Brahms, Bodies and Backpropagation

Artificial Neural Networks for Movement Classification in Musical Performance

Vanessa Yaremchuk
Input Devices and Music Interaction Laboratory
Centre for Interdisciplinary Research in Music
Media and Technology
McGill University
Schulich School of Music
Montreal, Canada
vanessa.yaremchuk@mail.mcgill.ca

Marcelo M. Wanderley
Input Devices and Music Interaction Laboratory
Centre for Interdisciplinary Research in Music
Media and Technology
McGill University
Schulich School of Music
Montreal, Canada
marcelo.wanderley@mcgill.ca

ABSTRACT

Two types of artificial neural networks are used to determine a sufficient subset of data for reasonable classification of musical instrument based on performance data from motion capture. Feedforward and recurrent networks are trained on subsets of joint angles and centre of mass from performances by violists and clarinettists. A successfully learned mapping from the reduced set of input data to the correct instrument classification implies that the data subset carries sufficient information to identify an instrument. After training, cross-validation is performed by testing networks on previously unseen data. Finally, performance is compared with that of humans performing similar classification tasks based on the same data.

Feedforward and recurrent networks are found to perform similarly when classifying test data. Instrument recognition rates by networks are comparable with human recognition rates over the various data subset conditions. The methods demonstrated here could also be applied to other movement analysis domains (e.g. gait, posture, kinematics, clinical/rehabilitation work).

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning—*Connectionism and neural nets*

General Terms

Experimentation, Performance

Keywords

artificial neural network, motion capture, feedforward, recurrent, music, performance, movement analysis

1. INTRODUCTION

The study of performer movements can be used toward understanding movement consistency within and across performers and musical instruments. Movement analysis uses objective data to measure, evaluate and understand performer movement. One approach to acquiring quantitative data is to use motion capture methods which record spatial coordinates of markers positioned on the musicians' bodies, along with corresponding temporal information, video, and audio. Information about the position of markers is used to calculate three dimensional positions and angles of limbs and joints.

While motion capture provides accurate data about a subject's movements, it is necessary to find a method to mitigate the overwhelming amount of data in order to study it. There are many different approaches for dealing with this issue. Examples include the use of functional data analysis[7], and sonification of movement data[6]. In addition to these methods of reducing the data to be further analysed, it is useful to explore what important information is in the movements. One possible solution is to use machine learning, and, in particular, artificial neural networks (ANNs).

The ancillary and expressive movements (i.e., those that do not directly influence the generation of sound) of performers convey information to the observer or audience. Ancillary movements have been studied across a variety of instruments including stationary instruments such as piano[4, 3], or marimba[2] as well as mobile instruments such as the bassoon, saxophone[2], clarinet[8], and violin[3]. Across these instruments, ancillary gesture was found to convey information about the performer and their expressive intent. This paper considers whether there are consistencies within groups of musicians performing with the same type of instrument.

When considering human perception of instrument identification, it can be useful to study the amount of information present, from a machine learning perspective, in the data. That is, if one considers musicians during performance, there is some minimal amount of information about movement required to make a correct instrument classification. Whether or not this set of information is also sufficient for human perception of the instrument is a further question, but one whose boundaries can be informed by the machine learning result. Similarly, it is useful to consider what is a sufficient subset of data for reasonable classification perfor-

MOCO'14, June 16-17 2014, Paris, France.

ACM 978-1-4503-2814-2/14/06 ...$15.00.
http://dx.doi.org/10.1145/2617995.2618011 .

mance. Training ANNs on different subsets of movement data is one way of exploring this topic. If a network can learn a function that maps from the reduced set of data as input to the correct classification of the instrument, then the data set represented by the input data is sufficient.

2. GENERAL METHOD

2.1 Training Data

Current and previous work at the Input Devices and Music Interaction Laboratory (IDMIL) includes the creation of databases of motion capture data from performances of various instruments by student and professional musicians recorded at the IDMIL. These data include spatial coordinates of the markers positioned on the musicians' bodies, along with corresponding temporal information, video, and audio.

ANNs were trained on subsets of data from motion capture sessions of violists and clarinettists playing excerpts of J. Brahms' Sonata Op. 120 No. 1. The piece is for piano and clarinet or viola, so the same excerpts were performed by both the violists and the clarinettists. The data used in training came from several motion capture sessions, each with a different performer. A subset of 50 frames from each session was used, corresponding to approximately 0.5 seconds of gesture data from each performer.

The motion capture system calculates positions of markers as well as angles at joints, so with the exception of the centre of mass networks, angles were used in order to eliminate the possibility of unintentionally training the ANN to distinguish between performers based on height. This was only a concern because the number of performers is fairly small; if hundreds of performers were being used the chances of ANNs of these size memorising heights is vanishingly small. It would also have been effective to use positional data and normalise them across all the performers, but as the angle data are already calculated by the motion capture system those additional preprocessing steps were avoided. The motion capture system also calculates a 2-dimensional projection of the centre of mass, which eliminates the axis corresponding to height, so this was used in the corresponding training sets instead of normalising the centre of mass data for performer height.

In all of the cases, the input was paired with a desired output of -1 if the data came from a session where a clarinet was played, and with a desired output of +1 if the data came from a motion capture session with a violist.

2.2 Network Architecture

Two types of network were used: a feedforward network and a recurrent network. The motion capture data is time-sequential, so this allowed investigation of possible improvements gained by a method that takes sequence into account. In a feedforward network, information only flows in one direction. It flows from the input layer to each consecutive hidden layer to the output layer. In a recurrent network, there are some connections that send information back to a previous layer. This allows the current input to affect the internal state of the network when the next input is presented. This means that order matters. The recurrent network has some sense of sequence; the feedforward network does not. Changing the order in which inputs are presented does not change the feedforward network's response to any individual input, but it can change the response given by a recurrent network. (This can be seen as similar to the difference between finite and infinite impulse response filters.)

Both feedforward networks (multilayer perceptrons) and recurrent networks (specifically layer recurrent networks) were implemented using MATLAB's Neural Network Toolbox. Both the multilayer perceptrons and the layer recurrent networks have one hidden layer, use tan sigmoid activation functions, and use mean square error for the network performance function (which weight adjustment during training is done to minimise). The feedforward network uses Levenberg-Marquardt for the backpropagation training function, and the recurrent network uses Bayesian regularization. The layer recurrent network has a feedback loop with a single time-step delay around each layer of the network with the exception of the last layer.

All of the ANNs had one output, and a set of inputs corresponding to a subset of motion capture data. For instance, an ANN trained to classify the instrument—given information about the performer's lower body—would have 18 inputs: one input for each of the three axes for each angle corresponding to the ankle, knee, and hip joints for each leg. During training, a value within error tolerance of -1 indicated clarinet, while a value within error tolerance of +1 indicated viola. If the output value did not fall within the error tolerance of either -1 or +1, it was considered to have indicated neither instrument.

2.3 Stopping Criteria

A decision needs to be made about when the network training phase is complete. There is not a single stopping criteria for multilayer perceptrons in general; instead there are a set of criteria each with different benefits to recommend it [5]. Consider an error surface with local and global minima, that is traversed during training. A possible convergence criteria is for the gradient, with respect to the network weights, to be within a small threshold of zero. (This is because the gradient approaches zero as it approaches a minimum in the surface.) Use of this criteria can sometimes lead to very long training periods. Another possible convergence criteria is when the change in average squared errors from the previous epoch to the current training epoch is sufficiently small. In general, this reduces the training length but increases the risk of ending training prematurely. A third option is for the convergence criteria to be based on generalisation performance, by using cross validation. In this case, the network is tested on a separate data set not included in the training set; this approach is useful in avoiding overfitting. Matlab's Neural Network Toolbox accommodates all of these options, allows them to be used in combination, and by default randomly divides the data into 3 sets: 60% for training, 20% to validate generalisation, and 20% for an independent test of the network. This work used a combination of the stopping criteria and compared cross validation results across criteria as discussed in section 3.

2.4 Network Evaluation

The networks produce an output for each frame of data from the motion capture session, but a single classification should be made for each full motion capture session. This produces two places in which an evaluation of network performance could be made: single frame response, and overall session response. The networks do not produce one response per motion capture session on their own, so it has to be decided how the collection of responses to all the frames in a session combine to indicate a classification of viola or clarinet or neither.

In deciding what constitutes a correct classification, one option is to use the same strict criteria used during network training. A drawback to this approach is that it is unlikely to deal well with previously unseen performers. There is a trade-off where overly relaxed criteria result in too strong of an effect from noise whereas too strict a set of criteria for correctness eliminates the potential for generalisation. Another option is to decide upon an acceptable degree of confidence for the particular classification task, or even to produce results based on more than one tolerance for the sake of comparison. However, this approach doesn't allow for as nuanced a comparison between networks that perform similarly well. In this work, performance is measured with respect to how large the error tolerance would need to be in order for an acceptable majority of classifications to be correct. These error tolerance ranges are produced for 50% +1 of the classifications to be correct. This has the additional benefit of producing a single performance measurement for each session.

In the cases where the necessary error tolerance ranges would overlap with a correspondingly large range for the other possible classification, the network has not performed the classification correctly. For example, if the necessary range for an acceptable majority of frame classifications from a viola performance to be considered correct is -0.5 to 2.5 then a correspondingly large range for clarinet data would be -2.5 to 0.5, and as they overlap the network classification of that particular file is not correct. In the remaining cases, the size of the ranges give a measurement of how well or how precisely the network made a classification. There are 32 test files from different motion capture sessions (2 sessions each from 16 different musicians) and an individual network produces a required error tolerance range for each file. The files used in testing are different from those used as sources for training data, so while a subset of the musicians in the test files have been used in the training sets, the specific performance sessions in the test sets are different from those used for the training sets.

3. RESULTS

3.1 Full Body Networks

Network performance is considered first in the case where the input layer is given all the angle measurements available for the performer's body from the motion capture sessions. This input set is made up of angles on the left and right sides of the body for the hips, knees, ankles, shoulders, elbows, wrists, neck, head, thorax, pelvis, feet, and spine. The networks have just one real valued output where, in training, a desired value of -1 indicates clarinet and a desired value of +1 indicates viola. The training data consisted of 50 frames from each session with 8 different performers (4 violists and 4 clarinettists).

The classification performance in these cases confirms that ANNs can distinguish viola and clarinet given just information about body movement. This result is unsurprising, as the data is very different between instruments, but it is a necessary step to ensure that the networks function as expected. The next question is which subsets of this information are enough to still perform the classification well. How much information is there in the way a performer's centre of mass moves, or in the positioning of a performer's legs? Is sequential information necessary, or is there a sufficient amount of information in data from a set of frames regardless of whether the network is able to encode a sense of their ordering?

network type	hidden units	# correct	average range
MP	4	32	0.003
MP	5	32	0.147
MP	10	32	0.132
LRN	2	32	0.0000643
LRN	3	32	0.0000772
LRN	4	32	0.011
LRN	5	32	0.007

Table 1: Networks were trained on the full set of body angles and tested on previously unseen motion capture data. This table shows the network type, the number of hidden units, the number of correct classifications out of a possible 32, and the average required error tolerance range of the network output for correct classifications. MP indicates that the network is a Multilayer Perceptron, and LRN indicates a Layer Recurrent Network. The average range is the average of the necessary error tolerance ranges given over all the motion capture sessions in the test set for the given network.

3.1.1 Performance

All of the full body networks correctly classified all 32 of the test files. That is, it was always possible to find an error tolerance range that contained the majority of the frame by frame classifications, but was small enough not to cause an overlap with the same size of tolerance around the other possible category. All of these test files were of motion capture sessions not used in building the training data. There were 2 files for each of 16 performers, with 8 performers who were previously seen by the network during training, and 8 performers who were completely new to the network.

It can be difficult to determine the appropriate number of hidden units to use, as the algorithmic details of the ideal mapping between the input and output of the network are most often not known if one has opted to use ANNs. It works well to train more than one network and compare, while also trying to keep the number of hidden units small. All of the networks in table 1 get 32 out of 32 classifications correct, and have between 2 and 10 hidden units. It can be seen that the recurrent networks have smaller error tolerance ranges in general, and that both network types have smaller tolerance ranges as the number of hidden units decreases. This might seem counter-intuitive initially, but using more hidden units can reduce generalisability in many cases [5].

Another detail that can be varied is the number of epochs for which a network is trained. Connected with this is the question of which stopping criteria are adequate. In the case of these full-body networks, training for more than 100 epochs does not change the number of correct classifications, and additional training beyond 100 epochs only slightly de-

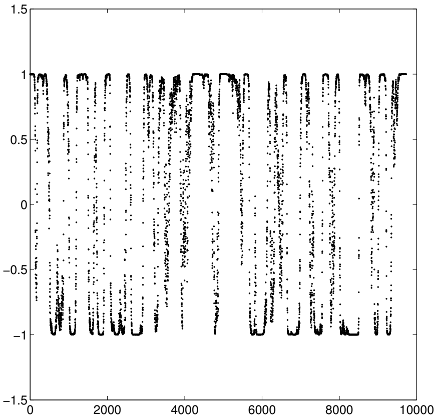

Figure 1: Output from a multilayer perceptron, with 10 hidden units, given leg angle data from a motion capture session with a previously unseen violist. The horizontal axis gives the frame number, and the vertical axis gives the network output. There is a point representing each output response for each frame from a single motion capture session. This is an example of a network not classifying a motion capture session correctly.

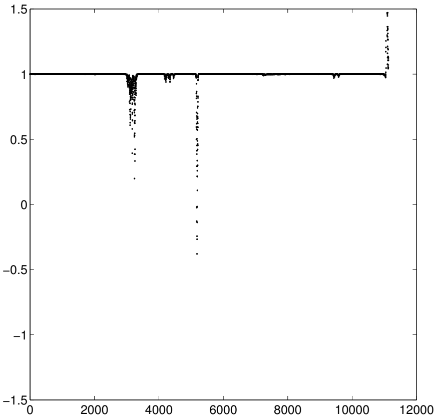

Figure 2: Output from a multilayer perceptron, with 10 hidden units, given leg angle data from a motion capture session with a previously unseen violist. The horizontal axis gives the frame number, and the vertical axis gives the network output. There is a point representing each output response for each frame from a single motion capture session. This is an example of a network classifying a motion capture session correctly.

creases the average required tolerance range. Training the networks until the stricter stopping criteria are met can take considerably more time, and does not appear to offer much benefit in this situation. It is not uncommon for network training to make a lot of progress early on then have a long stretch of slow progress until the network converges.

It is worth considering that, while only 50% + 1 of the frame by frame classifications within a full motion capture session are required to be correct in order for the session to be considered correctly classified, the ranges for error tolerance are fairly small. It is not the case that just over half of the network outputs fall somewhere above 0 and the remainder somewhere below it (or vice versa in the clarinet case). Those frame by frame classifications that are correct fall within quite narrow bands around the ideal output. While there are only two desired outputs (-1 or +1), the actual output values can be any real number, and they do not appear to have a distribution consistent with chance or noise. See figures 1 and 2 for examples of network output.

3.2 Lower Body Networks

The second set of networks were trained only on the angles of the leg joints. Again, both multilayer perceptrons and layer recurrent networks were used, and the input for both types of network was the angle data from the hips, knees and ankles of both legs. Each angle has three values (one for each axis) associated with it making a total of 18 inputs. As in the previous section, for both network types there was one real-valued output. Networks have been found to generalise better with fewer hidden units so long as there are still enough to allow the network to reach a viable stopping condition during the training phase[5], so networks were trained with small numbers of hidden units ranging from 3 to 15. This was done with both the feedforward and recurrent networks to allow for comparison between the architectures. (i.e., The number and type of connections differ, but the number of processing units in each of one type of network is also seen in the other.) Training data came from the same motion capture sessions as used in the previous section.

3.2.1 Performance

These networks do not classify all of the motion capture sessions correctly, so here it is worthwhile to split the testing set into two groups. None of the testing set is made from motion capture sessions that were used to create the training sets. However, half of the test sessions involve the same performers as those seen in the training set, while the other half of the sessions involve performers that were not seen in the training phase. An example of the output from a network for a single motion capture session that it has not classified correctly is shown in figure 1, and an example with correct classification (of a session with a different performer) by the same network is shown in figure 2. In both cases the network is responding to input from a session with a performer who was not seen in the training set. The performance results are shown in table 2 and, as expected, recognition was higher for familiar musicians.

Much like with the full body networks, training past a certain number of epochs does not seem to offer significant improvement in classification performance. For the layer recurrent network with 9 hidden units, for instance, the same number of correct classifications continue to be made past the first 20 epochs up until 70 epochs at which point the performance begins to degrade. This general result was consistent across the recurrent networks, with the expected eventual decrease in performance for those that were trained long enough. The multilayer perceptrons all reached stopping conditions within 25 to 37 epochs.

network	h.u.	known		unknown	
type		correct	avg. range	correct	avg. range
MLP	3	12	0.000169	2	0.0000482
MLP	4	14	0.130	12	0.135
MLP	5	11	0.0000276	10	0.095
MLP	9	14	0.022	12	0.066
MLP	15	12	0.000972	6	0.241
LRN	3	14	0.0000316	8	0.000158
LRN	4	14	0.0000523	9	0.001
LRN	5	12	0.000325	7	0.000249
LRN	9	12	0.011	9	0.042
LRN	15	10	0.0000258	9	0.000526

Table 2: Networks were trained on the set of lower body angles and tested on previously unseen data from known and unknown performers. This table gives the network type, the number of hidden units in the network (h.u.), the number of correct classifications out of a possible 16 (known) and 16 (unknown) and the average required error tolerance range for correct classifications. MP indicates that the network is a Multilayer Perceptron, and LRN indicates a Layer Recurrent Network. The average range is the average of the necessary error tolerance ranges given over all the motion capture sessions in the test set for the given network. The larger the average range, the greater the required error tolerance in order for the network to make classifications correctly.

There doesn't seem to be a consistent improvement in performance when recurrent connections are used. The multilayer perceptrons perform quite well despite not having any representation of sequence; this is possibly because there is a lot of information in posture. Leg stance, independent of movement, reveals enough that there is reasonable classification performance with that input alone. Perhaps there is another subset of data that might require sequential information in making correct classifications. The next section considers networks trained with just centre of mass data.

3.3 Centre of Mass Networks

The third set of networks were trained only on a 2-dimensional projection of the centre of mass data, that eliminates the axis corresponding to height. This was used to avoid the possibility of the network simply memorising performer heights, much like the previous section used angles instead of positional data to avoid the same problem. Considering the centre of mass removes the possibility of the networks recognizing lower body stance. The networks had 2 input units (one for each axis of the two-dimensional projection) and, as with the previous sections, there was one output unit.

3.3.1 Performance

Once again recurrent connections do not seem to improve classification performance. Networks were trained with 3 to 6 hidden units and the best performance was seen with 4 hidden units in both the multilayer perceptron and the layer recurrent network. These results are shown in table 3. Similar to previous sections, network training was stopped where cross validation performance was highest.

network	h.u.	known		unknown	
type		correct	avg. range	correct	avg. range
MLP	3	15	0.00000195	9	0.0142
MLP	4	16	0.00000219	9	0.0202
MLP	6	16	0.00000412	6	0.00681
LRN	3	9	0.0423	5	0.000639
LRN	4	16	0.00224	9	0.00436
LRN	6	14	0.162	7	0.575

Table 3: Networks were trained on the floor of the centre of mass and tested on previously unseen data from known and unknown performers. This table gives the network type, the number of hidden units in the network (h.u.), the number of correct classifications out of a possible 16 (known) and 16 (unknown) and the average required error tolerance range for correct classifications. MP indicates that the network is a Multilayer Perceptron, and LRN indicates a Layer Recurrent Network. The average range is the average of the necessary error tolerance ranges given over all the motion capture sessions in the test set for the given network.

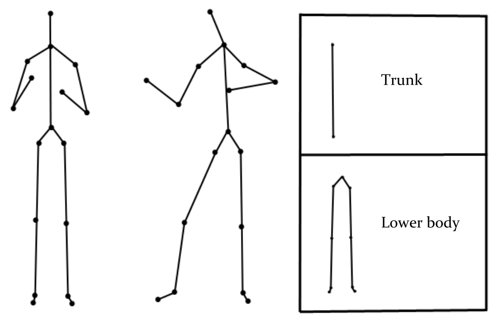

Figure 3: Kinematic figures created from motion capture data.

As anticipated, recognition of familiar musicians was higher than that of previously unseen musicians, although table 3 shows that the networks do generalize somewhat. The layer recurrent networks have similar tolerance ranges for both sets of data, but the multilayer perceptrons have a marked difference in that measurement, showing what could be interpreted as a stronger degree of confidence when classifying familiar musicians than when classifying novel musicians.

The results in this section suggest poor generalisation on the part of the networks. More specifically, while they generalise reasonably well to new data from the same musicians, they do not generalise well to new musicians, and this suggests that it is not truly instrument recognition that they have learned but something specific to the sets of musicians in the training set. A step towards improving upon this would be to train networks with data from a greater variety of musicians and compare the results.

4. COMPARISON WITH HUMAN OBSERVERS

The network performance can be considered alongside human performance in related perception tasks. At the IDMIL

	FB	LB	CM
Human	100	48 - 60	48 - 68
MP	100	75	56
LRN	100	56	56

Table 4: Humans were shown kinematic body displays and networks were given corresponding motion capture data. This table lists the percentage of displays (or files) where the subject (or network) correctly classified the instrument. FB is the full kinematic figure or the full set of body angles, LB is the lower body figure or the lower body motion capture angles, and CM is the trunk kinematic figure or the floor of the centre of mass.

there have been studies with human participants who were asked to identify various details of a performance when presented with restricted sets of data [1]. The same motion capture data from violists and clarinettists playing J. Brahms' Sonata Op. 120 No. 1 were used to construct kinematic stick figures like the one shown in figure 3. This is the same data used to create the testing sets of previously unseen performers in the work reported above, and so only that half of the testing results are considered in this section. The studies with human participants considered what sets of data were sufficient for human perception of instrument identification. Similarly, the work reported here used machine learning to investigate which subsets of data contain sufficient information for a classification to be made. Table 4 presents the human subject data alongside the results from the best performing trained networks in each category.

Humans and ANNs alike have no difficulty classifying instruments when shown the full set of body movements (i.e. the full kinematic stick figure for the humans and the angle data for the full body for the ANNs). When compared with human performance on similar tasks, the feedforward network trained on lower body data performs surprisingly well. It seems likely that this network is recognising patterns in posture since with a feedforward network there is no sense of order or sequence or time with regards to the input. (Whereas with a recurrent network, the order in which the input patterns are presented does affect the output.)

The network performance in the centre of mass case is comparable to that of the human participants in the trunk condition of the perception study. It would be interesting to investigate how human perception changes if the kinematic figures are based on familiar musicians. That is, if the data comes from musicians the study participants have seen perform.

5. CONCLUSIONS

Multilayer perceptrons and layer recurrent networks of comparable sizes performed similarly in classifying subsets of motion capture data. The recurrent networks did not appear to have an advantage and, in the case of the lower body networks, were even out-performed by the feedforward networks. Networks were trained with a number of hidden units ranging from 2 to 15 and, in both the lower body condition and the centre of mass condition, were found to have the highest performance on the test data with only 4 hidden units. In all cases, it was possible to overtrain the networks if very strict stopping conditions were set. Performance during testing was found to be highest for networks that were trained for less than 100 epochs and sometimes as few as 25.

Instrument recognition rates of the ANNs were found to be commensurate with that of human observers in related perception tasks for the various conditions. Both the network classification tasks and the human perception tasks were based on the same motion capture data. This similar level of performance of ANNs to that of humans suggests the potential for ANNs to be used in automated methods of movement analysis of large databases.

As an extension of this work, these networks would be good candidates for interpretation as they are fairly small. This could offer additional insight if the difference in performance on known and unknown musicians is uncovered. Future work will also involve simulations with additional neural network architecture variations to investigate if network performance can be improved overall, and if there are some conditions under which the recurrent connections increase instrument recognition rates over architectures that do not encode sequence.

6. ACKNOWLEDGMENTS

This research was partially funded by a Natural Sciences and Engineering Research Council of Canada (NSERC)-Create fellowship for Auditory Cognitive Neuroscience to the first author, and an NSERC Discovery Grant to the second author.

7. REFERENCES

[1] D. Bernardin, M. Trivisonno, M. M. Wanderley, B. Bardy, and T. Stoffregen. Visual perception of musicians' instrument from kinematic displays, 2010. Input Devices and Music Interaction Laboratory Unpublished Report.

[2] S. Dahl and A. Friberg. Visual perception of expressiveness in musicians' body movements. *Music Perception: An Interdisciplinary Journal*, 24(5), 2007.

[3] J. W. Davidson. Visual perception of performance manner in the movements of solo musicians. *Psychology of music*, 21(2):103–113, 1993.

[4] J. W. Davidson. Qualitative insights into the use of expressive body movement in solo piano performance: a case study approach. *Psychology of Music*, 35(3):281–401, 2007.

[5] S. Haykin. *Neural Networks and Learning Machines*. Number v. 10 in Neural networks and learning machines. Prentice Hall, 2009.

[6] V. Verfaille, O. Quek, and M. M. Wanderley. Sonification of musician's ancillary gestures. In *Proceedings of the 12th International Conference on Auditory Display*, pages 194–197, 2006.

[7] B. W. Vines, C. L. Krumhansl, M. M. Wanderley, and D. J. Levitin. Cross-modal interactions in the perception of musical performance. *Cognition*, 101:80–113, 2006.

[8] M. M. Wanderley. Quantitative analysis of non-obvious performer gestures. In I. Wachsmuth and T. Sowa, editors, *Gesture and Sign Language in Human-Computer Interaction*, pages 241–253. Springer Verlag, 2002.