Title:
An Electro-Mechanical Input Device for the Manipulation of Virtual Human Motion

Authors:
Thomas Kronfeld, tkro@cs.tu-chemnitz.de, Technische Universität Chemnitz
Guido Brunnett, brunnett@cs.tu-chemnitz.de, Technische Universität Chemnitz

Keywords:
Virtual Humans, Digital Factory, Human Factors, Virtual Prototyping

DOI: 10.14733/cadconfP.2018.332-336

Introduction:
The simulation of human work activity within digital factories is of growing importance because skilled workers are hard to find and the average age of the employees is increasing. However, for such simulations the motions of the virtual workers have to be created, which is highly time-consuming and as a consequence very expensive [4].

Motion simulation is based on key frame animation, which means that for every single frame of the simulation a posture of the virtual human model has to exist. Such animations can be generated either via motion capturing or by manually creating each posture [5]. Both methods have their benefits and drawbacks. Recording motion capture data requires a full-body motion capture system, which is very expensive in terms of acquisition and maintenance. Since existing motions can only rarely be used for a new task, there is a constant need of recording new motions. On the other hand, the resulting motion sequences are realistic and smooth. In a manual motion creation the planner has to construct a posture of the virtual human model for each single key frame. This includes dragging and rotating each joint into the desired position using the keyboard and the mouse and may lead to rather unrealistic motions. However, the designer has full control of the motion generation and he can adjust the motion whenever the environment changes.

To integrate the advantages of both approaches, we suggest to use a basic dataset of motion capture files which can then be modified using an intuitive input device. Although existing electro-mechanical devices can be used to create a new pose [2],[6], appropriate tools for the subsequent motion modification are still missing. Obviously, input devices that allow the modification of single poses only (see [7]) are not efficient to use. Rather, a method is needed to modify easily a whole motion sequence.

In this paper, we present a puppet device that is comparable in structure with the one developed in [7], but allows the modification of motion sequences as a whole instead of only adjusting single postures. This offers many benefits for factory planners. The user is able to modify a given motion sequence in a dynamic and intuitive way. In addition, our puppet has more degrees of freedom (DOF) than other comparable input devices and its overall shape and size is based on the anthropometric dimensions of the human body. As a result of our puppet's human-like appearance and its active feedback when performing a motion, highly realistic motions can be created. Our puppet is capable of both replaying existing motions on the puppet and recording the user's input. For this we developed algorithms for controlling and interacting with the device.

Structure of the System:
While developing the input device, the focus was on a simple and intuitive control and animation of virtual human models.

Analogous to a toy doll, the puppet's joints and limbs can be moved and positioned. The relation of size between body and limbs follows the anthropometric dimensions of the 50th percentile of the male residential population in Germany [1] and is uniformly scaled down to an appropriate size for an input device. This resulted in the puppet's total size of 70 centimeter. Based on the anthropometric data a CAD model of the puppet was created and its components were generated via 3D printing. A prototype of the puppet without external covering is shown in Fig. 1 (a). The puppet is separated into six limb groups, each containing the corresponding joints (Fig. 1 (b)).

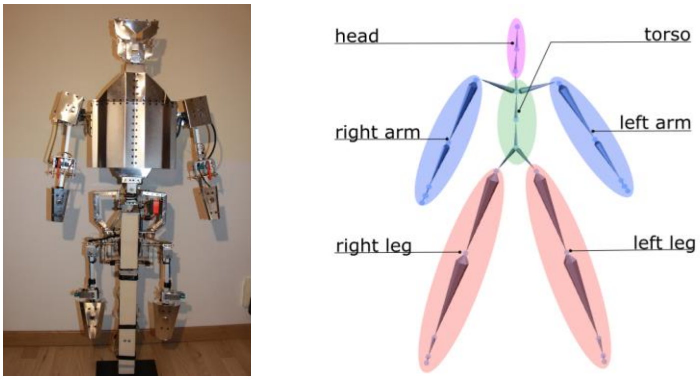
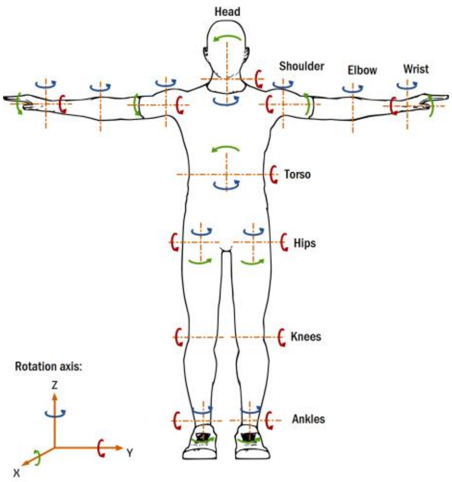


Fig. 1: Structure of the puppet: (a) Prototype, (b) Skeleton and the six limb groups.

Each group's movement can be individually activated and deactivated by the user. This can be done either using the software or by pressing the integrated interaction button on the limbs of the puppet. Deactivated limb groups will not be moved during the playback of motions.

As a result, a mechanical puppet, having 36 DOF, was built. Each DOF is controlled by a modified servo motor, which allows to trigger a desired joint angle as well as to read its current position. A schematic overview of the joints and their DOF is shown in Fig. 2.



Body Part	Joint	DOF (each)
Head	Neck	3
Arm	Shoulder	4
	Elbow	1
	Wrist	3
Torso	Waist	3
Leg	Hip	3
	Knee	1
	Ankles	2

Fig. 2: Integrated joints: (a) Kinematic scheme, (b) Number of DOF for each joint.

Each shoulder has four DOF – three for the rotation and one to pull the shoulder backwards and forwards. The range of motion, i.e. the joint angle limit, is an important factor in the assessment of human tasks during product and process planning. For example, when the engineer has to determine the capability of the worker to reach, grasp, or actuate various controls. In order to only permit realistic motions, end switches are integrated at the individual angle limit to guarantee that they cannot be exceeded. Those angle limits are based on the measured data of joint motion ranges of the Anthropology Research Project of NASA [3]. Another advantage of the integrated end switches is that an automatic calibration of the servo motors can be performed.

The puppet is attached to a stand at its lower back. This stand is positioned on a base box, which contains the power supply and a microcontroller. The controller offers two different ways to drive the puppet – via remote network control or via an integrated software. Playing back and requesting the joint angle data happens with 20 - 30 Hz, depending on the angular distance between two sequential postures.

<u>Software of the System:</u>
The software framework consists of two components: a GUI, which is installed on the controller, and a remote API. The remote API allows to control the puppet from another computer via network communication. The functions exposed by the API contain all features that are implemented in the framework. As a result, the user can configure the puppet, remotely load geometries and motions, start, pause or stop the motion playback. The motion sequence itself is sent via UDP protocol. Each datagram carries a single pose coded in ASCII format. The controller then loads the pose into the puppet device.

The controller on the other hand reads back the values of each servo motor and sends the current joint orientations back via UDP protocol, in a preselected frequency. Therewith the modification of motion sequences is possible.

The other component of the software framework, the GUI, monitors the current posture of the input device and visualizes it. The user is enabled to calibrate the puppet as well as to load, play back and save motions and ambient geometries. It is also possible to activate or deactivate the servo motors of each individual DOF. A screenshot of the visualization is shown in Fig. 3.

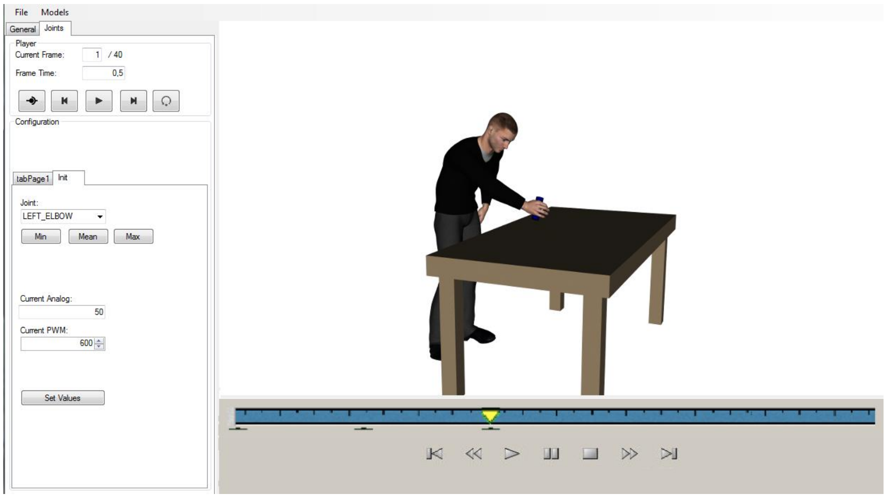


Fig. 3: Screenshot of the developed software.

When the user changes an angle joint of the puppet, the adjustment is transferred via a mapping function onto the virtual model. Beside the joint angle limits, the software also calculates the position of the joint angle hierarchy and stops the movement of single joints in order to avoid self-collision and collisions between the virtual model and the environment.

<u>Modification of Motion Sequences:</u>
Whenever the user wants to modify an animation, e.g. due to changes of the factory layout, he can start the playback within the software and the puppet begins to perform the motion sequence in a desired speed. In this way, the puppet is already positioned and oriented in the correct posture when the user decides to begin the modification at the favored frame of the sequence.

If single joints shall be modified, the joint or the corresponding limb group can be deactivated and adjusted during the playback. The system adopts the new position and aligns the motion accordingly. This process is iterative and can be repeated until the requested motion is generated. If this process leads to a collision with the environment, the software offers several options of how it should react. For an example, the system can reactivate the corresponding servo motors to solve this situation and avoid any collision.

The deactivation of joints and therewith the corresponding servo motors, leads to a basic problem, namely to maintain the posture against the gravitational force and the acceleration force acting through the movement of the other servo motors. If the servo motors are turned off, then each joint falls back in its rest position. For the purpose of holding the adjusted posture an integrated mechanical system individually aligns each joint's basic stiffness. Consequently a movement solely based on gravitational pull or on acceleration of the parent joint hierarchy is not possible.

A possible scenario is shown in Fig. 4, where a straight ahead movement is modified into a curve shape in order to reach another table.

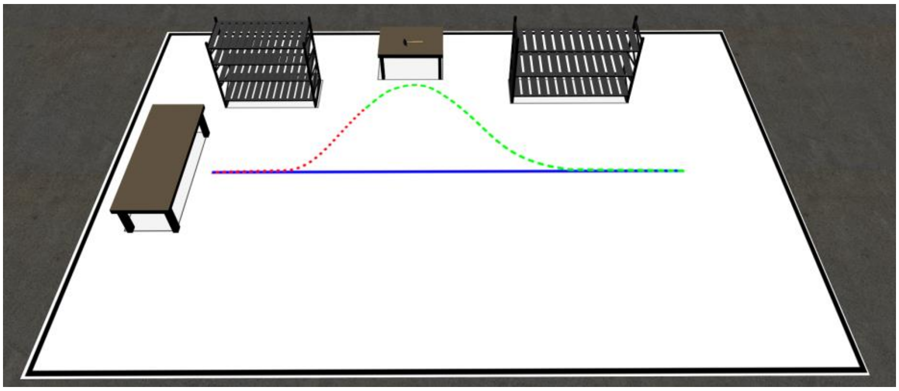


Fig. 4: Modification of a motion sequence: (blue) original motion sequence, (green) changed motion path, and (red) added frames at the end of the sequence.

Generating New Motions:

Although our electro-mechanical puppet is designed to allow an easy manipulation of virtual human motion, it might be necessary to build a new motion sequence from the ground up. For this purpose the user can either start from scratch or resort to a given dataset of basic motion and adjust the selected one. The database contains common motion snippets like walk, carry or grab.

In order to create new motions from scratch the user has to state their frame rate first. Then he begins to move one or more limb groups of the puppet by pressing the interaction button, which is integrated in every group. The software now receives the signal to record the movement of the corresponding limb hierarchy. The performed motion is transferred onto the virtual human model and visualized according to its context in time. As soon as the user releases the buttons, the recording is stopped and the current recording time is reset to the first frame. Now the user can continue with the recording of the next limb group. This can be repeated until every joint is defined. During this process, it is possible to play back the movement of already defined joints on the puppet as well as on the screen. This allows the user to better estimate the time progress of the motion.

Conclusion and Further Work:

The presented system, a human-like input device in combination with the software framework, addresses the needs of motion modification in advanced process planning. The API provides the ability to integrate the input device into existing planning software.

Its intuitive handling enables the creation and manipulation of motions even by untrained users without prior knowledge and experience in animation editing. The motion performing puppet is giving them a fast visual and haptic feedback of their changes, which are mapped onto a virtual human model in real-time. With its low expenditure of time and money compared to other solutions, it is applicable even to small and medium-sized enterprises.

The next step in enhancing this system is to evaluate the puppet regarding its robustness and accuracy. In addition, we are going to extend the contained motion database. As part of this extension of the database, we plan to development methods for using the input puppet as a tool to perform queries for motion data.

References:

[1] German Institute for Standardization: DIN 33402-2, Berlin, 2005. https://www.beuth.de/en/standard/din-33402-2/84092742.

[2] Glauser, O.; Ma, W.-C.; Panozzo, D.; Jacobson, A.; Hilliges, O.; Sorkine-Hornung, O.: Rig Animation with a Tangible and Modular Input Device, ACM Transactions on Graphics, 35(4), 2016, 1–11. https://doi.org/10.1145/2897824.2925909

[3] NASA Scientific and Technical Information Office: Anthropometric Source Book Volume I: Anthropometry for Designers, NASA, Yellow Springs, Ohio, 1978. www.dtic.mil/dtic/tr/fulltext/u2/a473151.pdf.

[4] Sundin, A.; Örtengren, R.: Digital Human Modeling for CAE Applications, G. Salvendy (Ed.), Handbook of Human Factors and Ergonomics, 3rd ed., John Wiley & Sons, Inc., Hoboken, NJ, USA, 2006, 1053–1078. https://doi.org/10.1002/0470048204.ch39

[5] Wang, X.; Chen, Q.; Wang, W.: 3D Human Motion Editing and Synthesis: A Survey, Computational and Mathematical Methods in Medicine, 2014, 1–11. https://doi.org/10.1155/2014/104535

[6] Wiegmann, D.; Brüggemann, H.; Rausch, A.: Design and Evaluation of a Human-Like Puppet as an Input Device for Ergonomic Simulation, V.G. Duffy (Ed.), Digital Human Modeling. Applications in Health, Safety, Ergonomics, and Risk Management: Ergonomics and Design. DHM 2017. Lecture Notes in Computer Science, Springer International Publishing AG 2017, Cham, Switzerland, 2017, 393–403. https://doi.org/10.1007/978-3-319-58463-8_33

[7] Yoshizaki, W.; Mochimaru, M.; Sugiura, Y.; Chiou, A.C.; Hashimoto, S.; Inami, M.; Igarashi, T.; Akazawa, Y.; Kawachi, K.; Kagami, S.: An actuated physical puppet as an input device for controlling a digital manikin, Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems - CHI ’11, ACM Press, 2011, 637–646. https://doi.org/10.1145/1978942.1979034