Decoupling Expressiveness and Body-Mechanics in Human Motion

Gustavo E. Boehs, Milton L. H. Vieira, Clovis G. Pereira
Departamento de Expressso Grafica
Universidade Federal de Santa Catarina
Florianopolis, Brazil
Email: gustavo.boehs@ufsc.br

***Abstract*—Modern motion capturing systems can accurately store human motion with high precision. Editing this kind of data is troublesome, due to the amount and complexity of data. In this paper, we present a method for decoupling the aspects of human motion that are strictly related to locomotion and balance, from other movements that may convey expressiveness and intentionality. We then demonstrate how this decoupling can be useful in creating variations of the original motion, or in mixing different actions together.**

I. Introduction

Realistic motion synthesis of the human movement is a difficult task. Therefore, many content creation pipelines rely on the use of motion capturing. Processing motion captured animations manually is time-consuming, due to the high density of the data. Many techniques have been developed to ease the manipulation of these animations and to re-purpose captured data to new applications.

A pervasive theme in the literature of motion capture processing is style transferring. The main idea is that by decoupling the stylistic component and the functional component of animations, a handful of samples can be re-combined extensively. In these works, the stylistic component refers to human moods like happy, angry or sad, and the functional component relates to any hu-man action like to run, to kick, to jump.

Our work aims at decoupling motion captured animation in a novel way by separating a Body-Mechanics Component from an Expressiveness Component. The intuition for our approach is that parts of the human movement are merely a function of the character's dynamics and, therefore, unintentional. For example, as a person walks faster its arms will instinctively swing wider, to keep the body in balance. Nonetheless, other movements are performed by an actor independently of the bodies' dynamics, such as arms that swing in the air to gesture.

We implement our approach using a neural network to determine a correlation between a set of parameters relevant to the character's dynamics and its pose, thus recovering the Body-Mechanics Component of the movement. We demonstrate the effectiveness of our approach by modulating, blending and re-purposing motion capture animations decoupled in this manner. Such applications were not possible using previous decoupling methods.

Our main contributions are:

- Establishing an alternative paradigm for de-coupling motion capture animations
- Formulating a method capable of generating non-expressive character poses from dynamic parameters

II. Background

A. Style Decoupling

The idea of decoupling the transforms that determine an emotional style from purposeful action was initially proposed by Amaya [1]. This method consisted in transferring the speed and amplitudes of one motion to another. The method has its application restricted to short animation cycles, and a user must manually determine the correspondence between these cycles. Hsu et al. [2] proposed a linear time-invariant model that can represent stylistic differences between similar motions. It relies on a correspondence algorithm that aligns motions automatically.

Brand and Hertzmann [3] have proposed a data-driven approach that builds a statistical model for interpreting motion variations caused by style. Another data-driven approach [4] presents the possibility of further decoupling the style of the movement from the identity of its actor. More recently Xia, et al. [5] have proposed an auto-regressive model that builds a neighbour mixture from a source database to stylize unlabelled heterogeneous human motion. This model has been extended by Yumer and Mitra [6] to support motions that differ significantly from those existing in the database.

B. Data Driven Pose Synthesis

The human body has many degrees of freedom, but not all poses are valid from an anatomical standpoint, and even fewer are perceived as natural looking. One way to constrain the many possible outcomes in generating human poses is to learn joint limits implicitly from a database, thus restricting the set of solutions to this problem.

Grochow, et al. [7] used the Scaled Gaussian Process Model Latent Variable (SGPLVM) to confine the results of inverse kinematics to a latent space determined by a set of sample motions, resulting in an IK can solution that is constrained to natural looking poses.

Lee, et al. [8] proposed the concept of motion fields, as an alternative to motion graphs. The main idea is to use the near-

est neighbours in a motion state database and determine the most likely character reaction to local and global perturbations.

Some approaches to the synthesis of human poses rely on deep learning. Holden, et al. [9] use autoencoders to learn the latent space of the entire unlabelled CMU Motion Capture Library (hereinafter referred to as the CMU dataset). Due to the properties of autoencoders, they can thus recover naturally looking human poses from ill-looking data.

Another use of deep learning, more closely related to what we propose in our method is that of Holden, et al. [10]. In that work, a Convolution Neural Network (CNN) is used to determine the correlation between the trajectory of the character and the position of its end effectors, to its final pose.

III. Method

Our main goal is to decouple human motion into two components: a Body-Mechanics Component, and an Expressiveness Component.

We define the Body-Mechanics Component as being that part of the actor's pose which is strictly related to the way humans move and keep in balance. We rely on the assumption that given a set of parameters related to the character's dynamics it is possible to infer the character's pose.

We define the Expressiveness Component as anything in the pose of the actor that is not a direct function of its movement and balance. We assume movements in this component to be purposeful and expressive. It is important to note a clear distinction between this definition of the expressiveness component and the definition of a stylistic component in previous works.

In previous works the style of the character refers to a mood in which the action is performed, such as happy or sad. This moods consist mostly of changes in the amplitude and speed of the movement. In our work expressiveness means any movement that is not a direct function of locomotion and balance, thus encompassing completely distinct actions like head articulation, hand gesturing, crossing legs, and so on.

We summarize our approach in the following steps:

1) Select neutral, non-expressive, motions from a motion capture database and generate parameters that describe the dynamics of such movements.
2) Train a neural network to establish the correlation between the actor's dynamic parameters and its poses.
3) Use the trained neural network to recover the Body-Mechanics Component of a given set of dynamic parameters.
4) Calculate the delta between the original pose and the Body-Mechanics Component to retrieve the Expressiveness Component.

A. Data Selection and Parameter Generation

All motion captured animation used to train the neural network was selected from the CMU dataset [11]. The specific non-expressive clips used to train the network were selected by people following this set of heuristics:

- Actors may be still or moving around in space.
- The actor's head must be facing forward and must not change direction unless if as a function of the bodies dynamics, ex: looks backwards when walking backwards.
- The actor's arms and legs must be standing or moving in conjunction with the remainder of the body;
- Actors may not be performing any symbolic gestures with any part of their bodies.

The chosen representation for the actor's pose was the local orientation of each of its joints, in the form of quaternions. Using joint orientations, instead of positions is beneficial for it is invariant to actor's size and proportions. Strictly for the description of the pose, we have discarded the transforms of the hips joint to remove the influence of the actor's global position and orientation.

The CMU dataset, like many other motion capture datasets, consists of purely cinematic data. As we intend to train a neural network capable of generating poses as a function of the dynamics of the movement, we generate the following features from the original poses: hips velocity in the latest instant (v_t), hips velocity in the latest third of a second ($v_{t'}$), footstep detection and feet air time, the hips' relative height to the floor (h), and the orientation of the thighs. These features were selected empirically in a trial and error process, based on their ability to disambiguate the data.

The hips velocity (v_t) is calculated by computing the differential between the global position of the hips in the current time (x) and the global position of the hips in the previous instant (x'). We have also divide the velocity vector by the actor's height, so that it is invariant to body size. For the hips velocity in the latest third of a second ($v_{t'}$) we averaged the previously calculated velocity vectors (v_t) in that period. For an animation running at 60Hz that means 20 samples.

The hips velocity in the current instant is instrumental in detecting differences between various modes of locomotion; the combination of this parameter with the velocity at a more prolonged span of time helped disambiguate samples with sharp changes in direction such as in jumps and 180 turns.

Like Holden, et al. [10] we have found that using footsteps is a useful parameter in disambiguating otherwise similar human poses. Our footstep detection is based on what is proposed by Lee, et al. [12], which means it is triggered when the height of the foot and its speed drop below a certain threshold. We have added a filtering term that favors time continuity to avoid the occasional false negatives yielded by the original algorithm. Additionally, we have found that using the duration for which the foot has been in the air as a parameter, can further disambiguate similar looking poses. This feature is useful for it holds implicit information about the actor's anterior states.

The hips' relative height to the floor (h), is calculated dividing the hips' height in the present instant by its height at a neutral T pose . It is an important feature to differentiate the way the body balances at various conditions, like standing, squatting, and tip toeing.

Finally, to improve the disambiguation within different states of movements where the dynamics are almost constant

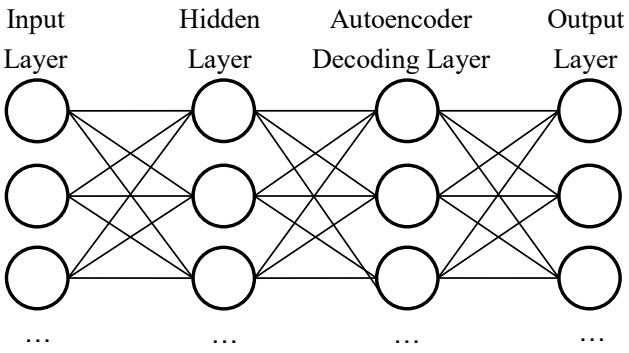

Fig. 1. Neural Network topology. The input layer consists of the body dynamics features. The hidden layer contains features that correlate inputs and the poses as encoded by the denoise autoencoder. Finally, the decoding layer correlates the encoded poses to the poses in their original form (quaternions).

(eg.: walking forward), we use the orientation of the actor's thighs.

B. Learning

In our method, a neural network is used to determine a correlation between the previously described parameters and the actual body pose of the so-called Body-Mechanics Component.

Using a network with a single hidden layered to perform such a task will, most of the times, yield visually satisfactory results. However, eventually, due to the hierarchical nature of the digital skeleton, small local errors may accumulate into drastically unnatural looking poses. That occurs because errors in lower levels of the hierarchy, like a spine joint, affect the poses of all higher-level joints.

To overcome this obstacle, we use an intermediary representation of the actor's pose constructed by a denoise autoencoder, as proposed by Vincent, et al. [13].

A denoise autoencoder consists of a regular autoencoder where the inputs are randomly corrupted. The underlying assumption is that the data bears redundancy, this enables the autoencoder to retrieve the original data from partially corrupted data.

In practice, it enforces the correlations within the quaternion dimensions of a joint, and in between one joint and all the others, effectively avoiding unnatural looking poses and out-of-limit joint rotations. We have trained the autoencoder with a large number of unlabelled samples gathered randomly from the CMU dataset.

A network is then trained to establish a correlation amongst the parameters that described the actor's dynamics (as detailed in section 3.1) and its pose, de-scribed as features of the denoise autoencoder.

Finally, we create a network with two hidden layers by stacking together the input, and hidden layers of the correlation network and the output layer of the autoencoder network, which decodes poses back to quaternion data. Figure 1 illustrates the network's topology.

C. Synthesizing the Body-Mechanics Component

The synthesis of a pose for the Body-Mechanics Component of the movement is a process that can be performed in real-time. We stream motion capture data from a file, or an on-line system, and then process this data to generate the same parameters that were used to train the network.

The result of simulating the network with such parameters is the orientation of all joints in the body, except that of the hips. We then use the hips' orientation and global position from the original motion.

D. Synthesizing the Expressiveness Component

As a corollary of our previously stated definitions, the synthesis of the Expressiveness Component can be summarized as the difference between the original pose and its Body-Mechanics Component.

Considering the most common representation of quaternion orientations in animation software (4x4 matrices) we describe the difference between these elements as:

$$E = O.B^{-1} \tag{1}$$

Where E is the matrix for the Expressive Component, O is the matrix for the original pose and B is the matrix for the Body-Mechanics Component.

Since no two persons move the same, and since the Body-Mechanics Component is composed of contributions of many different people, this operation will bear a certain imprecision. Due to the hierarchical nature of the skeletal representation, these errors sum up yielding significant distortions in the extremities of the body.

We, therefore, try to discard near zero contributions to the Expressive Component. For that, we multiply the contribution of each joint by an activity index. The activity index is a function of the angle of each joint in the Expressive Component. The following equation is used to compute this index:

$$A = (tansig(a).g)^b \tag{2}$$

Where a is the angle of each joint, and tansig is the tangential sigmoidal function; g and b are parameters that can be tweaked by the user to control the filtering. Increasing g will filter angles of a higher size, while increasing b will enhance the contrast between filtered in and filtered out values. The values of A are clamped between 0 and 1. For all examples in the results of this paper we have used the values 2 and 4, respectively for g and b.

Finally, we can resynthesize the Body-Mechanics Component as the difference between the original motion and the Expressive Component, thus recovering the data that had previously been filtered out. The following function describes such operation:

$$B' = O.E^{*-1} \tag{3}$$

To composite both components back together we need only to multiply them.

IV. Results

In total 96 Body-Mechanics only motion clips were selected from the CMU dataset. Sampled at 60Hz these clips sum up 18,009 poses. The denoise autoencoder was trained with a larger unlabeled dataset of 235 randomly selected clips sampled at 30Hz (33,531 samples). The dynamic features for all such clips were generated using an open source library [14].

We have trained shallow networks individually, in a greedy fashion and then stacked them together to form the final network.

The autoencoder used to encode the poses of the actor was trained using the Scaled Gradient Descent algorithm in MATLAB's Neural Network Toolbox. The best performance achieved for this individual network was a mean squared error of 0.0131 using 300 neurons. We have also tested training the autoencoder with 150, 50 and 15 neurons for which we have obtained mean squared errors of 0.0135, 0.0156, and 0.0196 respectively.

The layer correlating the dynamic parameters with the encoded actor poses was trained using 70% of the samples available and using the remaining 30% for testing. The algorithm of choice was also the Scaled Gradient Descent. The best performance achieved on this network was a mean squared error of 0.037.

After stacking the networks, we fine-tuned the resulting deep network to optimize its results. Here again, we used 70% the samples for training and the other 30% for testing. The best performance achieved was a mean squared error of 0.0286.

The on-line simulation of neural networks and further post-processing of the pose data was carried out in using an open source library for feed forward neural networks [15].

Figure 2 shows examples of actor's poses and their respectively decoupled Body-Mechanics and Expressiveness Components.

Although we do not use a neural network specifically designed to deal with time-series, it is possible to see time-sensitive behavior in the Body-Mechanics Component, like swinging arms and legs. We attribute that to the fact that temporal information is implicit in the in-put features such as the two velocity vectors and the feet's air time.

To demonstrate the applications of the proposed algorithm, we have used expressive animations from the CMU dataset in three different contexts: modulating actor's expressiveness, mixing components from various sources, and blending motions asynchronously. We have selected movements from different subjects within the CMU dataset in order to demonstrate the method's invariability to actor's height and body proportions. A list of the subject and motion numbers of all image and video examples is available in the supplemental material.

The videos corresponding to all such examples are in the supplement materials of this publication.

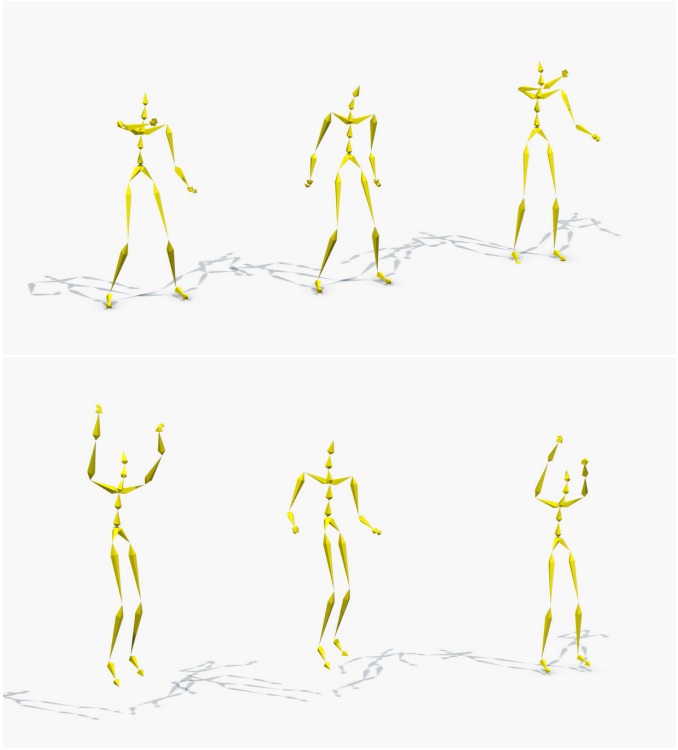

Fig. 2. Two examples of decoupled poses. Left-hand-side skeleton represent the original pose, the middle skeleton represents the Body-Mechanics Component generated by the deep neural network, and the right-hand-side skeleton represents the Expressiveness Component. The top image shows an actor washing a window; the bottom image shows an actor shooting a basketball.

A. Modulating expressiveness

A simple use of this decoupling algorithm is the ability to modulate an actor's expressiveness. Tuning expressiveness might be useful to correct or enhance, parts of the performance without distorting its body-mechanics. To achieved this, we multiply the orientation values of the Expressiveness Component before recoupling both components. Thus its influence diminishes or increases, depending on the multiplier. Figure 3 shows examples of this modulation.

B. Source Mixing

Another application of our algorithm is to combine the Body-Mechanics of one source with the Expressiveness from an entirely different source. Such approach can be useful to pick the best parts of an actor's performance and recombine it in one motion or to synthesize completely new motions. Figure 4 shows examples of this application.

Additionally, this could potentially be used to supplement parametrically synthesized human motions with intentional behavior.

C. Asynchronous Blending

A common problem in reusing motion captured animation is to blend the end of one motion with the begging of another, as poses might be completely different. A naive solution to this problem is to blend a transition over a specific set of frames.

Fig. 3. Examples of expressiveness modulation. Left-hand-side skeleton depicts expressiveness at 50% intensity; middle skeleton depicts the original pose; right-hand-side skeleton depicts the expressiveness at 150% intensity. The top image shows an actor boxing; the bottom image shows an actor dancing.

Fig. 4. Left-hand-side skeleton represents the source for the Body-Mechanics Component; middle skeleton represents the source for the Expressive Component, the right-hand-side skeleton displays the combination of both components from the different sources. The top image shows a combination of a jump and a gesture; the bottom image shows a combination of a walk and a laughter.

Using our technique, one can select different blending times for the Body-Mechanics and Expressiveness Components. Such approach can be advantageous for it allows greater control of theses transitions and may be instrumental in avoiding smoothing out important details. A sample of this application

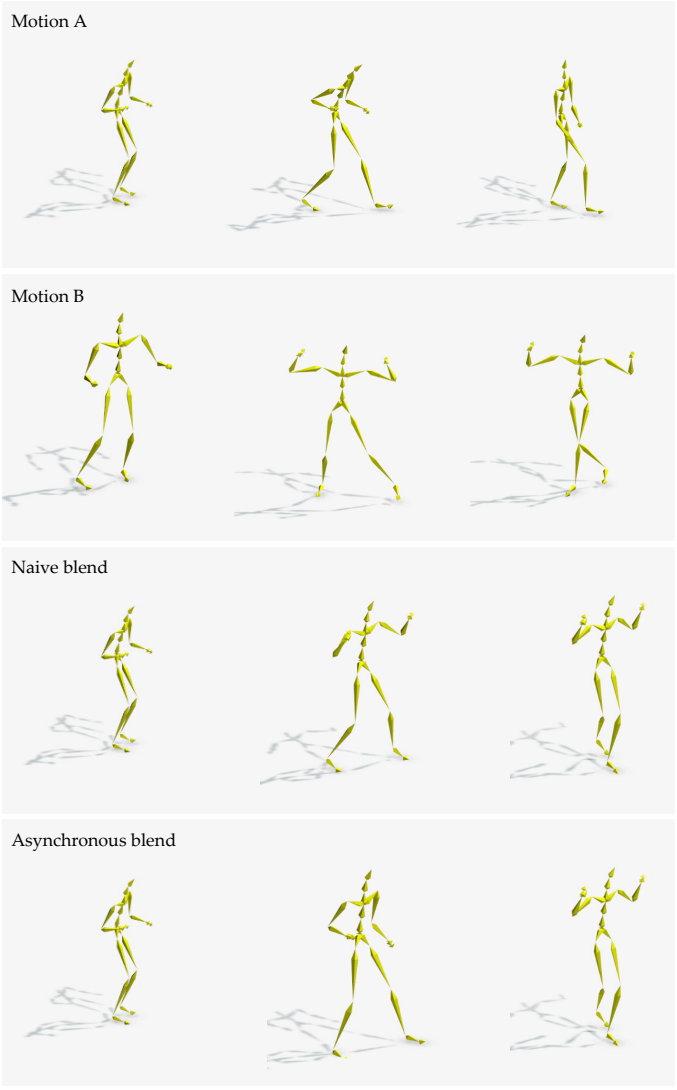

Fig. 5. Example of asynchronous blending. In a naive blend all movement in Motion A and motion B are blended at once. In an asynchronous blend body-mechanics and expressiveness can be blended at different moments to preserve details in the movement.

is shown in Figure 5.

Additionally, decoupling the motion in two separate components can potentially help more sophisticated blending techniques, like motion graphs [16], [17], by reducing the size of the pose space.

D. Comparison to Related Work

Previous work in the decoupling of human movement has focused mostly on separating functional action from modes in which those actions are enacted (like happy or sad). Such methods have been proposed as ways to transform existing motion to new styles [1]–[3], [5], [6] or editing motion and styles independently [4].

We have proposed a new way to decouple movement where we separate the aspects of movement involved in locomotion and balance from the remaining components. This alternative

approach enables new applications that have been exemplified in this article, like modulating expressiveness, source mixing, and asynchronous blending. This is different than previous approaches where the motion of the whole body is transformed in time and space to mimic a certain style.

Previous authors have proposed correlating dynamic properties of character movement to final character pose [10]. But none have done so as a means to decouple this two kinds of movements.

V. Discussion

The basic idea of our work is to provide a new paradigm for repurposing previously captured animations. The amount of motion capture data available in open and private datasets makes its recycling desirable and diminishes the need for new capturing sessions, which can be time-consuming.

We have also identified the potential to use this method of motion decoupling with other motion synthesis and motion blending algorithms, to enhance their behaviors and results.

A current limitation of our method is scaling the size of the training samples, due to the need of human selected training data. Clips need to be verified for the absence of expressive movement, and for consistency regarding their dynamic parameters. Another reason for this limitation is using a foot detector that relies on foot height, and that cannot work on uneven terrains. Therefore, our current implementation uses only a portion of the CMU dataset and might not be unable to deal with some kinds of movements.

References

[1] *Emotion from Motion*, ser. GI '96. Toronto, Ont., Canada, Canada: Canadian Information Processing Society, 1996. [Online]. Available: http://dl.acm.org/citation.cfm?id=241020.241079

[2] E. Hsu, K. Pulli, and J. Popović, "Style translation for human motion," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 1082–1089, Jul. 2005. [Online]. Available: http://doi.acm.org/10.1145/1073204.1073315

[3] *Style Machines*, ser. SIGGRAPH '00. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000. [Online]. Available: http://dx.doi.org/10.1145/344779.344865

[4] *Synthesis and Editing of Personalized Stylistic Human Motion*, ser. I3D '10. New York, NY, USA: ACM, 2010. [Online]. Available: http://doi.acm.org/10.1145/1730804.1730811

[5] S. Xia, C. Wang, J. Chai, and J. Hodgins, "Realtime style transfer for unlabeled heterogeneous human motion," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 119:1–119:10, Jul. 2015. [Online]. Available: http://doi.acm.org/10.1145/2766999

[6] M. E. Yumer and N. J. Mitra, "Spectral style transfer for human motion between independent actions," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 137:1–137:8, Jul. 2016. [Online]. Available: http://doi.acm.org/10.1145/2897824.2925955

[7] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popović, "Style-based inverse kinematics," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 522–531, Aug. 2004. [Online]. Available: http://doi.acm.org/10.1145/1015706.1015755

[8] Y. Lee, K. Wampler, G. Bernstein, J. Popovic, and Z. Popovic, "Motion fields for interactive character locomotion," *ACM Trans. Graph.*, vol. 29, no. 6, pp. 1–8, 2010.

[9] *Learning Motion Manifolds with Convolutional Autoencoders*, ser. SA '15. New York, NY, USA: ACM, 2015. [Online]. Available: http://doi.acm.org/10.1145/2820903.2820918

[10] D. Holden, J. Saito, and T. Komura, "A deep learning framework for character motion synthesis and editing," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 138:1–138:11, Jul. 2016. [Online]. Available: http://doi.acm.org/10.1145/2897824.2925975

[11] CMU, "Carnegie-mellon mocap database." [Online]. Available: http://mocap.cs.cmu.edu/

[12] J. Lee, J. Chai, P. S. A. Reitsma, J. K. Hodgins, and N. S. Pollard, "Interactive control of avatars animated with human motion data," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 491–500, 2002.

[13] *Extracting and Composing Robust Features with Denoising Autoencoders*, ser. ICML '08. New York, NY, USA: ACM, 2008. [Online]. Available: http://doi.acm.org/10.1145/1390156.1390294

[14] G. Boehs, "fabricutils," 2017. [Online]. Available: https://github.com/gustavoeb/fabricUtils

[15] ——, "feneuralnet," 2017. [Online]. Available: https://github.com/gustavoeb/feNeuralNet

[16] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 473–482, Jul. 2002. [Online]. Available: http://doi.acm.org/10.1145/566654.566605

[17] *Dynamic Motion Graphs*. ACM, 2012.