Towards Movement Generation with Audio Features

Benedikte Wallace[1], **Charles P. Martin**[2], **Jim Torresen**[1] and **Kristian Nymoen**[1]

[1]RITMO Center, University of Oslo
[2]Australian National University
benediwa@ifi.uio.no

Abstract

Sound and movement are closely coupled, particularly in dance. Certain audio features have been found to affect the way we move to music. Is this relationship between sound and movement something which can be modelled using machine learning? This work presents initial experiments wherein high-level audio features calculated from a set of music pieces are included in a movement generation model trained on motion capture recordings of improvised dance. Our results indicate that the model learns to generate realistic dance movements which vary depending on the audio features.

Introduction

Expressive movement is an intrinsic part of human life. Hand gestures, body language as well as dance can efficiently convey an emotional state. Simple movement patterns such as gait or arm movement can allow us to detect characteristics such as gender, personality or mood (Michalak et al., 2009; Pollick et al., 2001; Satchell et al., 2017). As such, a better understanding of body motion, and the analysis and generation of motion data is important to further develop fields such as human-robot interaction and human activity recognition. For dance movement in particular, generative models have potential as artistic tools for animation and choreography.

Research in embodied music cognition has identified several audio features that are relevant to how we move to music. Burger et al.'s (2013) work suggests that several mappings exist between different aspects of music and music-induced movement. The presence of a clear beat, for example, was shown to translate to faster movements of head and hands.

The work presented here is part of an ongoing research effort to examine how deep learning can be used to capture salient features of human movement, and especially dance movement, using full-body motion capture data and sound. As part of this work, we have collected a dataset of motion capture recordings of dance improvisation performed to six different musical stimuli. The improvisations are performed by experienced dancers and use contemporary music styles.

Here, we present the results of training a generative mixture density recurrent neural network (MDRNN) on our motion data and audio features which have been shown to affect certain aspects of movement to music. Without the inclusion of audio features, the MDRNN is able to generate sequences of movement which are (subjectively) realistic variations of the underlying training data. The results presented here indicate that the model retains this ability to produce movement variations when audio features are added. Our findings suggest that the model additionally learns that different audio features affect the way the body moves.

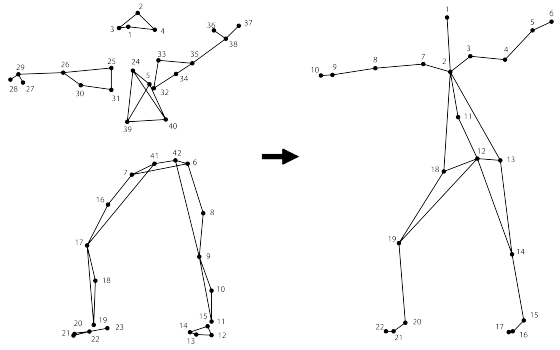


Figure 1: The 43 reflective markers translated to 22 points

Motion Capture Data

Our dataset contains 54 one minute motion capture recordings of improvised dance performed by three experienced dancers. Each dancer performs three one minute improvisations to six different musical stimuli. The dataset was recorded using a Qualisys optical motion capture system with 12 Oqus 300/400 series cameras which capture 43 reflective markers worn by the dancers. Figure 1 shows how the 43 marker positions were reduced to a 22 point skeleton representation using the MoCap Toolbox 1.5 (Burger and Toiviainen, 2013). Small gaps in the data were spline-filled using Qualisys Track Manager 2019.3 and a 2nd degree Butterworth filter with a .03Hz cutoff was applied to remove any marker jitter.

Recordings in our dataset have been normalized so that the root marker (a weighted average of markers 41, 42, 6 and 7 in Figure 1) is centred at the origin. Body segment lengths are averaged across the three dancers ensuring that the data is invariant to global position and individual body dimen-

Proceedings of the 11th International Conference on Computational Creativity (ICCC'20)
ISBN: 978-989-54160-2-8

sions. The data was captured at 240Hz and downsampled to 30Hz before model training to reduce the size of each example. The resulting 54 data tensors consist of 1800 frames (60 seconds at 30Hz) with 3-dimensional positions for each of the 22 points.

Two full motion capture recordings were withheld for testing while the remaining 52 examples were split into two sets, 80% were used for training and 10% for validation. Each example has been sliced into overlapping sequences of 300 frames and the spatial dimensions of each of the 22 points are scaled using min-max normalisation. The input to our model thereby consists of 78000 overlapping sequences of 300 frames with their corresponding audio features. The model performs a sequence-to-sequence mapping between the training examples (including audio features) and the shifted sequence of motion capture frames (excluding audio features – the model only predicts motion).

Representing sound

There are several aspects to consider when selecting appropriate audio features to represent the music examples to which the dancers were improvising. Several previous works (Fukayama and Goto, 2015; Lee et al., 2019; Seo et al., 2013) have largely focused on the rhythmical, beat-matching aspects when generating dance.

Although the presence of a clear beat can affect our urge to move, moving in sync with a beat is only one of several ways musical features influence the way we move. For the experiments presented here, we have chosen to use two high-level rhythm- and timbre-related features: pulse clarity and sub-band spectral flux. Previous work by Burger et al. (2013) has shown connections between pulse clarity (Lartillot et al., 2008) and overall body movement, as well as sub-band spectral flux and movement of the head and hands.

Pulse clarity is a high-level feature which measures how clearly the underlying pulse of the music is perceived. Pulse clarity is estimated using the overall entropy of the energy distribution of the frequency spectrum within a musical piece. We calculate a series of pulse clarity values for each musical stimuli using a sliding window of 5 seconds and a hop size of 0.08 seconds. This gives us a time series wherein each value corresponds to a single frame of the motion capture data.

Spectral sub-band flux measures spectral changes in different frequency bands of an audio signal. Alluri and Toiviainen (2010) found that the sub-band fluctuations in the region between 50 Hz and 200 Hz are related to the perceived "fullness" of a musical piece, while fluctuations in the region of 1600 Hz and 6400 Hz were linked to the perceived "activity" of the piece. These sub-bands also correspond to activity from rhythmic instruments such as kick drum and bass guitar for the lower frequency band and hi-hat and cymbals for the higher range.

We extract two frequency bands, one low-frequency band (50 Hz - 100 Hz) and one high-frequency band (3200 Hz - 6400 Hz) from the six musical stimuli. The spectral flux is then calculated using the same window and hop size as for the pulse clarity values resulting in a single sub-band flux value for each of the two bands for every frame of motion capture data. The audio features are appended to each data frame of the motion capture data to create the tensors used to train the generative mixture density recurrent neural network.

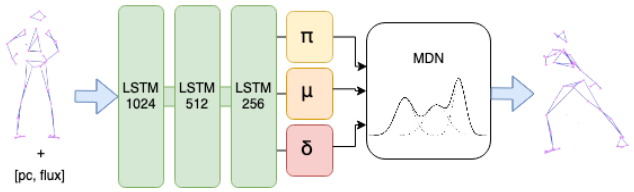
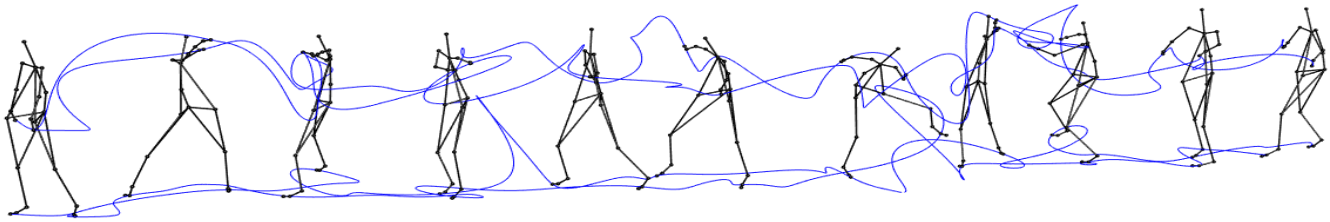


Figure 2: Sampling from the MDRNN. One frame of mocap data and audio features is sent through the model. The MDRNN outputs the parameters of a mixture distribution which is sampled to generate the next frame.

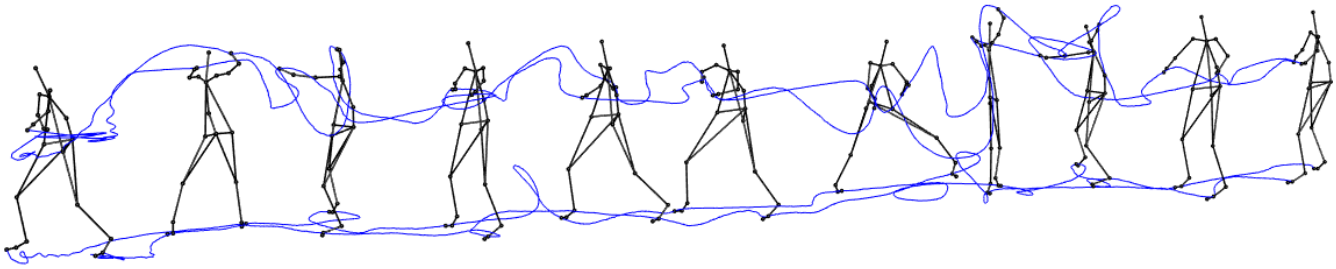
Mixture Density Recurrent Neural Networks

Mixture density networks (MDNs) (Bishop, 1994) treat the outputs of a neural network as the parameters of a Gaussian mixture model (GMM), which can be sampled to generate real-valued predictions. A GMM can be derived using the mean, weight and standard deviation of each component. The number of components needed to accurately represent the data is not known and is treated as a hyperparameter for our model. For the study outlined here, we have used 4 components. By combining a recurrent neural network (RNN) with an MDN to form an MDRNN we can make real-valued predictions based on a sequence of inputs. Figure 2 shows the model architecture of the MDRNN used in this work. The RNN consists of three layers of LSTM cells (Hochreiter and Schmidhuber, 1997). The three LSTM layers contain 1024, 512 and 256 hidden units respectively. The outputs of the third LSTM layer are in turn connected to an MDN. The LSTM layers learn to estimate the mean (μ), standard deviation (σ) and weight (π) of the 4 Gaussian distributions of the MDN. This approach has the advantage of control over the diversity and "randomness" of sampling, and control over the number of mixture components that allow training to account for situations where multiple predictions could be considered equally suitable. MDRNNs have previously been applied to various other tasks such as sketches (Ha and Eck, 2017), handwriting (Graves, 2013), and music control generation (Martin and Torresen, 2019).

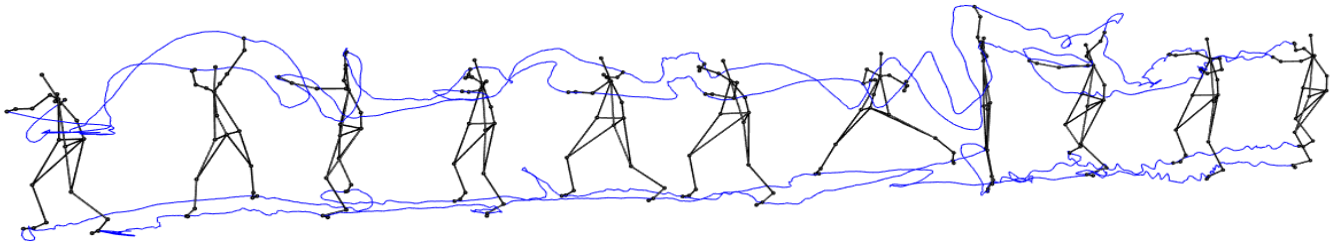
To optimize an MDN, we minimize the negative log-likelihood of sampling true values from the predicted GMM for each example. A probability density function is used to obtain this likelihood value. This configuration corresponds to 8.5M parameters. The loss function in our system is calculated by the `keras-mdn-layer` Martin (2018) Python package which makes use of Tensorflow's probability distributions package to construct the PDF. The model is trained using the Adam optimizer (Kingma and Ba, 2014) until the loss on the validation set failed to improve for 10 consecutive epochs.



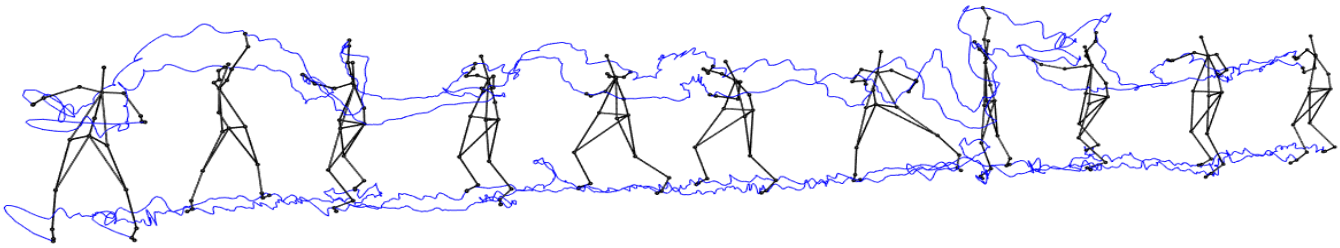
(a) The original movement sequence used to prime the model.



(b) Movement generated using the priming sequence with corresponding audio features. Movements are less smooth and expressive than the original recording, but the generated movement follows the priming example nicely.



(c) Sequence generated when the audio features from the priming example are replaced with features from a song not used in the dataset. The movements are more unstable and shaky.



(d) Here, audio features are replaced with features calculated from a white noise signal. While the overall sequence is similar to the priming example, the movements contain more variation between frames, causing jittery movement.

Figure 3: These figures show the trajectories of hand and toe markers over time (left to right). When generating motion using audio which was not used in training (3c) or white noise (3d) the movements become more unstable.

Altering Movement Using Audio Features

When the MDRNN is trained on movement without the addition of audio features it is able to generate movements which are, under visual inspection, realistic. In this section, we examine the effect of altering the audio input for a model trained on both movement and audio data.

To examine to what extent the model has learned a correlation between the audio features and the movement we generate motion using a priming technique. When using priming the input to the model is taken from one of the examples which were withheld during training. At each time step, the input consists of the 3D positions of the 22 points and the corresponding set of audio features for that time step. The model then predicts the positions of the 22 points at the next time step. Thereby, the model always predicts the next pose using the values from the priming sequence. By altering the audio features of the priming example the output can be evaluated to determine the effect which different audio features have on the generated output. We examine three such cases here. First, we investigate a sequence generated using the audio features associated with the priming example itself, that is, features calculated from the musical stimuli the

dancer was improvising to when the priming example was recorded. Secondly, we replace this audio with an excerpt from a song which was not part of the training data. Finally, features calculated from a white noise signal is used to replace the original audio features. Figure 3 show keyframes from the 3 generated movement sequences as well as the motion sequence used to prime the model.

Discussion

When generating movement using the audio features belonging to the priming sequence (Figure 3b), the model generates movement which largely follows the example (Figure 3a). While the overall movement sequence is similar, some expressiveness seems to have been lost. The generated motion could be said to resemble a "dead pan" performance of the original sequence, as trajectories of arms and legs are to some extent muted in comparison to the original. This may be due to the model generalising movement across the training data.

Figure 3c shows the sequence generated when the original audio features are replaced with features calculated from a song not included in the training data. The model produces a movement sequence which matches the priming sequence well, indicating that the model is able to predict the next frame of the motion data even with unseen audio features. Still, additional noise is visible in this example (when compared to 3b), suggesting that the model has not fully learned to generate smooth movements when unseen audio features are used.

In the final figure, 3d, the audio features are replaced with features calculated from a white noise signal. Here, trajectories are decidedly affected by the audio. As with figure 3c and 3b the model still predicts reasonable positions for the 22 markers at every frame, but with a larger variation between frames, causing the resulting sequence to display jittery movement. This indicates that the model does rely on structured audio to generate realistic movements at the micro (if not macro) scale.

Conclusions and Future Work

These results indicate that the MDRNN model could be used to explore how music and audio features affect the way the dancers move, and how this manifests itself in the movements generated by a deep neural network. Much work remains to obtain a comprehensive understanding of how MDRNNs can model cross-modal interactions like those between sound and motion. An important aspect is how we can best evaluate the performance of this model. Finding good qualitative and quantitative ways to evaluate creative data generated by models such as this one will be a central question in our future work. Going forward, we will focus on systematically exploring metrics to evaluate the generated movements, train the model on a larger dataset and experiment with alternate audio representations.

References

Alluri, V., and Toiviainen, P. 2010. Exploring perceptual and acoustical correlates of polyphonic timbre. *Music Perception: An Interdisciplinary Journal* 27(3):223–242.

Bishop, C. M. 1994. Mixture density networks. Technical report, Aston University, Birmingham, UK.

Burger, B., and Toiviainen, P. 2013. Mocap toolbox-a matlab toolbox for computational analysis of movement data. In *10th Sound and Music Computing Conference, SMC 2013, Stockholm, Sweden*. Logos Verlag Berlin.

Burger, B.; Thompson, M.; Luck, G.; Saarikallio, S.; and Toiviainen, P. 2013. Influences of rhythm- and timbre-related musical features on characteristics of music-induced movement. *Frontiers in Psychology* 4:183.

Fukayama, S., and Goto, M. 2015. Music content driven automated choreography with beat-wise motion connectivity constraints. *Proceedings of SMC* 177–183.

Graves, A. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Ha, D., and Eck, D. 2017. A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477*.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9:1735–80.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lartillot, O.; Eerola, T.; Toiviainen, P.; and Fornari, J. 2008. Multi-feature modeling of pulse clarity: Design, validation and optimization. In *ISMIR*, 521–526. Citeseer.

Lee, H.-Y.; Yang, X.; Liu, M.-Y.; Wang, T.-C.; Lu, Y.-D.; Yang, M.-H.; and Kautz, J. 2019. Dancing to music. In *Advances in Neural Information Processing Systems*, 3581–3591.

Martin, C. P., and Torresen, J. 2019. An interactive musical prediction system with mixture density recurrent neural networks. In *Proc. NIME '19*, 260–265.

Martin, C. 2018. keras-mdn-layer: Python Package.

Michalak, J.; Troje, N. F.; Fischer, J.; Vollmar, P.; Heidenreich, T.; and Schulte, D. 2009. Embodiment of sadness and depression—gait patterns associated with dysphoric mood. *Psychosomatic medicine* 71(5):580–587.

Pollick, F. E.; Paterson, H. M.; Bruderlin, A.; and Sanford, A. J. 2001. Perceiving affect from arm movement. *Cognition* 82(2):B51–B61.

Satchell, L.; Morris, P.; Mills, C.; O'Reilly, L.; Marshman, P.; and Akehurst, L. 2017. Evidence of big five and aggressive personalities in gait biomechanics. *Journal of nonverbal behavior* 41(1):35–44.

Seo, J.-H.; Yang, J.-Y.; Kim, J.; and Kwon, D.-S. 2013. Autonomous humanoid robot dance generation system based on real-time music input. In *RO-MAN, 2013 IEEE*, 204–209. IEEE.