WalkNet: A Neural-Network-Based Interactive Walking Controller

Omid Alemi ✉ and Philippe Pasquier

School of Interactive Arts + Technology
Simon Fraser University, Surrey, BC, Canada
{oalemi, pasquier}@sfu.ca

Abstract. We present WalkNet, an interactive agent walking movement controller based on neural networks. WalkNet supports controlling the agents walking movements with high-level factors that are semantically meaningful, providing an interface between the agent and its movements in such a way that the characteristics of the movements can be directly determined by the internal state of the agent. The controlling factors are defined across the dimensions of planning, affect expression, and personal movement signature. WalkNet employs Factored, Conditional Restricted Boltzmann Machines to learn and generate movements. We train the model on a corpus of motion capture data that contains movements from multiple human subjects, multiple affect expressions, and multiple walking trajectories. The generation process is real-time and is not memory intensive. WalkNet can be used both in interactive scenarios in which it is controlled by a human user and in scenarios in which it is driven by another AI component.

Keywords: agent movement · machine learning · movement animation · affective agents

1 Introduction

Data-driven movement animation manipulation and generation techniques use recorded motion capture data to preserve the realism of their output while providing some level of control and manipulation. This makes them more suitable for generating affect-expressive movements, compared to the physics-based approaches to modelling and generating movement animation. Data-driven techniques bring the possibility of augmenting a corpus of motion capture data so that human animators have more assets at their disposal. Furthermore, one can use movement generation models in interactive scenarios, where a human user or an algorithm controls the behaviour of the animated agent in real-time.

With the increasing demand for content for nonlinear media such as video games, a movement controller that supports generating movements in real-time based on the given descriptions has applications in AI-based agent animation, interactive agent control, as well as crowd simulation.

Data-driven methods allow for manipulation of the motion capture data, either by concatenating, blending, or learning and then generating data. Concatenation methods repeat and reuse the movements in a motion capture corpus by rearranging them, making longer streams of movements from shorter segments. In blending, the representations of two or more motion capture segments are combined to create a new segment that exhibits characteristics from the blended segments. Compared to other techniques, machine learning models are better at generalizing over the variations in the data and generating movements that do not exist in their training corpus. Some of the machine learning techniques also provide mechanisms for controlling and manipulating what is being generated, making them suitable for controlling virtual agents.

The body of the research on machine-learning-based movement generation has some challenges. Controlling the movements of an agent requires a description of the movement to be generated, and a machine learning model that is capable of mapping those descriptions to movement, in real-time. In this regard, the majority of the works suffer from one or more of the following: (1) they do not support controlling the generated movements (e.g., [3]), (2) they only support controlling a single factor (e.g., [1]), (3) the controlling factor is often not clearly defined with respect to an agent's internal state(e.g., [13]), or (4) the generation process is computationally and/or memory intensive (e.g., [14]).

In order to overcome the above limitations, we present WalkNet, a walking movement controller for animated virtual agents. At its core, WalkNet uses a neural network to learn and generate its movements based on a set of given controlling factors. The factors are chosen to work directly with the internal state of the agent, corresponding to the planning, expression, and personal movement signature dimensions of movement. In future, we intent to extend the model to support controlling the functional dimension as well. The agent can plan its walking movements based on any given trajectory. The affective state is modelled by the valence and arousal dimensions of affect. Furthermore, the movement generation model is capable of exhibiting distinctive personal movement signatures (styles). This allows for using the same model for a group of agents that each portray a different character. The main contributions of our approach are summarized below:

- Walknet provides control over multiple dimensions of movement in a single model.
- Learned over a limited sample of affective states (i.e., only high, neutral, and low points) and only two human subjects, WalkNet learns a generalized space of affect and movement signature.
- The generation process is real-time. Unlike graph and tree based structures, there is no need for search or optimization to generate desired movements.

2 Background and Related Work

Controlling Movement Generation In data-driven movement-generation approaches, different techniques, and the combinations of them are used to control

and manipulate the characteristics and qualities of motion capture data. These include organizing the data using specialized data structures, such as motion graphs [8,5], as well as blending and interpolating multiple segments. Regarding the machine learning models, there are multiple ways that they support controlling the generation: 1) Train a separate model for each point in the factor space. Each model is trained only on the data that correspond to that particular point, thus only imitating the same factor value. To control the generation, one has to switch between the models. 2) Using a parametric probability distribution, in which the parameters of the distribution are a function of the controlling factors [6], allows for controlling the statistical characteristics of the generated data. 3) By designing the machine learning model in a way that provides a mechanism for a factor variable to control the characteristics of the generated movements. In particular, Factored Conditional Restricted Boltzmann Machine (FCRBM) uses a context variable (Figure 3.b) that controls the behaviour of the network through gated connection between the observations and the hidden variables [12].

Machine Learning Methods for Movement Generation Machine learning models that are used for learning and generating motion capture data range from dimensionality reduction (DR) techniques (e.g., [10]), to the Gaussian Process Latent Variable Models (GPLVMs) (e.g., [14]), Hidden Markov Models (HMMs) (e.g., [2]), temporal variations of the Restricted Boltzmann Machines (e.g., [12,1]), Recurrent Neural Networks (e.g, [3]), and Convolutional Autoencoders combined with Feed-Forward Networks (e.g., [7]).

DR techniques do not handle the temporality of the motion capture data. Furthermore, the dimensionality-reduction-based techniques rely on preprocessing steps such as sequence alignments and fixed-length representation of the data. The main limitation of the GPLVMs is that they demand heavy computational and memory resources, which makes them unsuitable for real-time generation. HMMs overcome the limitations of the two aforementioned families of models but provide a limited expressive power regarding capturing the variations in the data. Neural networks provide a better expressive power than HMMs. Convolutional Autoencoders have shown promising results in generating motion capture data and offline controlling [7]. Factored Conditional RBM (FCRBM), with its special architecture that is designed to support controlling the properties of the generated data, has shown to be able to generate movements in real-time, and learn a generalized space of the movement variations [12,1].

Affect-Expressive Movement Generation Taubert et al. [11] combine a Gaussian process latent variable model (GP-LVM) with a standard HMM that learns the dynamics of the handshakes, encoded by the emotion information. Samadani et al. [10] use functional principal component analysis (FPCA) to generate hand movements. Alemi et al. [1] train an FCRBM to control the valence and arousal dimensions of walking movements.

Our Approach We build WalkNet on top of the previous work by the same authors [1], extending the affect-expression control with the walking planning

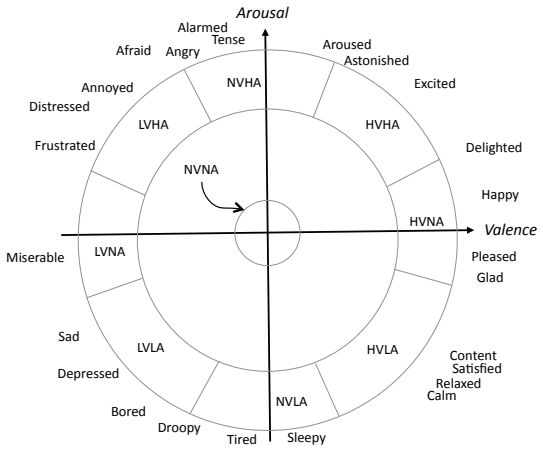


Fig. 1. The affect model described by valence and arousal dimensions with the 9 zones recorded in the training data. The mapping to the categorical emotion labels are based on Plutchik and Conte [9]. H: high, N: neutral, L: low, V: valence, and A: arousal.

and personal movement signature. Our work differs with graph-like structures as it does not require to build an explicit and fixed data-structure, does not require search and optimization for generating movement, and does not require storing the movement data for generation. It also differs from the work of Crnkovic-Friis and Crnkovic-Friis [3] as it provides a mechanism to control the generated data. It allows for real-time and iterative generation compared to the work of Holden et al. [7].

3 Training Data

For training the model, we use a set of motion capture data that provides movements with variations in walking direction (planning), the valence and arousal levels (expression), and the personal movement signature. As we could not find a publicly available motion capture database that provides movements with such variations, we recorded our own set of training data. The complete data set is publicly accessible in the MoDa database[1].

The training data includes the movements of two professional actors and dancers (one female, one male). Each subject walks following a curved figure-8-shaped path. The turning variations in this pattern allow the machine learning model to learn a generalized space of turning directions. To capture a space of affect-expression, each subject performs each movement with nine different expressions along the valence and arousal dimensions [9], shown in Figure 1.

[1] `http://moda.movingstories.ca/projects/29-affective-motion-graph`

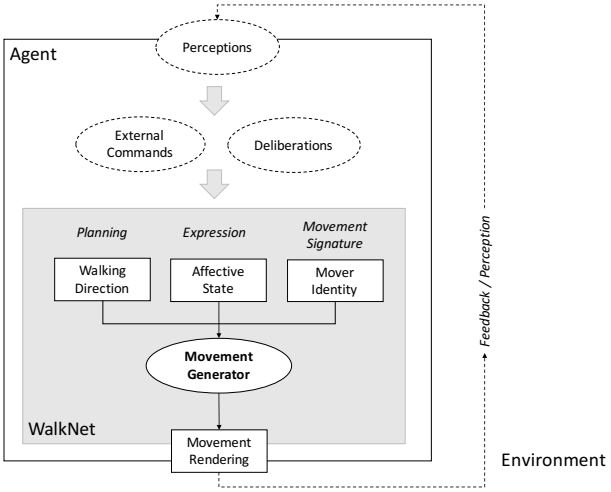


Fig. 2. The WalkNet controller, embedded in an agent model.

Using the dimensional representation of affect over the categorical systems allows for interpolation and extrapolation of the affect states, as well as transitions. Each valence and arousal combination is repeated four times to capture enough motor variabilities.

The original motion capture data consists of a skeleton with 30 joints, resulting in 93 dimensions including the root position, with their rotations represented in Euler angles. The data is captured at 120 frames-per-second. We use exponential maps [4] to represent joint angles to avoid loss of degrees-of-freedom and discontinuities. We replace the skeleton root orientation and translation by the delta values of the translational velocity of the root along the floor plane, as well as its rotational velocity along the axis perpendicular to the floor plane. We remove the dimensions of the data that are constant or zero and downsample the data to 30 frames-per-seconds. The final data set used for the training consist of 18 motion capture segments (2 subjects $\times$ 9 affective states), containing 37,562 frames in total, with 52-dimensional frames.

4 The Walking Controller

System Overview As shown in Figure 2, at the core of the WalkNet, the movement generator, a Factored, Conditional Restricted Boltzmann Machine (FCRBM), generates a continuous stream of movement. The movement stream is modulated by a set of controlling factors, determined from the internal state of the agent or through external commands. From an agency perspective, we organize these factors into different dimensions, mainly the *function*, the *planning*, the *expression*, and factors that together make the *personal movement signature* of the agent. WalkNet does not make any assumptions on how the agent

movement descriptor is set. Thus, making it flexible to be integrated into various agent models for different applications.

Agent Movement Descriptor We use an agent movement descriptor AMD to formalize the contributing factors to the agent's movements at time t along the dimensions of function (F), planning (P), expression (E), and personal movement signature (S):

$$\text{AMD}_t = \langle F_t, P_t, E_t, S_t \rangle$$

In WalkNet, the F is always set to walking. The planning dimension of walking is defined by $P_t = \langle D_t \rangle$ where D_t represents the direction that the agent intends to walk towards, relative to its current orientation. The expression dimension is defined by $E_t = \langle V_t, A_t \rangle$ where V_t and A_t stand for valance and arousal levels at time t respectively. Currently, we use the actor/performer's identity as a proxy to model the personal movement signature, through a weighted combination of a K-dimensional vector, representing K subjects:

$$S_t = \{I_t^1, I_t^2, .., I_t^K | \sum_k^K I_t^k = 1\}$$

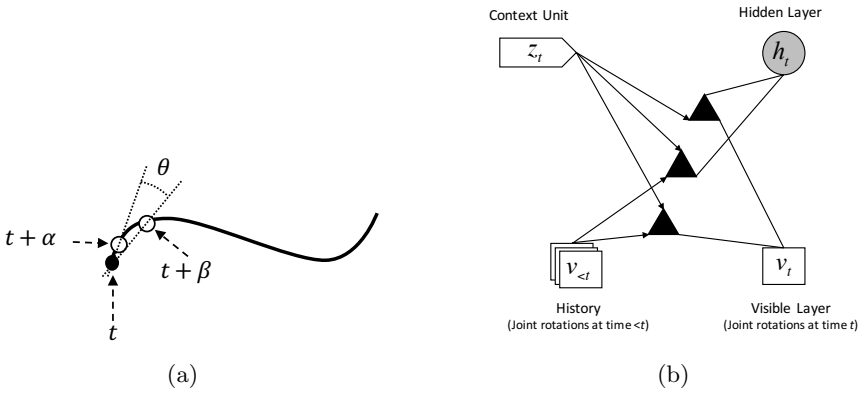
We recognize that this is a simple way of capturing movement signature. In the future, we plan on learning a representation that captures the personal movement signature.

Training Data Annotation Here we describe how we annotate our data to capture different states of the factors in the agent movement descriptor.

As we have two human subjects in our training data, we use a 2-dimensional label with a one-hot encoding scheme for the movement signature.

We use the valence and arousal representation of affect to annotate the expression of affect. Each movement segment in the training data is labeled with low, neutral, and high for both their valence and arousal levels. After experimenting with different ranges, we use the values of 1, 2, and 3 to represent low, neutral, and high levels in the annotations. Although the training labels are discrete, the valence and arousal values are continuous in nature, and for the generation, one can specify any real value within the range of $[0, 4]$, as the FCRBM is able to interpolate or extrapolate between those discrete states.

For annotating the heading direction, we determine the labels using a method that is inspired from Kover et al. [8]. For the label at frame t, considering the projection of the traveled path of the skeleton root on the ground floor, we select two points on the path, one at a very close distance to the current location, and another one at a slightly further location from the current location (Figure 3.a). We calculate the angle between the two lines that result from connecting the two chosen points and use this angle as a measure of the heading direction. After scaling the angle to have a value between -1 and +1, directions towards the right of the subject are associated with positive numbers, and directions towards the left of the subject are associated with negative numbers.



(a) (b)

Fig. 3. (a): Calculating the direction of the subject in the training data. (b): FCRBM's architecture with valence and arousal labels modulating the interactions between the past visible, current visible, and current hidden units.

Initial experiments showed that using only a one-dimensional label vector for modelling the direction parameter causes poor results when asking the model to generate movement for the values that are around the center of the continuum. The problem arises from the fact that the model associates high values with one end of the spectrum and low values with the other end of the spectrum, while semantically, there is no difference between each end. This issue is overcome by using a two-dimensional vector $D = [L, R]$ to annotate the two polarities of the direction. The two dimensions of this vector complement each other, following the relationship $R = 1 - L$ in a normalized case. Therefore, the direction is encoded as two labels, one for right and one for left.

As a result, each frame t of the training segment s is annotated with a 6-dimensional label of the form:

$$L_t^s = \langle I^{s^1}, I^{s^2}, V^s, A^s, R_t, L_t \rangle$$

Note that as the identity of the subject and the valence and arousal levels are fixed for each training segment, only R_t and L_t values are changed between each frame of the same segment.

Movement Generator We use an FCRBM to generate the movements of the agent. As shown in Figure 3.b, FCRBM learns the autoregressive, as well as the nonlinear temporal patterns in a time-series. Every weight in FCRBM is modulated by the value of its context unit, making it possible to change the energy landscape of the model by changing the value of the context unit, and effectively controlling the model's prediction. In WalkNet, the FCRBM learns to predict the next motion capture frame, given a recent history of the motion capture frames, as well as the movement descriptors fed to its context unit Z. This results in a predictive function in the form of:

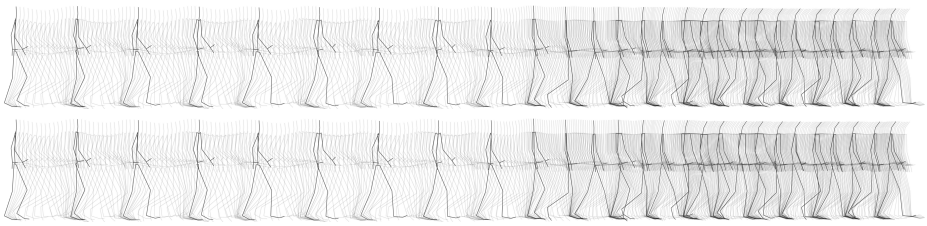


Fig. 4. WalkNet's output. Top: making a transition from a high valence and high arousal affective state to a low valence and low arousal state. Bottom: making a transition from a low valence and high arousal affective state to a high valence and low arousal state.

$$M_t = f(M_{<t}, Z_t), Z_t = \langle I_t^1, I_t^2, V_t, A_t, R_t, L_t \rangle$$

By iteratively calling this function and feeding it with the generated frames from the previous cycles, we can continuously generate movements that are modulated by the given descriptors.

5 Results

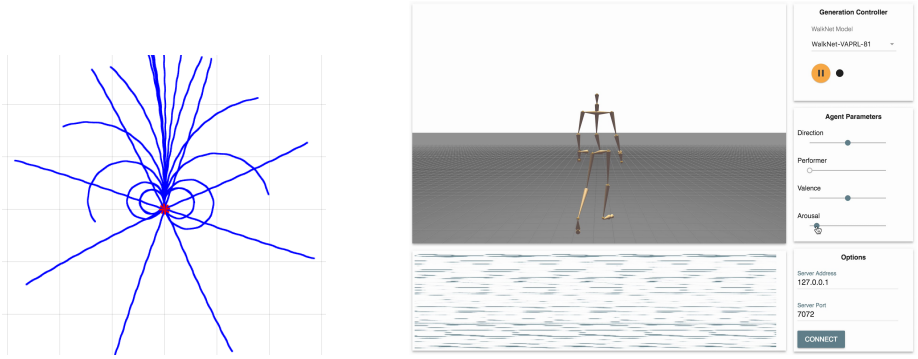
In this section, we demonstrate the capabilities of WalkNet in generating realistic motion capture data. We use an FCRBM with 150 hidden units and 400 factors, trained for 3000 epochs. The model takes 12 past motion capture frames as input and predicts the next frame, modulated by a vector of 6 dimensions (Z).

Affect Expression By specifying different values for the valence and arousal levels in the agent movement descriptors, WalkNet can generate a variety of affect expressions, even for those values that do not exist in the training data. This allows for generating walking movements for any point in a range of $[0, 4]$. With this, one can not only generate walking movements for high, neutral, and low levels of valence and arousal but also make transitions from one state to another (Figure 4).

In a previous work by the same authors [1], a study was conducted to validate the expressiveness of the movements. The analysis shows that independent human observers can successfully identify different levels of arousal. However, they can only correctly identify the low valence levels, and often confused the neural and high valence levels. The analysis reached the same results for the recorded movements of human actors as well. We believe that due to the lack of facial expression, recognition of valence through movements, as represented by a stick figure, is often challenging for humans.

Movement Signature WalkNet can generate signatures that are interpolations between the two actors. The difference in the generated movement signatures are demonstrated in the accompanied video[2].

[2] `https://youtu.be/3JBfGF4tsmA`



(a) (b)

Fig. 5. (a) The projection of the agent's movements on the ground floor plane, making turns with different angles. (b) The interactive controller.

Navigation As the results are demonstrated in Figure 5.a, different values for the direction factor generates movements along curves with different curvatures.

Interactive Control WalkNet through a graphical user interface (GUI) developed for this purpose. The GUI allows the user to choose the parameters of the model, while the agent's movements are rendered in 3D in real-time. A snapshot of the GUI is shown in Figure 5.b. A video of the GUI is also provided[2].

Generating each frame takes 0.0063 seconds on a MacBook Pro with an Intel(R) Core(TM) i7-4850HQ CPU at 2.30GHz. Thus, at 30 frames-per-second, it takes 0.1890 of a second to generate the movements for each second.

6 Conclusion and Future Work

This paper introduces WalkNet, a walking movement controller. It can generate realistically-looking walking movements in real-time, while modulating them using an agent movement descriptor that specifies the expression of affect through the movement, the walking direction, and the personal movement signature of the agent.

WalkNet is designed with integration into agent models in mind. It does not make any assumption on how the movement descriptor is specified, making it possible to be used in interactive scenarios, in which a user directly controls the agent's movements, or in scripted or AI-driven applications. For example, given a target path to follow, by observing the traveled path, the agent can continuously correct its course to stay on the target path.

In future, we plan to perform more formal and quantitative evaluation of the model. Furthermore, we intend to use more human subjects in the training data. Another future direction is to extend the model to include more than one type of movement (function). For example, allowing the agent to switch from walking to standing to sitting while performing hand gestures.

Acknowledgements

This work is funded by the Social Sciences and Humanities Research Council of Canada (SSHRC) through the Moving Stories Project, as well as the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

1. Alemi, O., Li, W., Pasquier, P.: Affect-Expressive Movement Generation with Factored Conditional Restricted Boltzmann Machines. In: Proceedings of the International Conference on Affective Computing and Intelligent Interaction (ACII). pp. 442–448 (2015)
2. Brand, M., Hertzmann, A.: Style Machines. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. pp. 183–192. ACM Press/Addison-Wesley Publishing Co. (2000)
3. Crnkovic-Friis, L., Crnkovic-Friis, L.: Generative Choreography using Deep Learning. In: Proceedings of the 7th International Conference on Computational Creativity (2016)
4. Grassia, F.S.: Practical Parameterization of Rotations Using the Exponential Map. Journal of Graphics Tools 3(3), 29–48 (1998)
5. Heck, R., Gleicher, M.: Parametric Motion Graphs. In: Proceedings of the 29th Representation Learning Workshop. International Conference on Machine Learning. pp. 129–136. ACM Press (2007)
6. Herzog, D., Krueger, V., Grest, D.: Parametric Hidden Markov Models for Recognition and Synthesis of Movements. In: Proceedings of the British Machine Vision Conference. pp. 163–172 (2008)
7. Holden, D., Saito, J., Komura, T.: A Deep Learning Framework for Character Motion Synthesis and Editing. ACM Transactions on Graphics (TOG) 35(4), 138–11 (2016)
8. Kovar, L., Gleicher, M., Pighin, F.: Motion Graphs. In: SIGGRAPH '02: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques. pp. 473–482. ACM Press (2002)
9. Plutchik, R., Conte, H.R.: Circumplex Models of Personality and Emotions. American Psychological Association (1997)
10. Samadani, A.A., Kubica, E., Gorbet, R., Kulić, D.: Perception and Generation of Affective Hand Movements. International Journal of Social Robotics 5(1), 35–51 (2013)
11. Taubert, N., Endres, D., Christensen, A., Giese, M.A.: Shaking Hands in Latent Space - Modeling Emotional Interactions with Gaussian Process Latent Variable Models. In: Advances in Artificial Intelligence, 34th Annual German Conference on AI. pp. 330–334. Springer (2011)
12. Taylor, G.W., Hinton, G.E.: Factored Conditional Restricted Boltzmann Machines for Modeling Motion Style. In: Proceedings of the 26th Annual International Conference on Machine Learning (2009)
13. Tilmanne, J., d'Alessandro, N., Astrinaki, M., Ravet, T.: Exploration of a Stylistic Motion Space Through Realtime Synthesis. In: Proceedings of the 9th International Conference on Computer Vision Theory and Applications. pp. 1–7 (2014)
14. Wang, J.M., Fleet, D.J., Hertzmann, A.: Multifactor Gaussian Process Models for Style-Content Separation. In: Proceedings of the 24th International Conference on Machine Learning. pp. 975–982. ACM Press (2007)