussian kernels for RNN-based beat tracking," in *Proceedings of the 26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 1919–1923.

[40] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[41] S. Bai, Z. J. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[42] M. E. Davies and S. Böck, "Temporal convolutional networks for musical audio beat tracking," in *Proceedings of the 27th European Signal Processing Conference (EUSIPCO)*, 2019, pp. 1–5.

[43] S. Böck, M. E. Davies, and P. Knees, "Multi-task learning of tempo and beat: Learning one to improve the other," in *Proceedings of the 20th International Society*

for Music Information Retrieval Conference (ISMIR), 2019, pp. 486–493.

[44] C. Guedes and C. Branco, “Extracting musically-relevant rhythmic information from dance movement by applying pitch-tracking techniques to a video signal,” in *Proceedings of the 12th Sound and Music Computing Conference (SMC)*, 2006, pp. 25–33.

[45] C. Ho, W.-T. Tsai, K.-S. Lin, and H. H. Chen, “Extraction and alignment evaluation of motion beats for street dance,” in *Proceedings of the 38th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 2429–2433.

[46] M. Ohkita, Y. Bando, Y. Ikemiya, K. Itoyama, and K. Yoshii, “Audio-visual beat tracking based on a state-space model for a music robot dancing with humans,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5555–5560.

[47] T. Shiratori, A. Nakazawa, and K. Ikeuchi, “Synthesizing dance performance using musical and motion features,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2006, pp. 3654–3659.

[48] G. Xia, J. Tay, R. Dannenberg, and M. Veloso, “Autonomous robot dancing driven by beats and emotions of music,” in *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, 2012, pp. 205–212.

[49] Y.-F. Huang, T.-P. Chen, N. Moran, S. Coleman, and L. Su, “Identifying expressive semantics in orchestral conducting kinematics,” in *Proceedings of the 20th International Society for Music Information Retrieval Conference (ISMIR)*, 2019.

[50] D. Schmidt, A. Smailagic, R. B. Dannenberg, D. P. Siewiorek, and B. Bugge, “Learning an orchestra conductor’s technique using a wearable sensor platform,” in *Proceedings of the 11th IEEE International Symposium on Wearable Computers*, 2007, pp. 113–114.

[51] R. Schramm, C. R. Jung, and E. R. Miranda, “Dynamic time warping for music conducting gestures evaluation,” *IEEE Transactions on Multimedia*, vol. 17, no. 2, pp. 243–255, 2014.

[52] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, “OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[53] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[54] V. Nair and G. E. Hinton, “Rectified linear units improve restricted Boltzmann machines,” in *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010, pp. 807–814.

[55] T. Salimans and D. P. Kingma, “Weight normalization: A simple reparameterization to accelerate training of deep neural networks,” in *Advances in neural information processing systems*, 2016, pp. 901–909.

[56] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[57] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

[58] M. E. Davies, N. Degara, and M. D. Plumbley, “Evaluation methods for musical audio beat tracking algorithms,” *Queen Mary University of London, Centre for Digital Music, Tech. Rep. C4DM-TR-09-06*, 2009.

[59] 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 *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*, 2014.