Continuum of Virtual-Human Space:
Towards Improved Interaction Strategies for Physical-Virtual Avatars

Arjun Nagendran[*]
Institute for Simulation and Training
University of Central Florida

Remo Pillat[†]
Division of Computer Science
University of Central Florida

Charles Hughes[‡]
Division of Computer Science
University of Central Florida

Greg Welch[§]
Institute for Simulation and Training
University of Central Florida

Abstract

In this work, a broad continuum of 3D space that encapsulates *avatars*, ranging from artificial to real in shape, appearance and intelligence is defined. The research focuses on the control of *physical-virtual avatars* that occupy a specific region in this space that may be suitable for interacting with elements in the environment. To facilitate this control, a paradigm called microposes is developed that overcomes the need for high network bandwidth during remote tele-operation of avatars. The avatar itself uses a control strategy that interprets the received microposes data and executes motions that appear natural and human-like in the presence of data loss and noise. The physical-virtual avatar is used in several training and learning scenarios. Results during testing reveal a reduced bandwidth requirement during remote tele-operation of physical virtual avatars and a good motion tracking performance with respect to a commanded pose from the inhabiter.

CR Categories: I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence—Intelligent Agents I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality

Keywords: Avatars, Physical-Virtual Avatar, Microposes, Remote-Operation

1 Introduction

A *virtual avatar* is described as a perceptible digital representation whose behaviors reflect those executed, typically in real time, by a specific human being [Bailenson and Blascovich 2004]. As an extension to this definition, a *physical-virtual avatar* is described herein as an entity that facilitates remote presence with an intention to interact with elements in the environment. Robotic entities can be used to exhibit a physical presence in such remote environments and provide the infrastructure needed to accomplish human-like tasks such as gesturing during conversations or more complex actions such as picking up and moving objects. In this context, persons who ‘drive’ their remote robotic counterpart are referred to as *inhabiters*. In order to achieve this, data that conveys the necessary action to be performed by the robotic entity at the remote site must be transmitted over a network. Limitations in bandwidth and network latency are challenges that must be overcome in order to facilitate this transmission efficiently. Additionally, the robotic avatar must be capable of interpreting this received data and execute a planned motion that captures the intent of an *inhabiter*. An important goal is that the generated movements should appear human-like, even if the input data includes artifacts and noise. In contrast to tele-robotics, the control paradigm for human-avatar control in this work is less tightly coupled, with the avatar having the flexibility to exhibit independent autonomous behaviors in situations where there may be noisy data or packet loss .

1.1 The Continuum of Virtual-Human Space and Physical Virtual Avatars

The notion of a virtual human has been explored for many years, in computer games, Virtual Reality, and beyond. Today, applications of virtual humans include telepresence, military and medical training, education, and healthcare, the notion of which has been explored for years, for example by Tachi et al. [Tachi 2009]. The idea of using a real, physical tele-robotic surrogate for viewing or manipulation in remote locations has been discussed for decades, and several studies have explored human interaction with a physical robot vs. a life-sized virtual robot [Bainbridge et al. 2008; Kidd and Breazeal 2008; Powers et al. 2007]. Investigations have consistently found that physical robots are more enjoyable, engaging, and effective. Advances include efforts to develop human - inspired [Coradeschi et al. 2006] or human-like animatronic robots that in some ways resemble or imitate the appearance and behavior of specific humans or living creatures [Breazeal and Scassellati 2002; Epstein 2006].

Such animatronic (physical) humans, though very compelling, will always be constrained to look like one particular human: one skin tone, one gender, one shape - one person. This is in contrast to purely virtual humans, which, among other possibilities, have complete flexibility to change shape, skin tone, and gender.

While there exist many such examples of specific manifestations of “virtual humans” we are not aware of any attempts to define a comprehensive space of characteristics, much less any efforts to explore such a space to develop some understanding for which characteristics matter for certain conditions and tasks. For example, the characteristics of virtual humans might be described by a 3D space comprising the artificial-real dimensions of virtual human appearance, intelligence, and shape as depicted in Figure 1. Virtual avatars (flat screen display) for instance could be made to appear like a particular human, exhibit artificial intelligence, but have no real shape (i.e.

---

[*] Synthetic Reality Lab, e-mail:arjun@cs.ucf.edu
[†] Synthetic Reality Lab, e-mail:rpillat@cs.ucf.edu
[‡] Synthetic Reality Lab, e-mail:ceh@cs.ucf.edu
[§] Synthetic Reality Lab, e-mail:gwelch@ist.ucf.edu

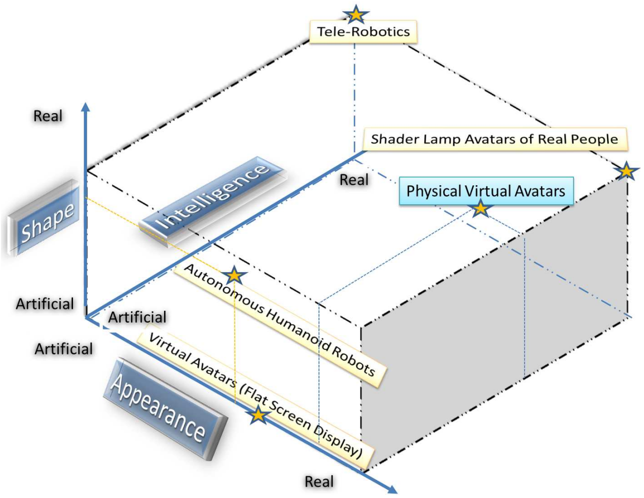


Figure 1: The 3D Continuum Space in which all virtual-human systems reside

physical manifestation) associated with them. A typical example could be a football player in a computer game. Note that the intelligence of this avatar can tend towards the real when controlled by a real human playing the game. Similarly, the appearance can tend towards artificial if a human-player customizes his avatar to look like someone else. Autonomous humanoid robots can be made to look similar to humans both in appearance and shape (depending on their degrees of freedom), but exhibit artificial intelligence. Tele-robotics on the other hand occupies one specific corner of the 3D space, since it is generally associated with human control - i.e. real intelligence. At the opposite corner lie Shader-lamp avatars [Lincoln et al. 2011] of real people since it is essentially tele-robotics combined with real appearance.

If one could combine physical presence and dynamic virtual appearance, one could transmit important non-verbal information typically conveyed in human-to-human interactions. This is the concept of *physical virtual avatars* as indicated in the 3D space shown in Figure 1. Examples include conversations between teachers and students, or physicians and patients. A combination of eye contact, body proximity, forward leaning, smiling, and touch can convey intimacy and trust; while averted gaze, distance, and the absence of smiling and touch can indicate detachment [Heath and Bryant 2000]. The next subsection describes the contributions of research reported here towards achieving realistic and functional *physical-virtual avatars*.

1.2 Problem Description

Robotic avatars enable physical tele-presence in remote locations, facilitating interactions with the remote environment and helping the *inhabiter* to accomplish situation-specific tasks. The functionality and effectiveness of these avatars are greatly influenced by the available network bandwidth, since data needs to be transmitted and received from a remote location for full functionality. Additionally, the robotic entity must exhibit human-like dexterity and compliance in its movements to appear natural during interaction despite the presence of noise and packet loss. The goals of the work presented here are therefore to

- minimize the amount of traffic (data) transmitted over the network, while maximizing the human-likeness of the avatar,
- design the avatar's actuation to match the kinematic and dynamic limits of human motion

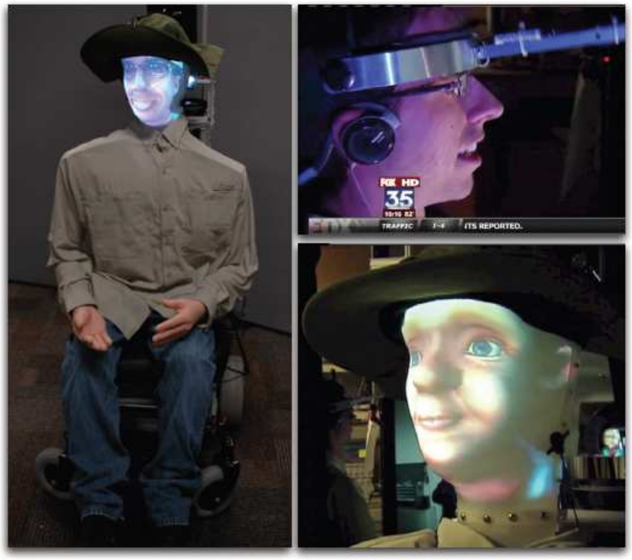


Figure 2: *Left*: Current PVA prototype, with a real human agent and appearance. *Top Right*: Real human appearance obtained from a head-worn camera. *Bottom Right*: Real human agent but synthetic appearance from the teacher training system.

The contributions of the work in order to achieve these goals are summarized below:

- the development of a concept called *microposes* that reduces network traffic while containing sufficient information to capture the intent of an inhabiter at the remote location
- the development of control strategies that interpret the received data to create realistic and smooth movements of the robotic avatar in the remote location

An example implementation of these contributions in a physical-virtual avatar (PVA) system is presented.

2 Modes of Operation

This paper focuses on the space in which a *physical-virtual avatar* can be pervasive along the following two dimensions: intelligence and appearance. Each of these dimensions is associated with a continuum, rather than a set of discrete states. Shape, although important, is not a focus of this paper since the presented work is applicable anywhere along the shape axis, provided the challenges associated with projection and rendering on varied surfaces are addressed.

Intelligence, as depicted in Figure 1 ranges from artificial (agent-based) to real (human controlled). An agent-based avatar is computer-driven, exhibiting a certain degree of intelligence that adheres to a rule-based system. Real avatars, on the other hand, can be controlled either partly or fully by a human in the loop.

Appearance of avatars ranges from purely artificial (virtual character) to purely real (real-time video capture of inhabiter), and is dependent on how accurately a human *inhabiter*'s appearance must be portrayed on the avatar.

Given the space in which *physical-virtual-avatars* exist, we focus on two specific instances that are differentiated primarily by appearance and shape. Both these instances exhibit a level of intelligence that lies between the real and the virtual, but may vary to a certain degree. The first instance is called a synthetic avatar and has artificial appearance and can exist anywhere in the continuum of the space axis i.e. ranging from flat screen displays (2D), to artificial 3D (stereo), to a robotic avatar (physical 3D). The second instance is referred to herein as a real avatar, whose appearance and shape closely matches the inhabiter, while intelligence lies between the artificial and the real.

2.1 The Synthetic Avatar

A synthetic avatar is of particular interest when the anonymity of the *inhabiter* needs to be maintained. This could involve their use in scenarios where a trainee is expected to interact with a specific cultural / professional representation of the avatar (trainer) in order to eliminate any bias that may be triggered as a result of appearance. The avatar may also be used by a person who has a medical condition or does not want to be identified for some reason, but still wishes to interact with people in a social setting. The appearance of the avatar can be altered synthetically by creating characters as required by a scenario. In the absence of human control, the characters are 'intelligent' to a certain degree, executing planned motions and responding to stimuli in the presence of a sensing system. Figure 4 shows one such synthetic avatar (manifesting both in 2D (c) as well as a robotic avatar (a, b)) being used in a classroom teaching scenario. The avatar is a representation of a middle school student and can be controlled by an interactor to help teachers hone their skills in the classroom. In the absence of a human controlling it, the avatar exhibits simple behaviors such as tracking the movements of the teacher in the classroom (and orienting its gaze towards them), showing a lack of interest, and engaging in conversation with other similar avatars. When an *inhabiter* takes control of the avatar, its movements and actions are driven by a low-network traffic paradigm called micro-poses using commodity game controllers such as the Kinect and a Razor Hydra, which is detailed in Section 3.1.

2.2 The Real Avatar

The use of a real avatar is better suited to situations that demand an actual physical presence of a specific person. In meetings, conferences, social gatherings, for instance, an identity for an avatar can be established through closely matching its appearance and actions to the person who is controlling it. It is expected that these avatars are initially capable of accomplishing simple tasks such as gestures, hand-shakes and movement within the remote environment. Eventually, more complicated tasks such as being able to pick up and move objects, and open doors could be performed by the avatar. This is what makes their shape important. To achieve the desired functionality, the actions of an *inhabiter* must be captured appropriately and transmitted to the avatar, where a control system processes this data to recreate smooth and natural looking motion in the presence of transmission delays and data loss. Section 4 describes one such system that is currently used for the head motion of an avatar, but can be extended to full-body motion to support a greater range of gestures.

3 Capturing and Transmitting an Inhabiter's Intent

Controlling an avatar remotely (whether synthetic, or real) requires efficient transmission of data from the *inhabiter* station to the physical avatar system while adhering to the available network bandwidth.

Traditional methods of avatar control involve some form of motion capture (e.g. infra-red cameras and motion capture suits) at the *inhabiter*'s end, and sending the joint-space coordinates to the remote site [Moeslund et al. 2006; Holte et al. 2012]. The system driving the avatar attempts to achieve these joint-space coordinates in the local reference frame of the avatar, thereby mimicking the actions of the *inhabiter*. One of the major bottlenecks in such a system is the amount of data transmitted over the network. This problem grows with an increase in the number of degrees of freedom - a requirement to ensure smooth and realistic natural motion of the avatar.

A second challenge with such systems is the ability to cope with loss of data during capture. Data for a particular joint may be lost due to occlusion at an *inhabiter*'s end [Lou and Chai 2010]. This discontinuity in data can result in rather quick and jerky motions at the avatar's end. While interpolation between discontinuities can be performed locally by the avatar, there is the risk of aliasing and also executing motions that were not necessarily performed or intended by an *inhabiter*.

From a technical vantage point, there is the need to reliably and efficiently capture and transmit behaviors and actions of an *inhabiter* over a long-distance network; the necessity to achieve this transmission even in the presence of limited resources like bandwidth and computational power; and the requirement to articulate these actions at the client site, providing smooth interactive performance. The presented system employs a communication protocol that results in low network bandwidth requirements, supporting the transmission of complex actions and behaviors between geographically remote sites. This benefit is achieved by employing a gesture-based paradigm that depends on blending of "micro-poses" to remotely control an avatar. The paradigm supports culturally appropriate gestures and significantly reduces the streaming transmission bandwidth requirements over long-distance networks. The reduced amount of information transferred from the inhabiter's end now requires an increased amount of intelligence at the avatar's end in order to interpret this information and effectively enable interaction. Referring back to Figure 1, it follows that this paradigm forces the *physical virtual avatar* to occupy a specific coordinate along the intelligence axis (whether synthetic or real), while being variable along the shape and appearance axes.

3.1 Gestures - Microposes

The avatar control system that is presented here evolved from a very literal system based on motion-capture to a gestural one based on a marionette paradigm [Mapes et al. 2011]. For the initial motion capture-based system, the inhabiter wore a vest and a baseball cap. The vest had retro-reflective markers on wrists, elbows and shoulders. The baseball cap had an easily recognized pattern of three retro-reflective markers in order to capture orientation as well as position of the head. This required nine 3d coordinates to be transmitted over the network. The problem with this approach was noise and undesirable data bursts. Noise is common in motion capture systems, and is most often managed by post-production (off-line, after the fact) clean-up of bad data [Lou and Chai 2010]. Since post-production is not an option in the real-time control of an avatar, the system was prone to significant noise. Additionally, sending this amount of data, resulted in 40kbps of network traffic i.e. ((9 articulation points x four floats + five floats for pose weights) at a rate of 30 frames per second).

To address the problems introduced above, a number of variants of the paradigm were developed, investigating each one in the context of its effect on noise, network traffic, the quality of the experience at the receiver end, and the cognitive and physical demands reported by inhabiters. The first and, we feel, most critical decision was to

Figure 3: Microposes for a virtual avatar named ‘Sean’ **(a)** shows Sean standing (translucent) and holding a pen (solid) **(b)** shows Sean leaning forward and turning (translucent) and slouching (solid) **(c)** depicts Sean laying on the desk (translucent) and raising his hand (solid)

develop the notion of micro-poses. These are a set of key poses that, in effect, form a basis set for the poses that an avatar is expected to perform, between which blending can occur. Some of these microposes are shown super-imposed on each other to view the ’motion-space’ of the avatar in Figure 3.

In order to trigger the microposes without using motion capture, the inhabiter was presented with a pose selector that looked like a pawn with a retro-reflective ball on its top. As the inhabiter moved the selector in 3D space, an infrared camera-based set-up captured the selector’s position. Avatar control was then done by moving the pawn close to the desk on which a template of the avatar and its associated behavior was placed. Once a selection was made by tapping on the template, the inhabiter raised the pawn which entered a virtual sphere that was populated with micro-poses. As one moved the pawn in this sphere, the system found the closest set of poses for weighted blending and then started a process of decaying the weights of current poses, while increasing those of the newly selected ones. The software at the avatar’s end then used the most recent pose and the new weighted pose to do a natural blend, with old pose weights damping out and new pose weights rising. Additionally, the inhabiter’s head was tracked to allow more precise control of the avatar’s head. The result was a low noise system that sent very small amounts of data for motions (typically just 10 bytes for the indices and weights of the current pose). Other data that was transmitted included the head orientation (15 bytes * 30fps = 3.6kbps). This control paradigm was successfully used in cross cultural training scenarios and an experiment with a middle school that involved forty-three middle school-aged girls practicing resist/avoid strategies with regard to peer pressure.

The pose blending system described focused on using a pose selector device as controller, but the interface is not very intuitive for inhabiters who want more natural controls. To achieve this, the Microsoft Kinect was integrated to capture upper body movement, while still using the infrared tracking cameras for head orientation. Upper body positions were measured relative to the micro-poses and used to select the two closest for blending, as was done with the pawn. The resulting variant used the same data packets as discussed above.

While the above two approaches reduce network demands and essentially eliminate noise, each has their problems. As noted above, the inhabiters were somewhat uncomfortable with the non-intuitive nature of the pawn, since this required remembering where the poses resided in an imaginary sphere. The Kinect circumvented this issue, but its full upper body control paradigm left little opportunity to trigger poses such as lying down on a desk, without the inhabiter having to physically perform this action. These limitations were overcome with the release of the Razer Hydra magnetically tracked controller.

The Hydra uses a magnetic field to detect absolute position and orientation of two handheld controllers. So long as the controllers are in front of the magnetic sensor and within a six-foot radius, the device operates with a reasonably accurate precision of one millimeter and one degree. Each controller has five digital buttons, one analog stick/button, one bumper button and one analog trigger. The buttons can be programmed to trigger situation-specific reactions and appearance-related features of an avatar such as smiles, winks, and frowns. The right controller is used for other avatar behaviors that are classified into gestures for the purposes of intuitive control. As with the simple pawn scheme and the Kinect-based control, a library of micro-poses unique to each virtual avatar is employed. Control of the avatar’s pose is done by gestures that are mapped to micro-poses, with variations in those gestures coming from being close to several poses and by twisting the controllers to get subtle deviations, e.g., hand shaking or pen clicking.

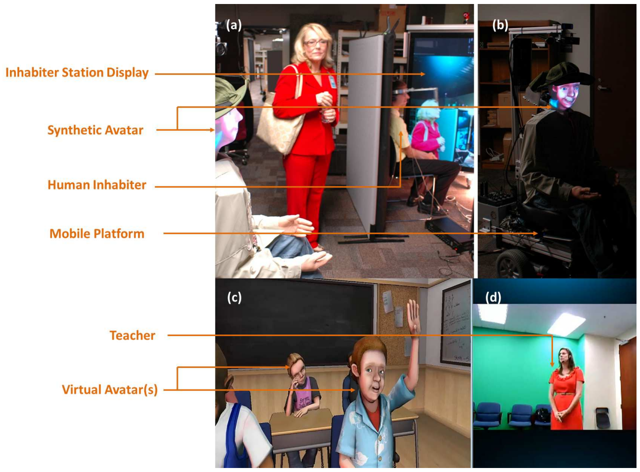


Figure 4: (a) shows a synthetic character manifested physically being driven by a human inhabiter. Also seen is the inhabiter station which allows a human who is driving the avatar to view and interact with his environment in real time. (b) shows the physical manifestation (robotic) of the synthetic avatar on a mobile platform (wheelchair) to allow motion within the environment. (c) shows a small class of virtual middle school children. (d) shows a teacher addressing this class. Notice that Sean has his arm raised in a pose very similar but not precisely the same as his corresponding micropose in Figure 3.

4 Realizing an Inhabiter’s Motion: The Robotic Avatar Control Paradigm

In section 3.1, an instance of using the microposes to control a synthetic virtual avatar in a classroom teaching scenarios was described. This system was extended to a real (robotic) avatar which could be used as a physical-virtual entity in the same classroom teaching scenario. In order to realize this system, the microposes received must be interpreted to create physically believable motion at the avatar’s end. The real (robotic) avatar (Figure 4(a, b)) is driven by a set of rotary actuator modules, with integrated sensors in each of the modules providing the positional and torque measurements required to achieve a tightly coupled closed-loop control paradigm. The human-like kinematic configuration of the robotic avatar results in a non-linear coupling between the joint space variables - these are decoupled using the well known Recursive Newton Euler (RNE) inverse dynamics [Featherstone and Orin 2000]. The effect is more pronounced with higher inertial loading of the structural components that make up the avatar and also during execution of higher velocity motion profiles. Microposes (as described in Section 3.1) specify a new target position for each joint in the kinematic chain of the robot. Achieving this target position requires a blending between the current and the new pose which in turn requires planning and executing a trajectory between two new joint-space coordinates so that it appears natural. An inverse kinematic solution first determines the joint-space coordinates at every instant in time to achieve the motion required by the avatar to represent the *inhabiter*. Computed torque control [Rhody and Heppler 1993; Mahanta and Bhagat 2006] is then used for open-loop trajectory control of the end effector, but is highly dependent on the inertial model of the arm. Therefore, it is coupled with a proportional - integral - derivative (PID) controller to perform trajectory tracking and ensure disturbance rejection during the movement [Lambrechts et al. 2006].

In terms of the actual implementation, the robot can be thought of as an inertial mechanism driven by actuators with controllable torques so that the dynamic equation now becomes

$$\tau = I(\theta)\ddot{\theta} + c(\dot{\theta}, \theta)\dot{\theta} + g(\theta) \qquad (1)$$

where τ is the vector of joint torques, θ is the n-dimensional joint-space position vector, $I(\theta)$ is an nxn symmetric inertia matrix, $c(\dot{\theta}, \theta)$ is the nxn tensor or velocity coupling matrix, where n is the number of joints, and g is the $nx1$ gravity-dependent vector.

For a physical robotic avatar, it is important to keep the motion natural looking during the interaction. With virtual characters, a reset of joint space variables can occur at anytime, in which case a transition from the current state space to the initial state space occurs in a single frame. This is barely noticeable due to the small duration over which the transition occurs. However, with a physical robot, such a transition is largely evident since a motion profile has to be executed to achieve the new state space.

A large change between the current and the new state space can result in rapid motion (high velocities), causing the effect to propagate along the kinematic chain due to the non-linear coupling described previously. The parameters of the controller must be tuned to be able to cope with such artifacts that can occur as a result of latency / loss in data packets over the network.

The microposes themselves cannot guarantee these joint motion limits. As a result, this is handled at the remote robotic client end using constraints that conform to a normal 'accepted' range of human hand motion. Actuation of the robotic avatar's head that demonstrates this smooth control paradigm is presented in the following section.

4.1 An Explicit Example - Schunk Head Control

The robotic avatar is equipped with a Schunk PW 70 pan-tilt unit that carries a rear-projected vacuform head as well as a projector assembly. Communication with the Schunk unit occurs via the message-oriented CAN bus. The goal of the proposed control scheme is to enable the avatar to directly mimic the head motion of the *inhabiter* during interaction while faithfully ensuring smooth, human-like motions. An OptiTrack IR System is used to capture the *inhabiter*'s head motions and transmitted to the Schunk which responds by minimizing the error between its current orientation and the received orientation in minimum time. The IR head tracking data is prone to noise resulting in fairly jerky movement at the avatar's end due to discontinuities in velocities. In addition, orientation targets are sometimes overshot, because the unit is not able to decelerate quickly enough. The only adjustable parameters in this mode are the maximum movement velocity and acceleration. Furthermore, the Schunk pan-tilt unit uses an internal brake to hold the head in a stable position after the full execution of a move, sometimes leading to delays in the startup of new motions.

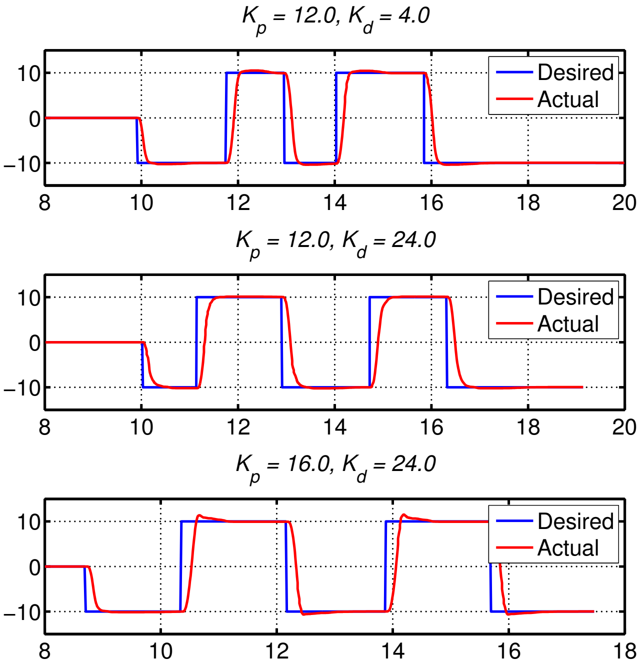


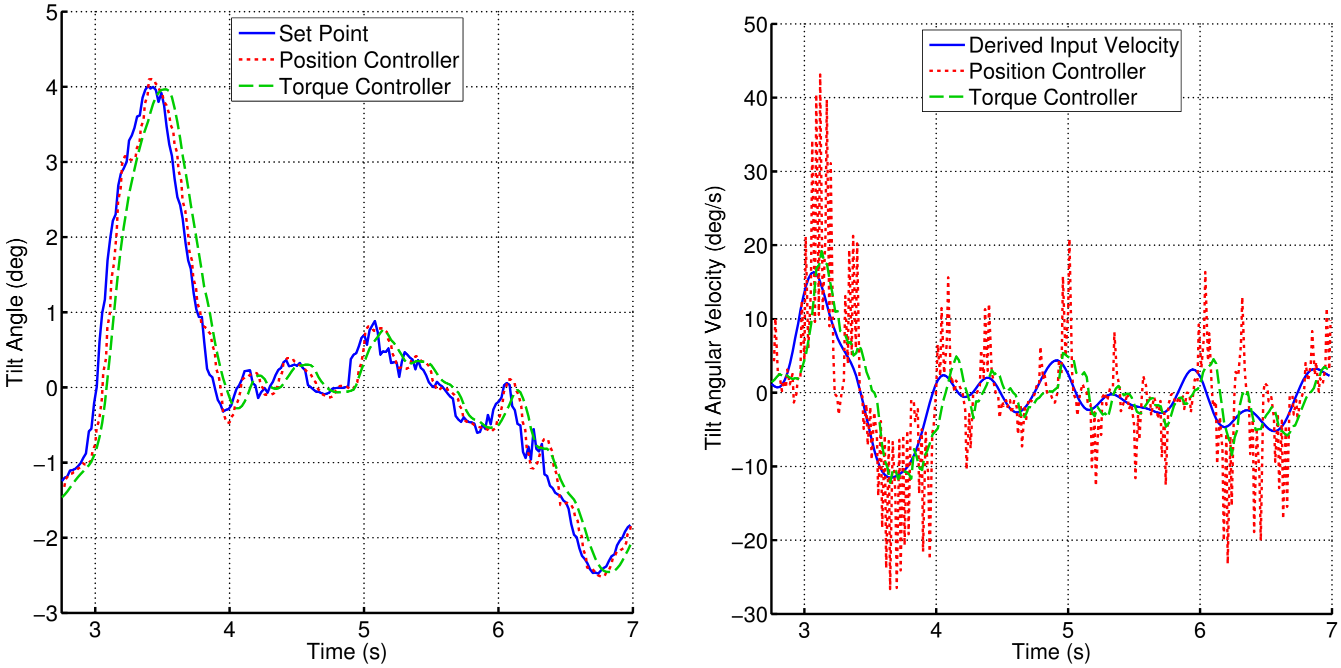
Figure 5: Response of the pan axis torque controller for different parameters. The x-axes show time (in s) and the y-axes show angle set points (in deg)

To achieve smoother and more responsive behavior, control of the Schunk is implemented by a regulation of the amount of current drawn, and hence is representative of torque control. In this mode, the Schunk unit receives target orientations at $100\ Hz$, but the achievement of these targets is validated through an external control system. The time-critical execution of the position control loop is handled in a separate thread, thereby providing fine-grained management of the final motion trajectories.

The desired tilt θ_{pd} and desired pan angles θ_{yd} are captured based on the *inhabiter*'s head movement and transmitted at a high frequency to the avatar. The difference of these desired angles to the actual positions measured by the pan-tilt unit (θ_{pa} and θ_{ya}) constitutes the error input e to the PID controller. Based on its proportional (K_p), integral (K_i), and derivative (K_d) parameters, appropriate torques are calculated by the controller and transmitted to the pan-tilt unit. It should be noted that there is an independent current/torque controller for each movement axis.

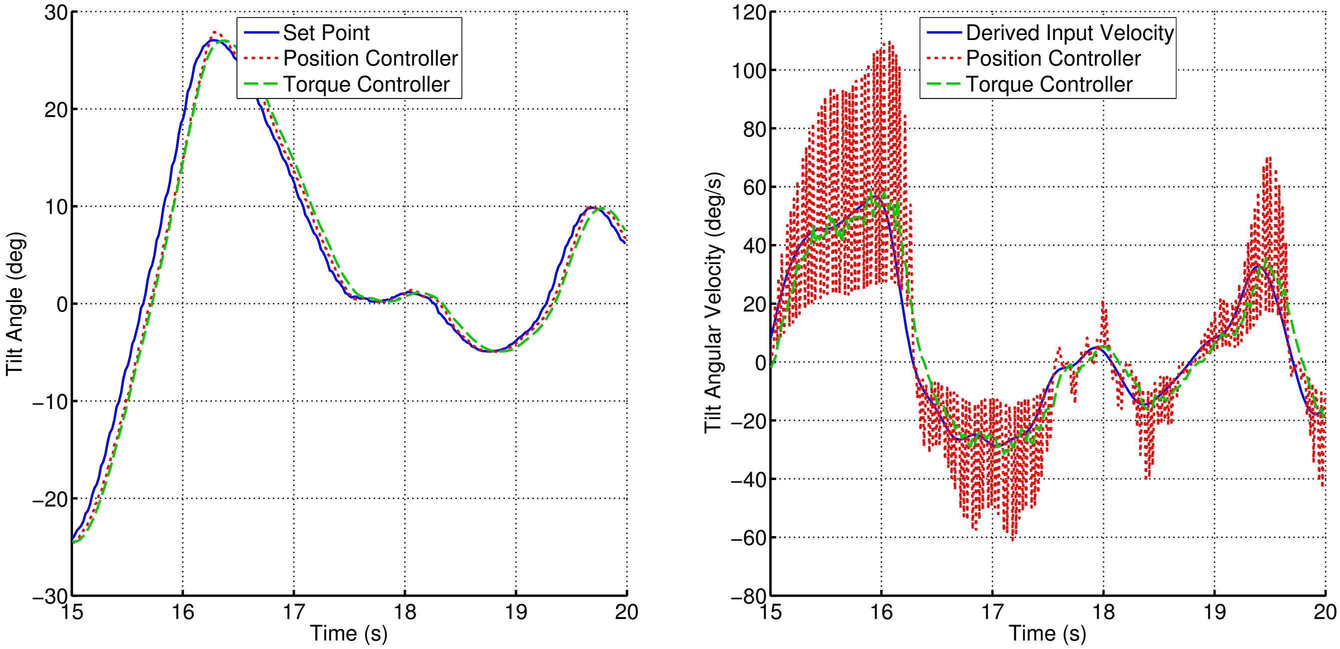
4.2 Discussion of Controller and Tracking Performance

The control system parameters were tuned by observing the system's characteristics in response to a step position demand. Figure 5 shows some of the tests performed for tuning the PID controller on the pan axis. A satisfactory set of parameters provides a compromise between signal rise time, overshoot behavior, and settling time. It was found that each movement axis required a set of different tuning gains, since the apparent inertia of the pan-tilt unit varies due to the tilt-axis being subject to the forces of gravity along its direction of motion. Standard tuning rules such as Ziegler-Nichols and Pessen-Integral [Pessen 1994] were used as a basis to settle on appropriate gains that provided the desired response with-



(a) Detail of tilt angle tracking performance of the two controllers (b/w 3s and 7s of motion)

(b) Corresponding angular velocity for profile shown in (a). The velocity magnitudes are relatively small



(c) Detail of tilt angle tracking performance of the two controllers (b/w 15s and 20s of motion)

(d) Corresponding angular velocity for profile shown in (c). The velocity magnitudes are comparatively large

Figure 6: Performance comparison of the position controller and the proposed torque controller. For clarity, data is only shown for the tilt axis. Please see the discussion in Section 4.1.

out exhibiting overshoot or oscillations.

A good balance of signal overshoot and convergence speed was found with the parameters ($K_p = 12, K_d = 24, K_i = 0$) for the pan axis and ($K_p = 12, K_d = K_i = 0$) for the tilt axis.

To compare the performance of the proposed torque controller to the commonly implemented position controller, we recorded a data set of random human head motions. Without any additional processing, the controllers should be able to faithfully replicate the desired angular position profile, while avoiding jerky, discontinuous, and visually distracting velocities.

The position controller implementation plans a trapezoidal velocity profile to move from the actual to the desired position. New position demands will trigger an automatic re-planning based on the current position, velocity, and acceleration. The implementation used for these tests employs a maximum velocity of $200^\circ/s$ and a maximum acceleration of $400^\circ/s^2$ for both axes.

Figures 6(a) and 6(c) show the angular position tracking performance of both controllers for two different subsets of the recorded data. Both the position as well as the torque controller track the set point angle very well, but the position controller converges noticeably faster on set point changes (about $50\ ms$ versus about $100\ ms$ for the torque controller). On the flip side, the position controller slightly overshoots its target angles, most notably at time instants $3.44\ s$ and $16.33\ s$. This overshoot is not present in the torque controller output.

Apart from the position performance, it is essential that the controller exhibits human-like smoothness in its velocity profile. Figures 6(b) and 6(d) show the derived angular velocities corresponding to the angle profiles in Figures 6(a) and 6(c), respectively. It should be noted that the angular velocities are derived through differentiation of the input signal (human head movement) and the output signals that the controllers produce. Although both controllers exhibit some level of oscillation in their derived output velocity, the torque controller variations occur at a significantly lower frequency and magnitude. The superior performance of the torque controller is especially apparent in Figure 6(d). Here, rapid changes in demand velocity lead to uncontrolled oscillations in the position controller response, while the torque controller shows the same magnitude of low-frequency variations that were apparent at the lower movement speeds in Figure 6(b). The high-frequency noise in the velocity signal of the position controller is undesired and will invariably lead to a degraded visual appearance of the executed motion.

5 Conclusion and Future Work

The defined 3D continuum of space based on appearance, shape and intelligence is sufficient to depict any system which is representative of physical presence in an environment. This may involve purely virtual avatars to purely real avatars and any combination in between obtained by varying the level of shape, appearance or intelligence. The control of any avatar in this space requires a method to capture an inhabiter's intent and transmit this data to the avatar for execution. A method to accomplish this, while keeping the network latency and bandwidth requirements to a minimum, is described using the concept of microposes. Control strategies that interpret the received microposes data and execute motions that depict the inhabiter's intent are discussed, with specific results on head-movement showing an adherence to normal human-motion ranges and likeness. Future work involves extending the proposed paradigm to fully mobile real avatars, and including the ability to dynamically interact with the environment (such as picking up and moving objects, opening doors etc.).

6 Acknowledgments

The material presented in this publication is based on work supported by the Office of Naval Research Code 30 (N000141210052), National Science Foundation (CNS1051067) and the Bill & Melinda Gates Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Office of Naval Research, National Science Foundation or the Gates Foundation.

We would like to thank Prof. Gerald Seet (Nanyang technological University) and Prof. Henry Fuchs (The University of North Carolina, Chapel Hill) for their valuable guidance and insight throughout the project. We also acknowledge Peter Lincoln, Ryan Schubert, Adrian Ilie (The University of North Carolina, Chapel Hill) and Dan Mapes (University of Central Florida) for their valuable contributions to the code base.

References

BAILENSON, J. N., AND BLASCOVICH, J. J. 2004. Avatars. In Encyclopedia of Human-Computer Interaction, W. S. Bainbridge, Ed., 1st ed. Berkshire Publishing Group, 64–68.

BAINBRIDGE, W., HART, J., KIM, E., AND SCASSELLATI, B. 2008. The Effect of Presence on Human-Robot Interaction. In IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 701–706.

BREAZEAL, C., AND SCASSELLATI, B. 2002. Robots that imitate humans. Trends in Cognitive Sciences 6, 11 (Nov.), 481–487.

CORADESCHI, S., ISHIGURO, H., ASADA, M., SHAPIRO, S. C., THIELSCHER, M., BREAZEAL, C., MATARIC, M. J., AND ISHIDA, H. 2006. Human-Inspired Robots. IEEE Intelligent Systems 21, 4, 74–85.

EPSTEIN, R. 2006. My Date with a Robot. Scientific American Mind June/July, 68–73.

FEATHERSTONE, R., AND ORIN, D. 2000. Robot Dynamics: Equations and Algorithms. In IEEE International Conference on Robotics and Automation (ICRA), IEEE, vol. 1, 826–834.

HEATH, R., AND BRYANT, J. 2000. Human Communication Theory and Research. LEA's Communication Series. L. Erlbaum.

HOLTE, M. B., TRAN, C., TRIVEDI, M. M., AND MOESLUND, T. B. 2012. Human Pose Estimation and Activity Recognition From Multi-View Videos: Comparative Explorations of Recent Developments. Journal of Selected Topics in Signal Processing 6, 5, 538–552.

KIDD, C., AND BREAZEAL, C. 2008. Robots at Home: Understanding Long-Term Human-Robot Interaction. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 3230–3235.

LAMBRECHTS, P., BOERLAGE, M., AND STEINBUCH, M. 2006. Trajectory Planning and Feedforward Design for Electromechanical Motion Systems. Control Engineering Practice 13, 2 (Feb.), 145–157.

LINCOLN, P., WELCH, G., NASHEL, A., STATE, A., ILIE, A., AND FUCHS, H. 2011. Animatronic Shader Lamps Avatars. Virtual Reality 15, 2-3, 225–238.

LOU, H., AND CHAI, J. 2010. Example-Based Human Motion Denoising. IEEE Transactions on Visualization and Computer Graphics 16, 5, 870–879.

MAHANTA, C., AND BHAGAT, P. 2006. Adaptive Fuzzy Self-learning Controller for Robotic Manipulators. In IEEE Conference on Robotics, Automation and Mechatronics, Ieee, 1–6.

MAPES, D. P., TONNER, P., AND HUGHES, C. E. 2011. Geppetto: An Environment for the Efficient Control and Transmission of Digital Puppetry. In International Conference on Virtual and Mixed Reality: Systems and Applications, Springer-Verlag, Berlin, Heidelberg, 270–278.

MOESLUND, T. B., HILTON, A., AND KRÜGER, V. 2006. A Survey of Advances in Vision-Based Human Motion Capture and Analysis. Computer Vision and Image Understanding 104, 2-3 (Nov.), 90–126.

PESSEN, D. W. 1994. A New Look at PID-Controller Tuning. Journal of Dynamic Systems, Measurement, and Control 116, 3, 553–557.

POWERS, A., KIESLER, S., FUSSELL, S., AND TORREY, C. 2007. Comparing a computer agent with a humanoid robot. In ACM/IEEE international Conference on Human-Robot Interaction (HRI), ACM, New York, NY, USA, 145–152.

RHODY, B., AND HEPPLER, G. 1993. Controlling the Dynamics of Mass Capture by a Single Flexible Link. In IEEE International Conference on Robotics and Automation (ICRA), IEEE Comput. Soc. Press, 496–501.

TACHI, S., 2009. Telesar2 Project Page. `http://projects.tachilab.org/telesar2/`, May.