Multitouch Puppetry: Creating Coordinated 3D Motion for an Articulated Arm

Michael Kipp, Quan Nguyen
DFKI
Campus D3.2
66123 Saarbrücken
Germany
{firstname.lastname}@dfki.de

ABSTRACT

Controlling a high-dimensional structure like a 3D humanoid skeleton is a challenging task. Intuitive interfaces that allow non-experts to perform character animation with standard input devices would open up many possibilities. Therefore, we propose a novel multitouch interface for simultaneously controlling the many degrees of freedom of a human arm. We combine standard multitouch techniques and a morph map into a bimanual interface, and evaluate this interface in a three-layered user study with repeated interactions. The multitouch interface was found to be as easy to learn as the mouse interface while outperforming it in terms of coordination. For the analysis, we propose a novel quantity-based coordination measure. For the systematic exploration of the design space, we suggest using dataflow diagrams. Our results[1] show that even complex multitouch interfaces can be easy to learn and that our interface allows non-experts to produce highly coordinated arm-hand animations with subtle timing.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

Keywords: Multitouch interaction, 3D user interfaces, character animation

INTRODUCTION

The traditional puppeteer uses complex coordinated hand movements to bring a wooden puppet to life. Everybody uses equally complex hand movements to master simple everyday tasks like tying one's shoes. This means that the human hand can control many degrees of freedom simultaneously, an idea known as *whole-hand input* in human-computer interaction [20]. However, since whole-hand input requires special hardware (e.g. data-gloves) traditional user interface design has long restricted itself to simple interaction techniques which can be afforded by conventional mouse and keyboard devices. With the advent of *multitouch* screens and touchpads as omnipresent input devices, we can now develop complex interaction techniques, building on simpler, well-known multitouch techniques (e.g. for rotation and scaling), thus explotinq what Sturman [20] calls *dexterity*, the ability of the hand to integrate movements and senses into higher levels of competence.

In this paper, we present an complex but easy-to-learn interface for controlling a 3D human arm (Fig. 1). We argue that it is high time to design, build and evaluate concrete cases of complex interaction techniques, which are usually more than the sum of their parts, and can therefore only be evaluated in the context of a concrete task.

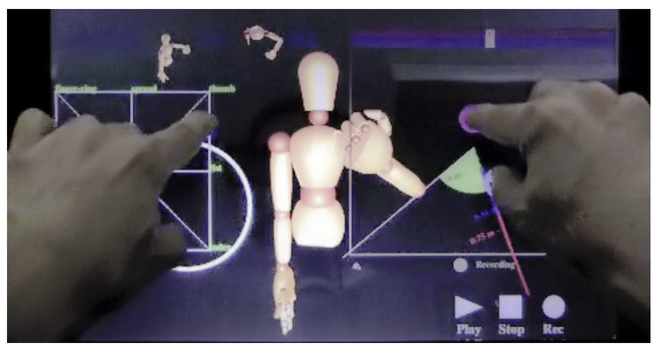

Figure 1: Our bimanual multitouch interface allows simultaneous control of the various degrees of freedom of a human arm.

Posing a humanoid skeletal structure is desirable for various applications including computer animation, film/theater/dance choreography and teleoperating robotic arms (in space or underwater). In computer animation, it allows not only more efficiency but also more creativity: so-called performance animation makes possible a spontaneous and improvisational exploration of ideas as opposed to the traditional pose-to-pose approach which requires planning ahead [22]. Performance animation can be used in conjunction with a voice track or a music piece in the background to immediately explore appropriate co-occurring motions (Fig. 2). While today character animation requires a high level of expertise, new interaction techniques can make it accessible to non-experts, including children, so that it becomes as easy and customary as drawing and painting.

[1]See video on http://www.embots.org/projects/multitouchpuppetry.html

ITS'10, November 7-10, 2010, Saarbrücken, Germany.

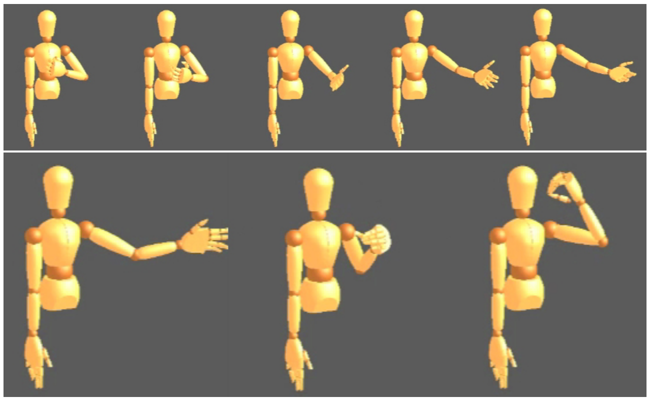

Figure 2: Two example motions created to accompany an audio track. Top: audio was a slow pop ballad, notice the gradual opening of the hand. Bottom: audio was a voice saying "... drinking Gin at 9 o'clock ...".

We prefer multitouch over special purpose technology (e.g. motion capture, 6 DOF mice, data-gloves) because it is widely available, low-cost, causes little fatigue and a number of standard interaction techniques already exist. Compared to performing the desired arm/hand motion oneself while being captured [6], multitouch is less tiring which can be crucial for professionals who have to perform motions over and over again. Moreover, our interface integrates well with traditional interaction techniques (buttons, menues, 2D navigation) so that it can be embedded into a traditional animation framework on a single interactive surface. Lastly, a certain "mapping distance" between control space and output space may even be beneficial in terms of artistic style[2].

In recent years much foundational work on multitouch interaction has identified properties and benefits of specific components of multitouch interaction [7, 15] and generalizable atomic tasks like 3D rotation (e.g. [10]). However, such studies usually apply to particular aspects only of any concrete interface, e.g. the study by Kin et al. [15] only applies to target selection tasks. While these foundations are important and continue to grow, we explore an integrated interface for a concrete task, to see how close we can get to the performance of the mouse while retaining the advantages of multitouch in terms of simultaneous control of many degrees of freedom. In summary, we see our contributions to the research field as follows:

- A novel bimanual multitouch interface for interactively posing a human arm in terms of hand location, arm swivel, hand orientation and hand shape
- Proposing dataflow diagrams for the systematic exploration of the multitouch design space
- A novel measure of coordination that takes into account motion quantity
- A three-layered experimental design over repeated interactions for evaluating a posing interface beyond first exposure

[2]For instance, in drawn cartoon animation, *rotoscoping* refers to a technique where the movement is copied from actual film recordings of real people. The result usually looks uncanny because of the too realistic movement compared to the cartoonish surface rendering.

RELATED WORK

Our suggested interface relates to the area of *performance animation* in computer graphics. Neff et al. [17] suggest an approach for mapping 2D mouse input to high-dimensional skeleton space with so-called *correlation maps* which transform the input to meaningful high-dimensional output (e.g. controlling hand location but rest of body moves according to previously found correlations with the hands). In contrast to our approach, control is less direct due to the mapping. Dontcheva et al. [6] use vision-based capture to allow non-expert animators create partial motions (a wave, a walk, head motion etc.). While this is a powerful approach it requires special hardware. Oore et al. [19] created two 6 DOF devices (bamboo sticks) that control one body group each (e.g. the legs). Both approaches use layering, i.e. the successive recording of different body parts, as a strategy to achieve a complete animation. Layered recording would be applicable in our approach as well, since we only control a single arm. However, these approaches do not control the whole range of output space features as our approach (from whole arm down to hand shape). In spatial keyframing [12] the authors provide a spatial map where each point corresponds to a mixture of poses of the target character. This technique has limitations in the complexity of possible output poses. We use a similar technique that we call *morph map* for controlling hand shape.

In our suggested interface we need to simultaneously control hand location (translation), arm swivel (rotation), hand orientation (rotation) and hand shape (selection and/or interpolation). Hancock et al. [11] provide a survey of possible rotation & translation techniques and give design guidelines: consistency, completeness, and snapping. We utilize consistency in the sense that we map translation input to translation output and rotation input to rotation output. We reject completeness since the human anatomy has rotational *joint limits* which are good to implement because they reduce error. This could be seen as a form of snapping. Another form of snapping that we implemented is collision avoidance, i.e. the subject cannot penetrate the torso with the controlled hand. A particularly successful form of 2D rotation is RNT [16] which combines translation and rotation in a single-touch interaction model. This could be utilized for our application in the sense that e.g. the height of the hand position implies a certain arm swivel. However, we wanted the user to have full control over arm swivel since this has potential in expressing different shades of status/personality (more dominant = elbows higher). In follow-up work, Hancock et al. [10] suggest a shallow-depth scenario where the z dimension is restricted to a small stratum. In their studies they examined different techniques for the simultaneous manipulation of translation and rotation and found the three-touch input to perform best. This shows that in general, complex interaction techniques can be better than supposedly simpler ones and that the separation of functionality unto two hands can be beneficial, which supports the design decisions in our interface.

For many applications it is desirable to be able to control different degrees of freedom at the same time. Sturman introduced the notion of *whole-hand input* with a data-glove and used the notions of naturalness, adaptability and dexterity to describe this type of interfaces [20, 21]. While multitouch interfaces restrict whole-hand input to a 2D surface, Sturman's principles still apply, especially the idea of dexterity which means that the hand can learn to combine multiple simple actions into one single smoothly performed action. To what extent the parallel control of many degrees of freedom is actually exploited can be quantified with various *measures*. Jacob et al. [14] introduce the measure of *integrality* using an algorithm that segments the trajectory and classifies segments as integral (movement in x-y and z) vs. city-block (movement in either x-y or z). The ratio of these numbers is taken as a measure of coordination. Zhai and Milgram [25] suggest *efficiency* as a measure for coordination. In the case of a translation this corresponds to the deviation from the shortest path, normalized against the length of the shortest path. Another measure they mention but do not use is *simultenaiety* which measure the overlap of activity in different DOFs. Balakrishnan and Hinckley suggest the measure of *parallelism* [2]. Note that all of these measures need a target, so they cannot be used in a free scenario. We suggest a novel measure of coordination which extends the measure by Jacob et al.

Multitouch interfaces open a wider design space which has been empirically explored. Kin et al. [15] found that one-finger direct-touch delivers a large performance gain over a mouse-based device. Based on their study (a selection task) they recommend to reserve multi-finger input for controlling multiple degrees of freedom. Positioning a point in 3D has been addressed in *balloon selection* [4] where two fingers on a surface control a string with a balloon attached, thus specifying a 2D position in the plane and a 1D value for height. This technique is very similar to our hand positioning technique and was found to be superior to e.g. more direct interaction in terms of low-fatigue and precision. Balakrishnan and Hinckley [1] found that two-handed tasks can be performed well, even if the kinesthetic frame of reference (input space) does not correspond to the graphical output space (e.g. by introducing a rotational offset). In our design it allowed us to introduce a visual offset from between input space and output and to apply a gain to e.g. the degree of rotation without having to expect degraded performance.

PROBLEM FORMULATION

Before describing the interface design, we want to formulate the problem of posing a human arm in 3D. This problem has two parts. First, which methods are used for posing a human arm and which are the resulting degrees of freedom? Second, how is the 3D structure visualized? For the rest of this paper, we assume a coordinate system where the y-axis is pointing up, the x-axis is pointing right and the z-axis is pointing out of the screen.

Methods for posing a human arm and resulting degrees of freedom

A 3D skeleton is usually modeled using a tree-like structure where each node, called a *joint*, is located in the local frame of reference of the parent node. This means that moving a parent joint (e.g. shoulder) implies moving all children joints (e.g. elbow or wrist). To change the pose of a hierarchical structure, one can either manipulate each joint separately (forward kinematics) or specify the target position of an "end effector" joint, like the wrist, and compute the angles of all parent joints (IK = inverse kinematics). We use IK for an intuitive positioning of the wrist. This means the user needs to input a 3D point target location. This leaves the so-called *arm swivel* unspecified (given that we solved the IK problem, imagine a line from shoulder to wrist; one can rotate the whole arm around this axis and would still have a valid pose). Therefore, we need another 1 DOF control for arm swivel. Experience has shown that it is furthermore beneficial to keep the arm swivel constant when the pose changes due to IK. For this, we need to define an absolute arm swivel which orients itself at the global y axis[3]. The same holds for hand orientation (wrist joint). It is confusing if the hand orientation constantly changes because parent joints change due to IK. Therefore, we also keep a constant absolute hand orientation with respect to the global y axis. We did not implement joint limits (which would prevent the skeleton from assuming anatomically incorrect poses) but our policy of keeping arm swivel and hand orientation global significantly reduces the probability of anatomically incorrect poses. Furthermore, we implemented collision avoidance between arm and torso so that the potential for incorrect poses is further reduced.

To sum up, we need controls for the following input parameters: hand location (3 DOF), arm swivel (1 DOF), hand orientation (1 DOF), and hand shape.

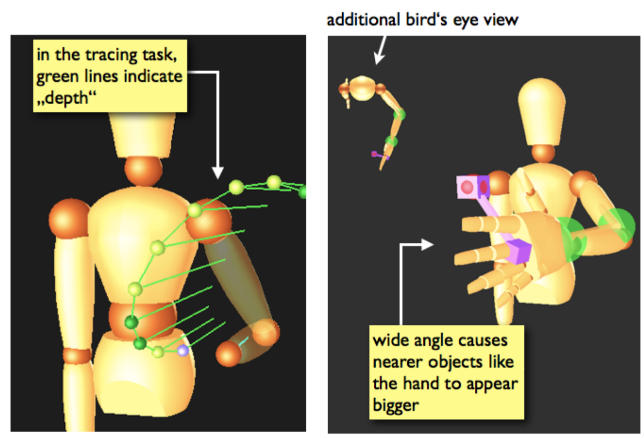

Figure 3: 3D cues for facilitating visual readability.

Visualizing the 3D human arm structure

When using regular 2D screens our pilot studies resulted in users having problems "parsing" a 3D scene which may negatively influence the validity of the user study. Therefore, we introduced a number of measures to improve 3D readability in terms of viewing angle, focal length and depth visualization (Fig. 3).

With regard to viewing angle we offered multiple views but found them of limited help since, during a task, most users focus on only one of the views at a time. We nevertheless included two smaller alternative views on top of the screen, including a bird's eye view to clarify depth. The main view

[3]This is a simplification due to the fact that the torso does not move. Otherwise, one would take e.g. the hip-neck vector as the "up" vector.

we angled slightly so that depth is present to a certain extent (and to guarantee visibility of the z-bars, see below). To exploit *relative size* as a 3D cue, we set the focal length of the camera to wide angle. This made closer objects (especially the hand) appear bigger in an exaggerated way. Finally, for the tracing task, we introduced *z-bars* which are lines parallel to the z-axis (coming out of the screen) that run between an object and the frontal x-y plane. These give a precise indication of the current depth (Fig. 3, left).

INTERFACE DESIGN

Multitouch Interface

The overall objective was to use as many input degrees of freedom as possible while keeping the interface easy to learn. First, we decided to co-locate the input and output space, i.e. the touch area is located on the same screen as the visualized skeleton. However, we rejected direct-touch because of our limited screen size (15.4") which would have caused occlusion. Occlusion is problematic since hand shape control is a target feature and the virtual hand would be completely covered by a directly controlling finger (Fig. 1). Therefore, we introduced a spatial offset between touch area and skeleton. Second, we devised a two-handed asymmetric interface where the right hand controls overall arm pose and the left hand controls hand shape and orientation. Third, we decided on a two-finger interface for each hand for controlling multiple degrees of freedom each (as recommended by [15]).

Dataflow diagram for exploring the design space

An important tool in the exploration of the design space is a graphical depiction of the interface logic in the form of dataflow diagrams (cf. [13]). Interfaces can be described as a mapping from an *input space* (finger coordinates, key presses, mouse scroll wheel position etc.) to an *output space* (arm swivel angle, 3D location of the hand) with several transformations in-between. Fig. 4 shows the logic of the multitouch interface (upper two diagrams) and the mouse interface (lower diagram). The diagrams explicitly shows options (which finger for which input, absolute vs. relative) and parameters (spatial offset, gain) and thus facilitates the exploration of the design space and the tuning of the final system. In our interface, "absolute" means that every point on the input space (= screen) is mapped to exactly one point in the output screen. While this can cause jumps in the output, positioning can be much faster and for people not used to 3D navigation the absolute mapping facilitates getting familiar with 3D space. While for this project we used the methodology only on paper, we are currently working on graphical tools that allow online adjustments at runtime to tune the interface. Especially for complex multitouch interface, a systematic tuning method can make all the difference.

Arm posing (dominant hand)

An arm pose can be specified with the 3D location of the hand and the swivel angle of the arm. In our interface (Fig. 1) this is performed with two fingers of the dominant hand. The index finger position controls the position of the hand in the x-y-plane. The distance between thumb and index finger controls the z coordinate of the hand, the depth. The two touch points define a line L. The angle between L and a fixed axis defines the arm swivel. This is an absolute way of defining rotation which we found more practical for controlling arm swivel in conjunction with 3D-locating the hand with single sweeping strokes. The gain for arm swivel had to be adjusted to ensure that the user stays within her "comfort zone" concerning the rotation of her hand. Adjusting the gain value is also relevant for the z-coordinate of the hand, where we again chose an absolute mapping. Doing such adjustments in realtime (see above) would significantly speed up the design process.

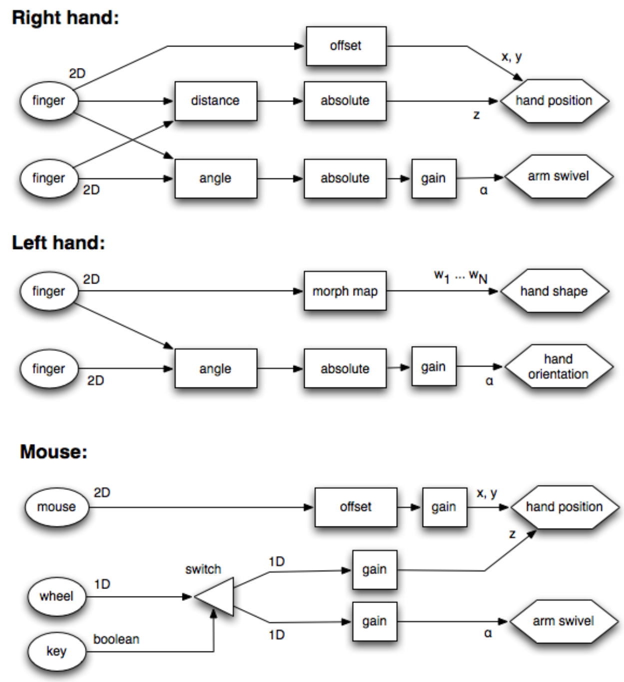

Figure 4: Diagrams of the bimanual multitouch interface (upper two) and the mouse interface (lower), showing e.g. how the left hand controls hand shape and orientation at the same time.

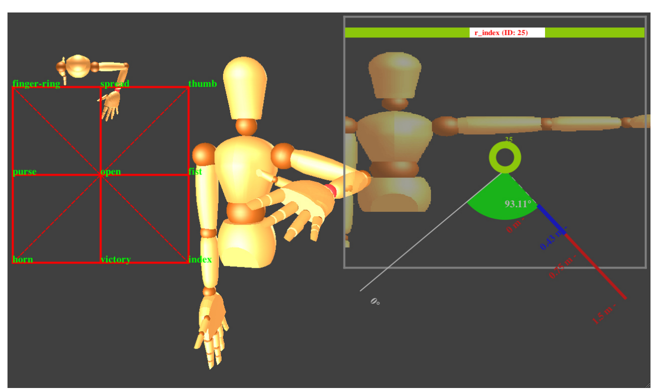

Figure 5: The multitouch interface: 3D skeleton in the center, arm control on the right, hand shape morph map on the left. Finger touch is indicated by a green circle and auxiliary lines for angle and depth.

Hand posing (non-dominant hand)

The non-dominant hand (in Fig. 5) controls two aspects of the virtual hand: the orientation of the palm and the actual hand shape (e.g. fist, open, indexfinger, see Fig. 6). This may look counterintuitive given Guiard's finding that the non-dominant hand usually controls the frame of reference while the dominant hand controls more precise movements within this reference frame [9]. However, in animation the movement of the hand in space seems to be the more important motion which may require more precision than controlling hand orientation. In this regard, our interface is asymmetric and in line with Guiard.

To control the orientation of the palm we used the usual two-finger rotation technique (index finger and thumb). To control hand shape we used a morph map (see below): the absolute position of the index finger determined the interpolation between shapes. We decided on an absolute control because otherwise we would have had to move the morph map grid on the screen.

To combine two different tasks like rotation and shape in a single control may sound counterintuitive at first. Note, however, that changing the rotation is possible without changing the shape (by keeping the index finger at the same spot, only moving the thumb, which is quite intuitive), so it is easy to separate the choices. Given this separability it seemed intuitive to combine aspects of the virtual *hand* in the non-dominant (left) control, while aspects of the virtual *arm* were combined in the dominant (right) control.

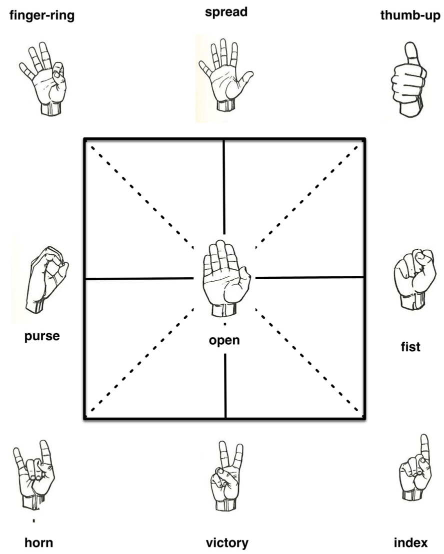

Figure 6: Morph map for hand shapes.

Morph map
For controlling hand shape we designed what we call a *morph map* (Fig. 6) which is similar to spatial keyframing [12]. Our morph map is a square with 9 key points: the center, all four corners and four mid-points between corners. Each key point is associated with a hand shape that is depicted on the map (e.g. fist, open hand, victory sign). When directly touching a key point, the associated hand shape is taken. When touching an arbitrary point p, the triangle that contains p is determined and the resulting hand shape is an interpolation between the three vertices v_1, v_2, v_3 of the triangle. The weights for this interpolation are computed by

$$w_i = \tilde{n}\left(1 - \frac{d(p, v_i)}{\sum_{k=1}^{3} d(p, v_k)}\right)$$

where d is Euclidean distance and $\tilde{n}$ is a normalizing factor to make the w_i add up to 1. We also tried using the three or four *nearest* points but this resulted in more jumps and unnatural poses. With our technique we reduce the number of boundaries where one of the interpolation ingredients is exchanged which causes potential jumps.

Since a morph map takes the high-dimensional space of all hand shapes and projects it down to two dimensions only, it cannot provide all shapes and all transitions between them. The placement of hand shapes is a design process which was done manually. We wanted hand shape locations to be learnable, easy to reach and facilitating frequent transitions. From a technical point of view, the interpolated shape at any point should look natural and transitions should look continuous. Our placement was a trade-off between design and technical considerations. In the center, we placed the open hand because it is the most frequent hand shape and best suited for a "pass-through" shape. Similar hand shapes were chosen as neighbors to guarantee natural interpolated poses. To increase learnability we tried to create a "topics" for each region, e.g. top-left for *spread-out hands* or the line from middle-left to middle-right as *closed hand shape* and the bottom row as *special finger-out hand shapes*.

Mouse interface (control condition)
To evaluate our interface we had to compare it against a "baseline" interface that we could use as a control condition. Therefore, we designed a *mouse* interface and made it as quick and easy as possible for fair comparison. Mouse movement was translated to the x-y-plane of the virtual hand. The mouse's track wheel controlled the z-coordinate (depth) of the hand. With a function key the user changed the *mode* of the track wheel to (a) hand rotation or (b) arm swivel. While the mouse has the drawback that mouse position and track wheel are clearly separated controls, it is often take as a baseline because of its high familiarity [5]. For controlling the hand we required the user to "clutch". This was done by pressing the *right* mouse button and not, as usual, the left one. This allowed the user to simultaneously operate the mouse wheel with another finger. While this sounds inconvenient, subjects picked it up very quickly and allowed more coordinated motions.

USER STUDY
During the design of our interface we ran multiple pilot studies. One insight was that the mouse interface usually won over multitouch in terms of performance and subjective preference in the *first* session, probably due to the high familiarity. However, we also found that after several sessions performance converged. We therefore decided to conduct a repeated-interactions study with 7 sessions spread over a period of 3 weeks to give subjects a chance to adapt to the new interface and to analyze the respective learning curves.

To evaluate the different aspects of performance, coordination and user satisfaction, specifically in terms of creativity, we devised a three-level design, similar to the evaluation by Barnes et al. [3]. They used a three-layered evaluation technique to evaluate a tangible interface for cutout animations. The three layers were: low-level dexterity, mid-level story task and high-level storytelling. In our design we use the three tasks of pose matching (similar to docking tasks [10, 25]), trajectory tracing and a creative task. In our context, coordination refers to the idea that the multitouch interface allows the simultaneous control of multiples DOFs and how this can be quantified.

Screen Setup

We used a 15.4" multitouch screen manufactured by Stantum and positioned it at an angle of about 45 degrees in front of the subject. The multitouch device has a screen resolution of 1280x800 and a touch repositioning accuracy of < 4 pixels (< 0.5 mm). It was connected by USB to a computer running both the multitouch driver and the application.

The screen was divided into three areas (Fig. 5). The 3D skeleton to be manipulated was in the center. On the right side was the touch area for arm control. In this touch area, a static skeleton figure was displayed for better reference. On the left side, the morph map for hand control was displayed. The displayed skeleton, animated with Java3D, had an articulated arm structure with the following joints: shoulder (3 DOF), elbow (1 DOF), wrist (1 DOF: rotation around lower arm axis). The five-finger hand was modeled with 2 joints for the thumb and 3 joints for the other fingers (2 DOF at the "root" of the finger, 1 DOF for others). We used the CCD algorithm for fast IK solving [24] .

Subjects and Procedure

Six paid subjects (3 female, 3 male, ages 21-30, students, German native speakers) participated in a series of 7 sessions each. Five had no prior exposure to multitouch, one had tried it several times but not on a regular basis. All subjects were asked to not use multitouch interfaces externally for the duration of the whole experiment.

Subjects were scheduled such that between sessions there was a break of 1-2 days, except for the 7th session which was conducted 5-9 days after session 6 to see whether performance would degrade differently between conditions. In each session, subjects did two tasks for each device (mouse, multitouch), except for session 3 and 6, where a third "creative task" had to be done as well. In each session, the subjects would first perform task 1 with interface A, then task 1 with interface B, then task 2 with interface A, then task 2 with interface B. Interfaces A and B were mouse/multitouch respectively (cross-balanced). Before task1, the subject would run through a device-dependent training set to train (a) movement in the x-y plane, (b) movement along the z-axis and (c) full 3D movement. Before starting task 2, each subject ran another training set before doing the actual movements. After completion of each task, subjects were asked to fill in a questionnaire where multitouch and mouse were compared with a set of questions using semantic differential scales, asking e.g. "which interface do you prefer?". In sessions 3 and 6, the subject performed creative task 3, only for multitouch, after finishing both tasks. After task 3, the subject filled in a questionnaire about the usefulness of the multitouch interface in task 3. After the whole experiment each subject went through a structured interview.

For tasks 1 and 2 we collected 10 poses (task 1) and 10 trajectories (task 2) per session. The resulting 120 poses and trajectories were randomized, split into 10 sets and ordered so to be cross-balanced over the interface condition. The poses were manually created, whereas the trajectories were extracted from human recorded motion capture data.

Tasks

Task 1: Pose matching In this task, the subjects had to position the arm such that it matched a graphically indicated "ghost arm" (Fig. 7), including arm swivel and hand orientation but excluding hand shape. In the multitouch interface, the right hand controlled hand location and arm swivel and the left hand controlled hand orientation. In the mouse interface, subjects could point and click/drag directly on the skeleton, using the wheel for depth. For hand orientation and arm swivel the wheel plus a keyboard key (CMD and ALT) were used. This task encourages a sequential strategy where the hand is first moved in the x-y plane, then in z, and finally, arm swivel and hand orientation are adjusted.

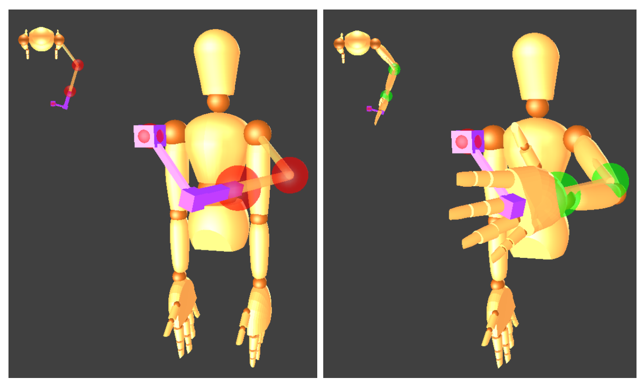

Figure 7: In task 1, subjects matched a static arm pose, including hand orientation.

Task 2: Trajectory tracing In the second task we presented a *trajectory* (Fig. 8), visualized with a series of spheres that the user was asked to swipe through in the correct order. To clearly show this order, a "glow" was cyclically propagated through the spheres. To reduce visual clutter, only the first part of the trajectory was shown. When the subject hit a sphere (and it was next in the given order), the sphere would vanish, accompanied by an audio signal, and a new sphere at the end of the path would appear, unless the end had already been reached. Note that the trajectories were taken from motion capture recordings of a human performer so that they modeled the characteristics of potential gesture strokes which would be the goal of actual character animation activities. Since the subjects only controlled hand location, excluding arm swivel and any hand controls, we replaced the articulated hand with a simple sphere. We also made the arm slightly translucent to reduce occlusion of the target trajectory.

Task 3: Creative task To test the multitouch interface with the full range of controls, we added a third task, to be done

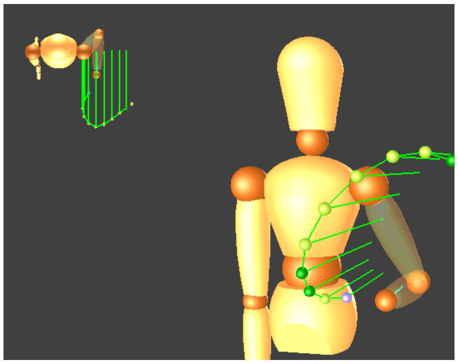

Figure 8: In task 2, subjects traced a given path, depicted with green spheres.

with multitouch only. We found no equivalent mouse controls that would have made a fair comparison. This task allowed us to see whether performance animation is generally feasible with our interface and to get an impression of quality to expect. The subject was instructed to create "appropriate" movement for four different kinds of audio tracks while listening to them:

- M1: pop song[4] (strong beat)
- M2: pop ballad[5] (slow, romantic)
- V3: male voice track[6] (harsh, energetic)
- V4: female voice track[7] (soft, melodic)

The audio pieces were short (music pieces: 40 sec, voice tracks: 10-20 sec) and the subject could listen to them twice before creating an the movement. Each subject had two attempts for every audio piece. The creative task was performed in session 3 (to guarantee a certain proficiency) and in session 6 (to examine improvements). Fig. 2 depicts two motion segments from M2 (top) and V4 (bottom). The produced clips showed some nice temporal dynamics (both smooth, slow motions and erratic beats) and examples of complex coordination of e.g. change in hand shape together with a forward motion of the hand (top row in Fig. 2).

Measuring Coordination

We consider the multitouch interface's greatest strengths the simultaneous control of multiple degrees of freedom. Various measures have been suggested, some of which depend on a target trajectory, e.g. *efficiency* [25] and *parallelism* [2]. While such measures could be used on tasks 1 and 2 they are unsuitable for the creative task 3 where no prescribed/optimal paths exist. Jacob et al.'s measure of *integrality* can be applied to any movement [14]: they computed the ratio of movement in the x-y direction vs. movement in z direction. However, their measure looked, for any given frame i, only whether there was movement at all in x-y or z, ignoring the exact amount of movement. However, if we look at the x-y plane vs. the z axis in movements of the human arm we found that the quantity of movement in each of the two spaces mattered. Movement mainly in the x-y plane with minimal z movement would look unnatural and this should be reflected in the measure.

Therefore, we suggest an alternative measure of coordination which takes *quantity* into account. We do this by integration, i.e. we sum up the distance travelled in each of the dimensions (e.g. x-y plane vs. z-axis) for a certain time interval Δt and compare these. Total coordination is measured by splitting the motion in question into N intervals of size Δt. We first define coordination $C^i_{\Delta t}$ for each interval where $i \in \{0, \ldots, N-1\}$. Technically, it is convenient to choose a Δt that is a multiple of the frame rate, in our case 40ms, because this is the distance between two consecutive frames. We can then measure the distance travelled between frames $k-1$ and k, both for the x-y plane $d_{xy}(k)$ and the z axis $d_z(k)$. We then sum up all minimums $S_{min} = \sum \min(d_{xy}, d_z)$ and all maximums S_{max} that are in Δt, and define $C^i_{\Delta t} = \frac{S_{min}}{S_{max}}$ which yields a value in $[0, 1]$ where 0 means movement only takes place in one of the dimensions whereas 1 means that movement quantity is exactly equal in both dimensions. We define the *overall coordination measure* as the average of the interval measures:

$$C_{\Delta t} = \frac{1}{N} \sum_k C^k_{\Delta t}$$

The resulting value is again between 0 (= no coordination) and 1 (= perfect coordination). However, note that the bigger Δt is, the lower coordination becomes because the two sums in $C^i_{\Delta t}$ diverge with growing interval size with $S_{min} < S_{max}$, making the quotient converge to zero. We therefore chose the smallest possible Δt, in our case 40ms.

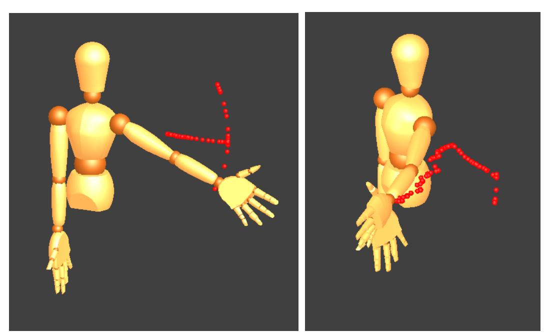

Figure 9: Two examples of applying our coordination measure C. The left trajectory lies almost completely in the x-y plane and has $C = .01$, whereas the right one travels through x-y and z, and yields $C = .14$.

We filtered out jumps in the trajectory that were due to absolute positioning. We discarded every frame k where $d_{xy}(k)$ or $d_z(k)$ exceeded 1.5 times[8] the respective standard deviation. See Fig. 9 for example trajectories.

Results

Completion time We first compared how well subjects performed on tasks 1 and 2, comparing mouse and multitouch,

[4] Bee Gees "Night Fever"
[5] Celine Dion "My Heart Will Go On"
[6] German literary critic Marcel Reich-Ranicki in the TV show "The literary quartet"
[7] American writer Elisabeth Gilbert at her TED talk in 2009
[8] Since jumps can be quite small given the restricted input area we had to make the filter this strict.

by measuring completion time. Fig. 10 shows average completion times for each session (std. dev. omitted for visibility). The curves clearly show a strong learning effect during sessions 1-3 (task 1) and 1-4 (task 2) and a convergence against an asymptote that seems to be the same for both multitouch and mouse under each task. For each task, we compared mouse vs. multitouch performance with a Wilcoxon signed rank test which showed no difference. The shape of the curve corresponds to comments of the subjects. Most of them found using the multitouch unfamiliar and difficult in the beginning but already found the second and third session much easier.

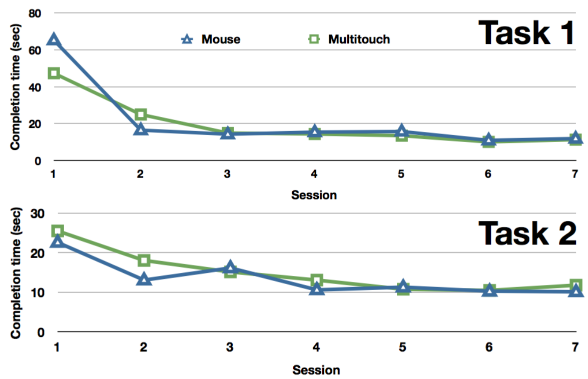

Figure 10: Average completion times over all sessions.

Coordination We used our coordination measure C for two questions. First, which interface (mouse/multitouch) encourages higher coordination? Second, does the degree of coordination with the full multitouch interface (task 3) increase over time?

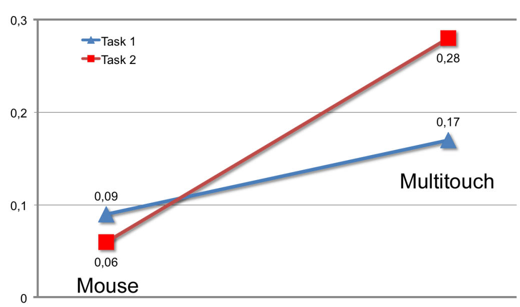

Figure 11: In terms of coordination, multitouch (right) significantly outperforms mouse (left) in both tasks.

For the first question we hypothesized that the multitouch interface encourages a higher degree of coordination. Note that we measure coordination only for *hand movement*, comparing multitouch with mouse. Concretely, we computed coordination C on the trajectories of mouse/multitouch tasks in session 7, in the most advanced state of training (Fig. 11). Using a two-tailed Wilcoxon signed rank test we found that the multitouch trajectories had a significantly higher coordination compared with the mouse trajectories. In task 1, multitouch coordination was 0.17 vs. mouse coordination 0.09 (Z=2.20, p<.05). In task 2, multitouch coordination was 0.28 vs. mouse coordination 0.06 (Z=2.20, p<.05). Thus, in both tasks our hypothesis was confirmed.

For the second question, we compared coordination for the creative task (task 3) in session 3 against the later session 6. Although average coordination increased from .19 to .22, this did not become significant in a two-tailed Wilcoxon test ($Z = .73; p = .46$).

Questionnaire and interviews Our post-task questionnaire contained 16 questions regarding usability and user experience. We asked for a decision for either multitouch or mouse on a 5-point differential scale where the middle position was labelled "both are equal". For analysis, we transformed this to an interval between -2 for mouse and $+2$ for multitouch. We summed up ratings for each question over all subjects, sessions and tasks (because the results were similar for both tasks) and computed a chi square statistic, comparing the ratings against the expected neutral value of zero. We adjusted the alpha level according to the Bonferroni method to 0.003. The following questions were answered significantly *in favor of* the multitouch interface (all others were not significant):

- Which interface was more useful? ($M = .43; \chi^2 = 19.64; p < .003$)
- Which interface allowed a faster solution of the tasks? ($M = .62; \chi^2 = 35.93; p < .001$)
- With which interface were you more satisfied? ($M = .43; \chi^2 = 23.00; p < .001$)
- Which one would you recommend to others? ($M = .62; \chi^2 = 32.07; p < .001$)
- Which device was more fun to use? ($M = 1.13; \chi^2 = 67.89; p < .001$)
- Which interface do you prefer? ($M = .75; \chi^2 = 45.82; p < .001$)

Discussion

Our results first of all show that the major advantage of the mouse, the high familiarity, can be made up for with a good combination of multitouch controls for the specific task of posing a human arm. It was important to look at a series of interactions to see the development over time: the biggest leap seemed to have taken place already in session 2. In the interviews, the subjects reported that in the first session they found the multitouch interface difficult but that already in the second session they felt more comfortable, having lost their "fear" of the multitouch. One major difficulty in the first session was also the understanding of the 3D scene, even with our multiple visual enhancements, but this was usually gone in the second session. However, in future studies it may be advisable to control for 3D understanding problems by including self-ratings.

In terms of performance, both controls perform equally well for matching a pose and tracing a trajectory. However, while multitouch can warrant the same performance as the mouse, it offers more control when we consider the full interface which is impossible to implement with a mouse interface because there are not enough continuous output degrees of freedom. In the creative task, subjects made use of all controls, moving the hand in all directions (including z), changing hand shape and orientation (see companion video). Our coordination measure showed a clear advantage of multitouch over mouse in terms of simultaneous control of the two dimensions of x-y plane and z-axis. The fact that this difference was more accentuated in task 2 confirms our design idea to enforce more coordination with the trajectory tracing task. Given our results, we would deem a coordination below 0.1 low because this corresponds to the sequential mouse strategy. For good coordination we would take the multitouch value of task 2 as an indicator, together with the value from the creative task, so that values above 0.2 already can be taken as reflecting good coordination.

In terms of learnability the fast convergence may be explained by the fact that the tasks of pose matching and tracing were rather simple. While we created this particular interface as a case study to prove that complex interfaces can be learned quickly, tasks 1 and 2 only tested a subset of the interface. Therefore, we investigated the full interface in task 3, the creative task, and found high coordination values, close to those of task 2. However, the subjects did find the full bimanual control to be challenging when the morph map was added in the creative task. The effectiveness of the morph map for hand shape control must certainly be further investigated in terms of optimal hand shape placement and in terms of whether it combines well in the bimanual interface. Further analysis is also required because of the low sample size (N=6). More specific investigations include looking at how our coordination measure develops over time[9].

As for subjective measures we found that most of the subjects preferred the multitouch device in their ratings. Subjects found the multitouch interface "more useful" and allowing a "faster solution of the tasks" although this was not the case according to objective measures. Those who favored the multitouch in the interview called it "more sexy" and "more intuitive". This is in line with the idea that multitouch allows a more playful and creative exploration of motion space. When looking at the outcomes of our creative task, it is striking how much people played with tempo, both of the hand movement as well as the hand shape change (e.g. slowly opening a hand). The results of the creative task can only be reported verbally here (Fig. 2). The resulting videos show that motion is coordinated in various respects. In one example the hand opening is performed in parallel to moving the hand in 3D space. In all videos, the tempo of the music/speech is clearly reflected in the movement. We also saw nice examples of interrupting/correcting the current motion. In the voice tracks iconic gestures are used to underline concepts in speech ("drink", "behind"). Our companion video shows some examples at the end of the video. Future work will include a quantitative approach to evaluating animation results, e.g. by letting independent judges rate the quality.

In our user studies we focused on novice users. While this is in line with our goal to provide an easy-to-use interface for the general public, it would be interesting how *expert users* like professional animation artists would react to this interface. Testing expert users is more difficult because experts differ not only in their level of expertise but also in the tools they use and in additional requirements they may have (e.g. manual camera control). With each tool comes a set of individually well-trained interaction techniques. In terms of subjective evaluation this could lead to more reservations when it comes to changing old habits. On other hand, 3D artists are also used to adjust quickly to new tools and environments. In future work, the both the initial performance and learning curve of professional animation artists could be compared to our current results of novice users.

Alternative approaches to performance animation use special input hardware or computer vision. Special hardware has the disadvantage of additional cost and the inconvenience of having another single-purpose device on the desk. In contrast, new vision-based systems like Microsoft's Kinect allow unobtrusive, low-cost detection of full-body motion. While this opens up exciting possibilities for interface design we still see several advantages for multitouch in the area of animation. First, multitouch is low-fatigue since small finger motions are translated to large arm motions. Second, multitouch screens can provide visual annotations that facilitate control like e.g. the *shadow guides* described in [8]. Third, multitouch can better accommodate non-realistic motions like bulging eyes or wagging a tail that have no clear correspondence to human body motion. Lastly, vision-based systems still have difficulties recognizing hand shape, although it was shown that this can be robustly solved using a specially colored glove [23]. For future work, we plan to include alternative input techniques to compare them against our suggested multitouch interface, especially for the creative task where no equivalent mouse interface could be found.

CONCLUSIONS

We presented a complex yet easy-to-learn multitouch interface for interactively posing a human arm, including hand location, arm swivel, hand orientation and hand shape. We reported details on designing the interface which combines well-known techniques for rotation and zoom with a novel form of a morph map. For evaluation we designed a three-layered user study with repeated interactions. Only in repeated interactions (7 sessions spread over 3 weeks) can the learning process be observed. In our case, the most significant improvements took place within the first three sessions. We were able to show that multitouch and mouse have identical performance for pose matching and trajectory tracing tasks. However, multitouch achieved a significantly higher degree of coordination when compared with the mouse variant. For this we introduced a novel measure for coordination which takes into account movement quantity. In subjective user ratings, the multitouch was generally preferred and comments indicate that most subjects became comfortable after completing the second session. While much of current multitouch research is concerned with laying the foundations of how to evaluate atomic components of multitouch interaction, we consider it equally important to design complex integrated interfaces that test general guidelines against the conditions of a concrete domain.

Future work

Future work aims at extending our approach to the whole body and testing it on large screens for collaborative animation. Scaling to full-body control can be done in three principal ways. First, using *layering*, different motion aspects are recorded separately and later combined [6]. Second, mul-

[9]For technical reasons, we were not able to reconstruct the data to compute coordination over all seven sessions. So we cannot report this.

tiple people can *collaborate* on a single puppet, as done in the movie industry [21]. Third, various body parts can be *intelligently coupled* (e.g. hand with upper body) using correlation maps [17] or using the simplicial configuration modeling approach [18]. All approaches involve the development of novel multitouch controls for every kind of human motion: facial expression, body posture, stepping, walking etc.

Acknowledgements
We thank Kerstin H. Kipp (Saarland University) for her competent advice in the user study. This research has been carried out within the framework of the Excellence Cluster *Multimodal Computing and Interaction* (MMCI), sponsored by the German Research Foundation (DFG).

REFERENCES

1. Ravin Balakrishnan and Ken Hinckley. The role of kinesthetic reference frames in two-handed input performance. In *Proc. of UIST*, pages 171–178, New York, NY, USA, 1999. ACM.
2. Ravin Balakrishnan and Ken Hinckley. Symmetric bimanual interaction. In *Proc. of CHI*, pages 33–40, New York, NY, USA, 2000. ACM.
3. Connelly Barnes, David E. Jacobs, Jason Sanders, Dan B. Goldman, Szymon Rusinkiewicz, Adam Finkelstein, and Maneesh Agrawala. Video Puppetry: A performative interface for cutout animation. *Proc. of SIGGRAPH ASIA*, 27(5), 2008.
4. Hrvoje Benko and Steven Feiner. Balloon selection: A multi-finger technique for accurate low-fatigue 3d selections. In *Proc. IEEE Symposium on 3D User Interfaces*, 2007.
5. Hrvoje Benko, Shahram Izadi, Andrew D. Wilson, Xiang Cao, Dan Rosenfeld, and Ken Hinckley. Design and evaluation of interaction models for multi-touch mice. In *Proc. Graphics Interface*, pages 253–260, 2010.
6. Mira Dontcheva, Gary Yngve, and Zoran Popović. Layered acting for character animation. *ACM Trans. Graph.*, 22(3):409–416, 2003.
7. Clifton Forlines, Daniel Wigdor, Chia Shen, and Ravin Balakrishnan. Direct-touch vs. mouse input for tabletop displays. In *Proc. of CHI*, pages 647–656. ACM Press, 2007.
8. Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris, and Daniel Wigdor. Shadowguides: visualizations for in-situ learning of multi-touch and whole-hand gestures. In *Proc. of ITS*, pages 165–172. ACM Press, 2009.
9. Yves Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behavior*, 19:486–517, 1987.
10. Mark Hancock, Sheelagh Carpendale, and Andy Cockburn. Shallow-depth 3d interaction: design and evaluation of one-, two- and three-touch techniques. In *Proc. of CHI*, pages 1147–1156. ACM Press, 2007.
11. Mark S. Hancock, Sheelagh Carpendale, Frederic D. Vernier, Daniel Wigdor, and Chia Shen. Rotation and translation mechanisms for tabletop interactions. In *Proc. of TABLETOP*, 2006.
12. Takeo Igarashi, Tomer Moscovich, and John Forbes Hughes. Spatial keyframing for performance-driven animation. In *Proc. of SCA*, pages 107–115. ACM Press, 2005.
13. Robert J. K. Jacob, Leonidas Deligiannidis, and Stephen Morrison. A software model and specification language for non-wimp user interfaces. *ACM Trans. Comput.-Hum. Interact.*, 6(1):1–46, 1999.
14. Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and M. Preston Mullen, Jr. Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact.*, 1(1):3–26, 1994.
15. Kenrick Kin, Maneesh Agrawala, and Tony DeRose. Determining the benefits of direct-touch, bimanual, and multifinger input on a multitouch workstation. In *Proc. Graphics Interface*, pages 119–124, 2009.
16. Russell Kruger, Sheelagh Carpendale, Stacey D. Scott, and Anthony Tang. Fluid integration of rotation and translation. In *Proc. of CHI*, pages 601–610. ACM Press, 2005.
17. Michael Neff, Irene Albrecht, and Hans-Peter Seidel. Layered performance animation with correlation maps. *Computer Graphics Forum*, 26(3):675–684, 2007.
18. Tom Ngo, Doug Cutrell, Jenny Dana, Bruce Donald, Lorie Loeb, and Shunhui Zhu. Accessible animation and customizable graphics via simplicial configuration modeling. In *Proc. of SIGGRAPH*, pages 403–410. ACM Press, 2000.
19. Sageev Oore, Demetri Terzopoulos, and Geoffrey Hinton. A desktop input device and interface for interactive 3d character animation. In *Proc. Graphics Interface*, pages 133–140, 2002.
20. David Joel Sturman. *Whole-hand Input*. PhD thesis, Massachusetts Institute of Technology, 1992.
21. David Joel Sturman. Computer puppetry. *IEEE Comput. Graph. Appl.*, 18(1):38–45, 1998.
22. Frank Thomas and Ollie Johnston. *The Illusion of Life: Disney Animation*. Hyperion Press, New York, 1981.
23. Robert Y. Wang and Jovan Popović. Real-time hand-tracking with a color glove. *ACM Transactions on Graphics*, 28(3), 2009.
24. Chris Welman. Inverse kinematics and geometric constraints for articulated figure manipulation. Master's thesis, Simon Fraser University, 1993.
25. Shumin Zhai and Paul Milgram. Quantifying coordination in multiple dof movement and its application to evaluating 6 dof input devices. In *Proc. of CHI*, pages 320–327. ACM Press, 1998.