Tracing from Sound to Movement with Mixture Density Recurrent Neural Networks

Benedikte Wallace
University of Oslo, Norway
RITMO, Department of Informatics
benediwa@ifi.uio.no

Charles P. Martin*
University of Oslo, Norway
RITMO, Department of Informatics
charlepm@ifi.uio.no

Kristian Nymoen
University of Oslo, Norway
RITMO, Informatics & Musicology
kristian.nymoen@imv.uio.no

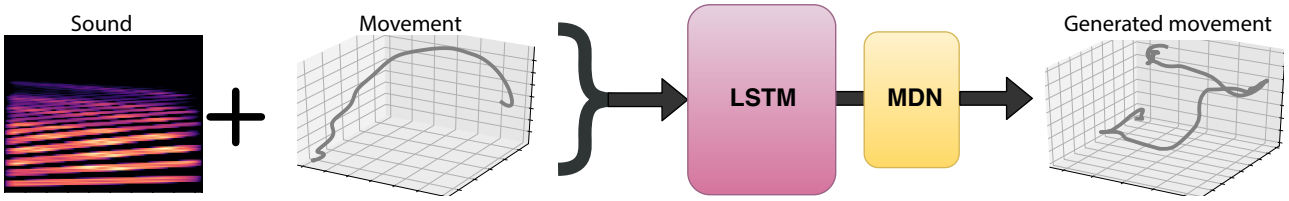


Figure 1: Generating sound-tracings with mixture density recurrent neural network (MDRNN). Sound features and previous movements are inputs to the network and the outputs are predicted movements.

ABSTRACT

In this work, we present a method for generating *sound-tracings* using a mixture density recurrent neural network (MDRNN). A sound-tracing is a rendering of perceptual qualities of short sound objects through body motion. The model is trained on a dataset of single point sound-tracings with multimodal input data and learns to generate novel tracings. We use a second neural network classifier to show that the input sound can be identified from generated tracings. This is part of an ongoing research effort to examine the complex correlations between sound and movement and the possibility of modelling these relationships using deep learning.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; • **Applied computing** → **Sound and music computing**; • **Information systems** → *Multimedia and multimodal retrieval.*

KEYWORDS

Sound-Tracing, Mixture-Density Networks, Recurrent Neural Networks, Music Information Retrieval

ACM Reference Format:
Benedikte Wallace, Charles P. Martin, and Kristian Nymoen. 2019. Tracing from Sound to Movement with Mixture Density Recurrent Neural Networks. In *Proceedings of 6th International Conference on Movement and Computing (MOCO'19).* MOCO, Tempe, AZ, USA, Article 31, 4 pages. https://doi.org/10.1145/3347122.3371376

1 INTRODUCTION

Our perception of sound is intrinsically entangled to our perception of movement [6]. Hearing a sound involves not only bottom-up processing of sonic features but also a top-down understanding of what sort of objects and sound-producing actions were involved in creating the sound. Inputs to different sensory modalities have common references in metaphoric language (e.g., bright, soft, high, energetic), and such cross-modal metaphors have been studied extensively in order to understand human perception. One such approach is to study gestural renderings of short sounds, what Godøy et al. [7] refer to as *sound-tracing*, noting that people connect sonic shapes to gestural shapes.

The research presented in this paper is motivated by studies of sound-tracing. We ask to what extent an artificial neural network is able to mimic humans in gestural rendering of sounds. This task is particularly challenging for several reasons. First, because the neural network needs to model both human movement constraints and auditory perception, as well as the bridge between the two modalities, and secondly, because humans do not seem to use any single strategy for coupling auditory features to movement features. Rather, a variety of strategies for mapping between sound and movement have been identified [3, 11, 17]. To account for a range of equally valid strategies for rendering gesture from sound, we employ a mixture density network (MDN) architecture whereby the outputs of an artificial neural network are interpreted to be the parameters of a Gaussian mixture model (GMM). A GMM can potentially model these multiple strategies simultaneously and be sampled to provide a number of valid predictions. Since both sound and movement are unfolding in time, we use a recurrent neural network (RNN) with long short-term memory (LSTM) units which make predictions based on the network's state at previous time steps, making it possible for the network to learn temporal dependencies.

The mixture density recurrent neural network (MDRNN) employed in our work learns to generate sound-tracings using multimodal training data. We further experiment with sampling from the probability distribution of the MDRNN with different degrees of randomness. By way of visual inspection of the tracings generated by this model, we find that the results produce tracings similar to those found in the training data. Further, we find that in training a

*Also with Australian National University, Canberra.

MOCO'19, October 2019, Tempe, AZ, USA

classifier to identify which of the sounds the generated tracings correspond to we achieve an accuracy of approximately 89% indicating that there are clear differences between, and commonalities within tracings conditioned on each sound. In the following section, we provide a brief overview of relevant background. In Section 3 we present the dataset and method applied in our project. In Section 4 we present and discuss the results for the generated sound-tracings before concluding and looking ahead in Section 5.

2 BACKGROUND

2.1 Sound-tracing

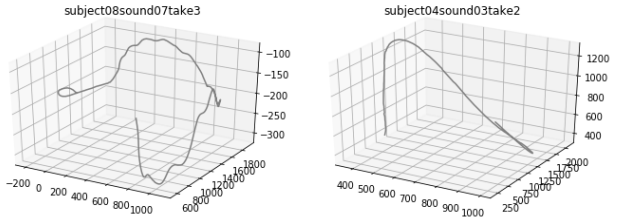
Initial exploratory studies of tracing sounds were performed by Godøy et al. in 2006, with participants tracing on a 2D tablet [7]. Küssner and Leech-Wilkinson [12] later documented that musical training had an effect on the choice of strategy used for mapping between sound and shape in tablet drawings. In following years several studies have been performed using motion capture technology in order to record tracings in three dimensions [3, 11, 16, 17]. Although participants move freely, most participants apply a limited selection of strategies to solve the task. Taking the tracing of musical pitch as an example, Kelkar and Jensenius identified a total of 6 different strategies for tracings of melodic contour [11]; a broader picture than several previous studies that focused on the high correlation between pitch/brightness and vertical movement [5, 16, 18]. Others have also shown a relation between pitch and hand distance, strategies with both strongly positive and strongly negative correlation between the two were found [11, 17].

2.2 Sequence prediction

The challenge addressed in our paper is a sequence prediction problem—predicting future time steps based on previous and current events. Previous research on movement synthesis have applied Restricted Boltzmann Machines [1], Convolutional Autoencoders [10], and recurrent neural networks (RNN) [4]. We have chosen an RNN architecture, which is a suitable choice for sequence prediction tasks which require that the temporal aspects of the data are represented. The recurrent connections between neurons in such networks allow the model to retain information between time steps in its internal state. Thereby, the output of the RNN at each time step is affected by the information learned at previous time steps. This makes RNNs a suitable choice for many sequence prediction tasks such as drawings [9], chord progressions [13], text [19], and movement.

Mixture density networks (MDNs) [2] treat the outputs of a neural network as the parameters of a Gaussian mixture model, which can be sampled to generate real-valued predictions. This approach has the advantage of both generating real-valued predictions, as well as 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. An RNN can be combined with an MDN to form an MDRNN that can make real-valued predictions based on a sequence of inputs.

To optimize an MDN, we minimize the negative log-likelihood of sampling true values from the predicted GMM for each example.



(a) Tracing of sound 7 **(b) Tracing of sound 4**

Figure 2: Sound-tracing examples from the original sound-tracing experiment [16] which constitutes our training data.

Table 1: Dataset description

Data type	No. Examples	Variables	Representation
Sound files	10	Pitch, Timbre, Loudness	128 bin Mel-spectrograms. Frame rate 100 Hz
Motion capture	410, each paired with one sound file	X, Y, Z position	Differentiated X, Y, Z data. Frame rate 100 Hz

The probability density function (PDF) is used to obtain this likelihood value. In our case, the GMM consists of K n-variate Gaussian distributions. For simplicity in the PDF, these distributions are restricted to having a diagonal covariance matrix, and thus the PDF has the form:

$$p(\boldsymbol{\theta}; \boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k; \boldsymbol{x}) \tag{1}$$

Where π are the mixing coefficient, μ, the Gaussian distribution centres and Σ the covariance matrices. The loss function in our system is calculated by the `keras-mdn-layer` [14] Python package which makes use of Tensorflow's probability distributions package to construct the PDF.

MDRNNs are becoming well-established tools in the generation of creative data. They have previously been applied to musical sketches in two dimensions as part of a smartphone app [15], to sketches [9], and handwriting [8]. While an MDRNN has previously been applied to motion capture data [4], until now, it has not been employed to model sound-to-gesture relationships.

3 METHODS AND MATERIALS

3.1 Dataset

The dataset used to construct training data for the MDRNN is summarized in Table 1. It consists of 10 sound files and 410 motion capture recordings of 15 subjects moving a rod through the air in order to trace each sound [16]. The motion capture recordings were time-synchronised to the sound playback. A NaturalPoint Optitrack motion capture system with 7 Flex V100 cameras was used to track the 3D position of the tip of the rod at a frame rate of 100 Hz. Figure 2 displays examples of sound-tracings recorded for sound 4 and 7.

3.2 Pre-processing

The motion recordings contained 50 data frames prior to the sound file playback. The recordings were trimmed to only contain those data frames where sound was played back. To avoid starting position bias we used the first finite difference between motion capture frames as training data.

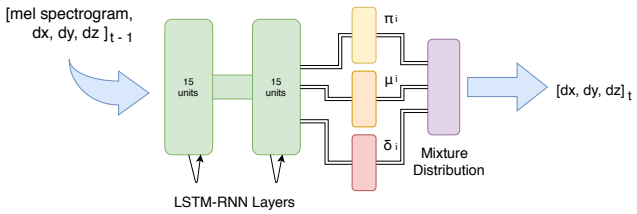


Figure 3: The MDRNN structure, two layers of LSTM units are followed by the MDN layer. The resulting GMM can be sampled to produce the next movement.

The sound files, synthesized in Max/MSP, range in duration from 2 to 4 seconds, and vary in pitch, timbre and loudness. For the purpose of this experiment, mel-scaled spectrograms with 128 Mel bands were extracted from each sound. The sound files have a sample rate of 44.1 kHz, by selecting a hop length of 441 samples the time scale of the spectrograms correspond to the 100 fps motion capture recordings.

3.3 Experiment design

Two versions of the MDRNN were produced to experiment with predicting the movement features present in the dataset. The first version took only movement as input; while the second took both movement and sound features. More precisely, for each time step t, the movement-only network attempts to map $(dx_t, dy_t, dz_t) \rightarrow (dx_{t+1}, dy_{t+1}, dz_{t+1})$. The sound-and-movement network maps $(m_1, \ldots, m_{128}, dx_t, dy_t, dz_t) \rightarrow (dx_{t+1}, dy_{t+1}, dz_{t+1})$ where the m_i are 128 mel-scaled frequency weights obtained through a STFT procedure. The aims of our experiments were:

(1) To determine whether convincing movement traces can be obtained using the movement-only network and our dataset.
(2) To determine whether the addition of sound features allow movements to be controlled by conditioning prediction on a certain sound.

For validation of the sound-and-movement network, an automated classification of generated traces was performed by training a neural network that attempts to identify which sound was used to generate a trace. This classifier can tell us whether different sound inputs lead to recognisable traces. The network comprises three layers of 30 LSTM units and a dense layer with softmax activation (19030 parameters) and operates on 100-step long sequences of movement data. Two versions of the classifier were trained, one on the original sound-tracing dataset and one on a set of 1000 generated tracings—100 for each of the 10 sounds.

3.4 MDRNN and Training

The generative RNN used in this work consists of two layers of LSTM cells each with 15 hidden units. The outputs of the second LSTM layer are in turn connected to a Mixture-Density network. The model learns to estimate the mean, standard deviation and weight of 5 normal distributions. Figure 3 shows an overview of the MDRNN architecture. This configuration corresponds to 3,560 parameters for the movement-only model, and 11,800 parameters for the sound-and-movement model.

Both models were trained using the Adam optimizer and on epochs of shuffled data until the loss on a validation set (10% of the dataset) failed to improve for five consecutive epochs. The movement-only network trained for 50 epochs with a final validation loss of 2.96 and the sound-and-movement network trained for 40 epochs with a final validation loss of 4.49.

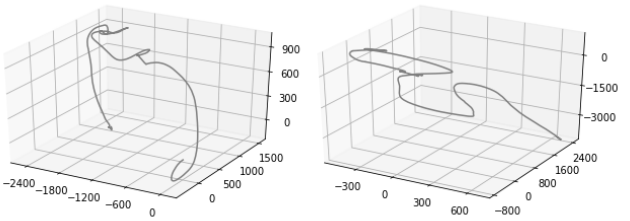


Figure 4: Examples of tracings generated by the MDRNN trained only on motion capture data with no sound features

4 RESULTS AND DISCUSSION

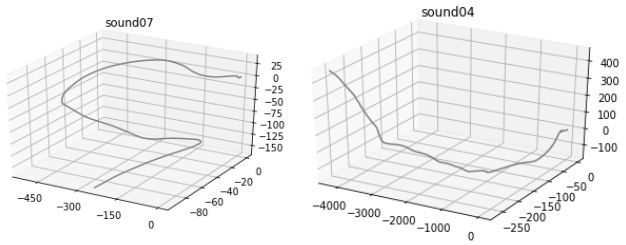
4.1 Movement-only tracings

As a starting point for this exploration, the model is trained on the movement of the rod without including any information pertaining to the sounds the subjects were moving to. During training, the motion capture recordings are segmented into overlapping sequences of 100 time steps, with each time step having a corresponding target value consisting of the position variation in the following time step. Figure 4 shows examples of tracings generated by this model. The examples show that the network generates reasonable motion sequences with some variability.

4.2 Sound-and-Movement Tracings

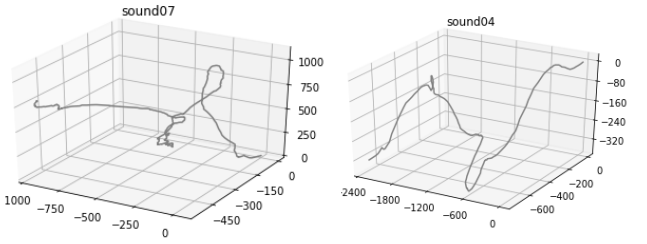
To generate movement using sound and motion features in combination, the 128 Mel bands are appended to the three position variables associated with the current time step of each training example. Further, these motion and sound examples are segmented into overlapping sequences of 100 time steps, equivalent to 1 second. The target values used in training the network are the same as when generating movement from only motion data, the 3 position values for the next time step. Thus, the model learns to predict the upcoming position of the rod given only the current sound features and its preceding position. When sampling a prediction from the output of our MDN layer we can choose to increase the width and/or height of the Gaussian distributions, the mixture components of our model. By scaling σ, the standard deviations, or π, the mixing weights, we affect the probability of sampling from certain regions of the probability distribution for our three position variables. These values affect the smoothness of generated tracings. Figures 5 and 6 show tracings generated when using sound 7 and sound 4 to condition the predictions and sampling with low and high randomness respectively. The sounds were chosen arbitrarily to give a comparison between original and generated tracings and the effect of changes in sampling temperature.

The tracings generated by our model are remarkably similar within a group of tracings to the same sound input. There are also quite clear differences between groups that were generated to different sounds. To assess the within-group similarities and between-group differences we perform an automated classification of the generated sound-tracings using an LSTM classifier. An



(a) Tracing of sound 7 **(b) Tracing of sound 4**

Figure 5: Tracings generated by the MDRNN sampled at low temperature $\sigma = 0.0, \pi = 1.0$



(a) Tracing of sound 7 **(b) Tracing of sound 4**

Figure 6: Tracings generated by the MDRNN sampled at $\sigma = 0.5, \pi = 2.5$

accuracy of 89% was obtained on a hold out set of 250 traces. In comparison, when training the classifier on the original dataset, an accuracy of 75% was obtained on a hold-out set of 62 traces, similar to the previously reported classification accuracy for this dataset using a Support Vector Machine classifier [16]. This indicates that there is more variability in the original dataset than in the set of generated tracings.

Although the classifier is able to distinguish between the sound-tracings, the perceptual connection between sound-tracing to the sonic input is vaguer. When attempting to classify the generated tracings using the classifier trained on the original sound-tracing data we achieve an accuracy of only 17%. While it is easy to make a connection between the tracing and the sound in the original dataset, the perceptual link between generated tracing and sound is less clear. The lack of such a link may come down to a number of things, such as the size of dataset or the data representation, and indicates that there is still a way to go before our model mimics human cognition of sound and movement.

5 CONCLUSIONS AND FUTURE WORK

This paper presents a novel system for generating sound-tracings using multi-modal input consisting of sound and movement. The proposed method involves training an MDRNN on motion capture data and sound from a sound-tracing experiment. The trained model generates new instances of sound-tracings using the sound features of mel-scaled spectrograms. Previous work has shown that humans can express movements related to sounds in a reliable way. This work contributes new evidence that similar couplings can be learned and generated by an ML system. In future work, we will examine additional metrics for evaluating the tracings generated by the model. We wish to examine more closely the similarities between tracings relating to the same sound, and also what sound-tracing features are shared between the original data and the generated tracings. Further, we intend to perform additional experiments on generating movement from sound using more complex data such as choreography using full-body motion capture and music.

ACKNOWLEDGMENTS

This work was partially supported by the Research Council of Norway through its Centres of Excellence scheme, project number 262762.

REFERENCES

[1] Omid Alemi, Jules Françoise, and Philippe Pasquier. 2017. GrooveNet: Real-time music-driven dance movement generation using artificial neural networks. *networks* 8, 17 (2017), 26.
[2] Christopher M. Bishop. 1994. *Mixture density networks.* Technical Report. Aston University, Birmingham, UK. http://publications.aston.ac.uk/373/
[3] Baptiste Caramiaux, Frédéric Bevilacqua, and Norbert Schnell. 2010. Towards a Gesture-Sound Cross-Modal Analysis. In *Gesture in Embodied Communication and Human-Computer Interaction*, Stefan Kopp and Ipke Wachsmuth (Eds.). Springer, Berlin Heidelberg, 158–170.
[4] Luka Crnkovic-Friis and Louise Crnkovic-Friis. 2016. Generative choreography using deep learning. In *Proceedings of the Seventh International Conference on Computational Creativity*. 272–277.
[5] Zohar Eitan and Roni Y. Granot. 2006. How Music Moves: Musical Parameters and Listeners' Images of Motion. *Music Perception* 23, 3 (2006), pp. 221–248.
[6] Rolf Inge Godøy. 2018. Sonic Object Cognition. In *Springer Handbook of Systematic Musicology*, Rolf Bader (Ed.). Springer, Berlin, Heidelberg, 761–777.
[7] Rolf Inge Godøy, Egil Haga, and Alexander Refsum Jensenius. 2006. Exploring music-related gestures by sound-tracing: A preliminary study. In *Proceedings of the COST287-ConGAS 2nd International Symposium on Gesture Interfaces for Multimedia Systems (GIMS2006)*. 27–33.
[8] Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv eprints arXiv:1308.0850* (2013).
[9] David Ha and Douglas Eck. 2017. A neural representation of sketch drawings. *arXiv eprints arXiv:1704.03477* (2017).
[10] Daniel Holden, Jun Saito, Taku Komura, and Thomas Joyce. 2015. Learning motion manifolds with convolutional autoencoders. In *SIGGRAPH Asia 2015 Technical Briefs*. ACM, 18.
[11] Tejaswinee Kelkar and Alexander Refsum Jensenius. 2018. Analyzing Free-Hand Sound-Tracings of Melodic Phrases. *Applied Sciences* 8, 1 (2018).
[12] Mats B. Küssner and Daniel Leech-Wilkinson. 2014. Investigating the influence of musical training on cross-modal correspondences and sensorimotor skills in a real-time drawing paradigm. *Psychology of Music* 42, 3 (2014), 448–469.
[13] Hyungui Lim, Seungyeon Rhyu, and Kyogu Lee. 2017. Chord Generation from Symbolic Melody Using BLSTM Networks. In *Proceedings of the 18th International Society for Music Information Retrieval Conference*. 621–627.
[14] Charles Martin. 2018. keras-mdn-layer: Python Package. https://doi.org/10.5281/zenodo.1482348
[15] Charles Patrick Martin and Jim Torresen. 2018. RoboJam: A musical mixture density network for collaborative touchscreen interaction. In *Int'l. Conference on Computational Intelligence in Music, Sound, Art and Design*. Springer, 161–176.
[16] Kristian Nymoen, Kyrre Glette, Ståle Skogstad, Jim Torresen, and Alexander R Jensenius. 2010. Searching for Cross-Individual Relationships between Sound and Movement Features using an SVM Classifier. In *Proceedings of the International Conference on New Interfaces for Musical Expression*. 259–262.
[17] Kristian Nymoen, Rolf Inge Godøy, Alexander Refsum Jensenius, and Jim Torresen. 2013. Analyzing correspondence between sound objects and body motion. *ACM Transactions on Applied Perception* 10, 2 (2013), 9:1–9:22.
[18] Elena Rusconi, Bonnie Kwan, Bruno L. Giordano, Carlo Umiltà, and Brian Butterworth. 2006. Spatial representation of pitch height: the SMARC effect. *Cognition* 99, 2 (2006), 113 – 129.
[19] Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 1017–1024.