Expert Systems With Applications 48 (2016) 26–34

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Neural network for dynamic human motion prediction

Mohammad Bataineh*, Timothy Marler, Karim Abdel-Malek, Jasbir Arora

Virtual Soldier Research Program – Center for Computer-Aided Design, The University of Iowa, Iowa City, IA, USA

ARTICLE INFO

Keywords:
Digital human modeling
Artificial neural networks
Motion prediction
General regression neural network

ABSTRACT

Digital human models (DHMs) are critical for improved designs, injury prevention, and a better understanding of human behavior. Although many capabilities in the field are maturing, there are still opportunities for improvement, especially in motion prediction. Thus, this work investigates the use of an artificial neural network (ANN), specifically a general regression neural network (GRNN), to provide real-time computation of DHM motion prediction, where the underlying optimization problems are large and computationally complex. In initial experimentation, a GRNN is used successfully to simulate walking and jumping on a box while using physics-based human simulations as training data. Compared to direct computational simulations of dynamic motion, use of GRNN reduces the calculation time for each predicted motion from 1–40 min to a fraction of a second with no noticeable reduction in accuracy. This work lays the foundation for studying the effects of changes to training regiments on human performance.

1. Introduction

The use of digital humans is becoming more prevalent with the upstream design of any product or process that involves human interaction or human systems integration. In order to enhance the design process as effectively as possible, computationally fast human simulation and analysis become critical. The faster one can simulate and obtain feedback concerning human performance, the faster one can evaluate and refine designs. A cornerstone of human performance is simulated motion, which often requires dynamic analysis (consideration of forces, acceleration, and inertia, not just kinematics). However, accurate dynamic simulation and analysis can be computationally demanding, depending on the task being simulated. Therefore, this work presents the use of an artificial neural network (ANN) to provide fast motion simulation.

Whether the motion tasks are simulated using data-based methods (Chaffin, 2002; Moeslund, Hilton, & Krüger, 2006), which depend on motion capture systems to track, record, and reproduce human motion during various tasks, or using physics-based methods (Xiang et al., 2010), which primarily entail using optimization to predict motion, techniques for capturing one's history and the consequent strategy for completing the task continue to be a challenge. Why do different people with similar capabilities and size, for instance, enter the same vehicle in different ways? Although various human modeling methods can capture the nuances of one's motion or the cause and effect demonstrated with changes in parameters, few methods offer the ability to capture one's *strategy* in approaching a task without significant input from the user. Hence, there is a need for an algorithm that produces real-time motion prediction and can incorporate a history of experience.

Motion-capture-based methods are limited in terms of their ability to produce different motions that correspond to changes in the task parameters, because the underlying algorithm depends on prerecorded data that cannot be changed due to the change in the task conditions. In addition, these methods do not incorporate dynamics; they do not capture effects of loads and inertia. Alternatively, physics-based methods like predictive dynamics (PD), which is an optimization-based motion prediction algorithm (Xiang et al., 2010), tend to be more flexible in showing the effects of changes in task parameters, especially with respect to dynamics. In addition, these methods are predictive; they predict human performance with minimal dependence on prerecorded data. Computational speed, however, can be a limiting factor with PD, when real-time performance feedback is needed. Depending on the task being simulated and its settings, PD can require up to 40 min (the PD algorithm is run on a Windows 7 computer with an Intel® Core™ i3 processor and 8 GB of RAM), even with small changes in the configuration. A variety of techniques are being explored to address this challenge, and the use of an ANN is especially promising.

An ANN can be used for real-time motion prediction and can be integrated with the physics-based motion simulation method, PD, in order to improve the computational speed when predicting a motion task. An ANN also provides a platform for incorporating alternative sources for real-time motion prediction, such as motion

* Corresponding author. Tel.: +1 319 331 5454.
E-mail addresses: bataineh.moe@gmail.com (M. Bataineh), tmarler@engineering.uiowa.edu (T. Marler), amalek@engineering.uiowa.edu (K. Abdel-Malek), jasbir-arora@uiowa.edu (J. Arora).

http://dx.doi.org/10.1016/j.eswa.2015.11.020

capture data, if desired. Unlike other simulation tools, ANNs are capable of providing acceptable simulations for a problem without the need for complex time-consuming algorithms. This work (1) demonstrates the feasibility and advantages of using ANNs for direct motion prediction, and (2) presents the use of one of the ANN types as an appropriate network for successful simulation of such problems.

An ANN is a mathematical model for predicting system output, inspired by the structure and function of human biological neural networks. Compared to other simulation and statistical tools, ANN is fast and produces relatively accurate and acceptable simulations for complex systems. ANNs entail two steps or processes. First they are *trained* using some form of pre-existing data. Essentially, optimization is used to set model parameters. After its training process is completed, it is then *run* and provides relatively fast output given various input conditions. ANNs can be powerful tools for generalizing, which means providing accurate and acceptable results for all inputs conditions, many practical problems (Coit, Jackson, & Smith, 1998; Twomey & Smith, 1998), and hence have been used successfully in many digital human model (DHM)-related problems (Bataineh, 2015; Bataineh & Marler, 2013; Bataineh, Marler, & Abdel-Malek, 2013; Bu, Okamoto, & Tsuji, 2009; Kang, Kim, Park, & Kim, 2007; Li, Li, & Song, 2007; Zhang, Horváth, Molenbroek, & Snijders, 2010; Zhao, Zheng, & Wen, 2010). Motion-related applications include, but are not limited to, robotics and controller system motion, motion analysis, reconstruction of dynamic objects, and time-series dynamic prediction and classification. In general, the main use of ANNs has been focused on human model posture prediction (Jung & Park, 1994; Zhang et al., 2010) and motion prediction of robotics and dynamics systems (Frank, Davey, & Hunt, 2001; Stakem & AlRegib, 2008). One approach proposed the use of multiple ANNs in controlling a robot manipulator (Y. H. Kim & Lewis, 1999). The system was evaluated successfully on a two-link robot manipulator, demonstrating the ANN's ability to handle the nonlinear unknown parameters in the system manipulator. Moreover, the feedforward-backpropagation network, which was trained by gaits of various people, is used to recognize humans automatically (Yoo, Hwang, Moon, & Nixon, 2008). In other motion-related applications, ANNs have been used as an indirect source that led to providing improved motion predictions. Lung tumor motion during respiration was predicted in advance using ANNs (Isaksson, Jalden, & Murphy, 2005). Most of the preceding scholars use feedforward-backpropagation ANNs with single or few outputs to preserve accuracy.

So far, ANNs have been applied only to very specific scenarios in DHM problems and have not been developed for robust use with complex problems like whole-body dynamic motion prediction. In general, ANNs have been used to solve confined systems with a relatively small number of inputs and outputs. Most applications have involved feedforward-backpropagation networks, which have memory and accuracy issues when used with a large number of inputs and outputs. Thus, this work explores using other types of ANNs for relatively large and complex human-modeling problems.

The overarching hypothesis is that if designed/selected properly, ANNs can in fact be used to simulate human motion quickly and accurately, despite a relatively large number of outputs. We contend that a radial-basis network (RBN) is most appropriate, because it has the smallest number of parameters to be set when simulating a problem with a large number of outputs. In addition, it provides a global solution when optimizing the network parameter values (during the training process). Specifically, we propose using a general regression neural network (GRNN), which is a type of RBN. The work integrates GRNN with PD to increase the computational speed of PD with minimal detriment to accuracy. This is shown using two different simulated tasks with a large number of outputs (on the order of hundreds), both of which run in under one second. Eventually, PD can be replaced by GRNN, which is trained to provide a standalone instant motion simulation. The next section describes the necessary parameters (inputs and outputs) of the PD simulated tasks, as well as details of the underlying ANN architecture.

2. Methods

2.1. Digital human model and physics-based motion prediction

As a foundation for the proposed method, this section summarizes the digital human model as well as PD. This work capitalizes on and adds to a foundation of virtual human modeling capabilities, housed within a human model called Santos (Abdel-Malek et al., 2006; Abdel-Malek et al., 2007). Santos, as shown in Fig. 1, is a highly realistic, biomechanical computer-based human that predicts, among other things, static posture, dynamic motion, joint strength, and development of fatigue. Such capabilities can be used to predict and assess human function, providing task performance measures and ergonomic analysis. Thus, in a virtual world, Santos can help design and analyze various products and processes. In addition, Santos can help study and evaluate various restrictions and impediments, such as fatigue, reduced range of motion, environmental obstacles, etc.

A key aspect of any virtual human is the ability to simulate human posture and motion realistically and quickly while considering external and internal loads/forces. With respect to motion, there are traditionally two types of dynamics problems that need to be addressed. In the first problem, called forward dynamics, the external forces and torques on the system are known and the motion of the system is desired. The problem is solved by integrating the governing equations of motion forward in time using a numerical algorithm. In the second problem, called inverse dynamics, the motion of the system is known (i.e., from motion capture), and the forces and torques causing the motions are calculated using the equations of motion. Both of these problems can be solved using traditional multi-body dynamics software. The problem of *predictive dynamics* arises when one wants to simulate the human motion for any task. In this problem, both the joint torques and the motion of the joint are unknown. Therefore, the problem becomes more difficult to solve. With the PD approach, the joint angles (one for each degree of freedom, or DOF) essentially provide design variables that are determined through optimization. The objective function(s) is one or more human performance measure, such as energy consumption, discomfort, and joint displacement. Including the dynamic equations of motion as constraints then ensures that the laws of physics are satisfied.

The specifics of PD (Xiang, Arora, Rahmatalla, & Abdel-Malek, 2009; Xiang et al., 2010) are summarized as follows. In general, predicting dynamic human motion is approached as an optimization problem (Arora, 2004), as shown in Eq. 1. This formulation provides the context for the discussion of inputs and outputs used with the proposed neural network. Joint angle profiles over time are represented as B-spines, and the control points, which dictate the shape of the profile curve, serve as design variables in an optimization problem. The problem entails determining design variables $\boldsymbol{q}$, which represent the control points (i.e., joint angle profiles) of all body DOFs, in order to minimize the objective function, $f(\boldsymbol{q})$, subject to the physical equality ($h_i(\boldsymbol{q})$) and inequality ($g_j(\boldsymbol{q})$) constraints. The control points ($\boldsymbol{q}$) form B-splines for all DOFs that simulate the motion of the DHM. Having more control points in a B-spline leads to more accurate motion simulation but increases the number of design variables and can thus increase computational complexity. $\boldsymbol{q}_{MoCap}$ is a vector