Data-driven Finger Motion Synthesis for Gesturing Characters

Sophie Jörg*
Carnegie Mellon University
Clemson University

Jessica Hodgins†
Carnegie Mellon University
Disney Research, Pittsburgh

Alla Safonova‡
Disney Research, Pittsburgh

Figure 1: Animations with synthesized finger motions: (a) ok gesture, (b)-(c) extracts from a conversation, (d) attention gesture.

Abstract

While capturing the body movements of actors to create animations for movies, games, and VR applications has become a standard practice, finger motions are usually added manually as a tedious post-processing step. In this paper, we present a surprisingly simple method to automate this step for gesturing and conversational characters. In a controlled environment, we carefully captured and post-processed the finger and body motions from multiple actors. To augment the body motions of gesturing or conversational characters with plausible and detailed finger movements, our method selects finger motion segments from the resulting database taking into account the similarity of the arm motions and the smoothness of consecutive fingers motions. We investigate which parts of an arm motion perform best at discriminating gestures with a leave-one-out cross-validation and use the result as a metric for the similarity of arm motions. Our approach provides good results in a number of examples with different gesture types and is validated in a perceptual experiment.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: finger motions, animation, gesture synthesis, hand motions, virtual characters

Links: DL PDF

* e-mail: sjoerg@clemson.edu
† e-mail:jkh@cs.cmu.edu
‡ e-mail:alla@disneyresearch.com

1 Introduction

Hand and finger motions are omnipresent in our daily life. We use our hands to interact with objects and with each other. When communicating, we use them to punctuate our speech, point to an object of interest, signal our opinion, and to convey emotions. Communication – and therefore hand and finger motions – also play an increasingly important role in digital applications. Examples are virtual worlds, such as Second Life, digital games, such as L.A. Noire, or any type of teaching, training, or advice application that uses embodied conversational agents. If we want to create believable and compelling virtual characters, we therefore need to generate convincing hand motions.

Motion capture has become a widely used technique for animating realistic, humanlike, virtual characters for movies or games. Nevertheless, the elaborate motions of the fingers with their high number of degrees of freedom and small size, are still not easy to capture. As a consequence, hands are rarely captured in their full complexity. In general, two or three markers on fingertips and two or three on the palm are used as a reference [Kitagawa and Windsor 2008] and the fingers are keyframed manually, which is a tedious, labor-intensive process.

In this paper, we propose a method to automatically add plausible finger motions to body motions. Our algorithm draws on an input body motion and a previously recorded database of body and finger motions to generate finger movements that match the input body motion (see Figure 2). We locate suitable finger animations in the database using the similarity of the wrist movements and the matching of consecutive finger motions. Our approach preserves the naturalness and subtlety of motion captured movements, without requiring the time-consuming manual post-processing. As our method is intended as a post-processing step, it does not impact the motion capture session, saving valuable production time.

In this paper, we propose a method to automatically add plausible finger motions to body motions. Our algorithm draws on an input body motion and a previously recorded database of body and finger motions to generate finger movements that match the input body motion (see Figure 2). We locate suitable finger animations in the database using the similarity of the wrist movements and the matching of consecutive finger motions. Our approach preserves the naturalness and subtlety of motion captured movements, without requiring the time-consuming manual post-processing. As our method is intended as a post-processing step, it does not impact the motion capture session, saving valuable production time.

We apply our algorithm to gesturing and conversational characters. It is not intended for object manipulations, such as grasping. Conversations, speeches, or debates are frequent situations in movies or virtual applications and the main focus of conversational agents. Gestures are an intrinsic part of language. When communicating with others, we exhibit a wide variety of gestures, which again encompass a very diverse range of finger motions. We demonstrate our approach with several example databases, consisting of gestures, casual conversations, debates, and giving directions (see Figure 1).
Figure 2: Our method automatically synthesizes finger movements for an input body motion based on a database of motions where both, finger motions and body motions, are present.

Our main contribution is the development of a data-based method to automatically add realistic finger motions with unprecedented detail to the body motions of conversational characters. We furthermore explore the validity of several parameters as predictors for the resemblance of finger motions and find that a combination of wrist rotation and wrist position leads to the best results. The rest of the paper is organized as follows: After discussing related work in Section 2, we detail our method in Section 3. In Section 4, we describe how we tested multiple finger motion predictors with a leave-one-out cross-validation applied to a set of controlled gestures. Our results, including examples with different gesture types and a perceptual study, are presented in Sections 5. We conclude by discussing the advantages and limitations of our approach and considering future work in Section 6.

2 Related Work

Several approaches have been suggested to simplify and quicken the animation of finger motions for virtual characters. They can be divided into methods to capture finger movements and algorithms to compute them. The latter ones mostly consist of data- and physics-based approaches.

Capturing finger motions is a challenge. The high number of joints in a very small space requires many small markers. Contrary to the face, where the markers are roughly located on a two-dimensional plane situated vertically in space, markers attached to fingers can point in any direction including the floor. The capture space therefore needs to be very well covered, which results in a smaller area for the same number of cameras. Frequent occlusions require time-consuming manual post-processing. Multiple glove based solutions exist [Wang and Popović 2009; CyberGlove Systems 2012; Meurand 2012], which, however, have their own drawbacks, such as a lower accuracy, difficult or time-consuming calibration, and drift [Kahlesz et al. 2004]. Computer vision algorithms can also capture finger animations [Athitsos and Sclaroff 2003] but are bound to a small, controlled area, which is suitable only for specific tasks [Wang et al. 2011] and can not yet capture full body motions. Consumer systems such as the Kinect have not reached the level of accuracy required to adequately capture finger motions across the natural human workspace.

Majkowska et al. [2006] propose a method to motion capture body motions and hand motions separately and to automatically assemble them. This approach allows for a smaller capture space when only hand motions are recorded. Nevertheless, it requires the actor to repeat the same hand motions as accurately as possible, which would be very hard for longer scenes with natural conversations without a specified choreography. Our approach, in contrast, does not require to capture the same scene twice. Ye and Liu [2012], in parallel to us, also observed that detailed finger motion can be created solely based on the wrist movements and the motion of a handled object. Their approach focuses on the challenge of manipulation strategies and therefore greatly complements our approach.

Most research on creating finger motions has focused on animating specific tasks, such as playing a musical instrument [ElKoura and Singh 2003]. Manipulation tasks have been successfully simulated using physics based approaches [Liu 2008; Liu 2009; Pollard and Zordan 2005], often using motion capture to measure parameters or forces [Kry and Pai 2006]. Neff and Seidel [2006] employ a similar method to animate relaxed hand shapes, deriving the parameters needed for the physics-based simulation from video recordings. A different solution to accurately animate hands and finger motions are detailed anatomically correct hand models [Tsang et al. 2005]. These models are usually difficult to animate correctly.

Finger motions for conversational gestures represent a different type of challenge. Their motion is not constrained by contacts to an object and the active part of their motions is not directed toward a physical goal. Gestures are considered an integral part of the act of speaking or producing an utterance [Kendon 2004]. Even though, gestures do not have a strict grammar or a fixed vocabulary, they do follow rules and have a structure that allow us to understand and interpret each other’s gestures [McNeill 1992]. Cassell et al. [2001] use a rule-based approach derived from linguistic analysis and gesture studies research to synthesize gestures based on a written text. Stone et al. [2004] animate the gestures of a conversational character with a database of recorded speech and captured motions, fitting the gestures to its generated voice. Levine et al. [2010] train a probabilistic model to create gestures based on the prosody of the voice and a data-base of gestures. Neff et al. [2008] focus on creating different styles of speaker motions with a video or text as an input. Contrary to those approaches, our method focuses on the detailed motion of the fingers. Furthermore, we do not take into account the meaning of a conversation or the rhythm in the audio track as those are already conveyed in the gestures. There is no knowledge of the prosody or utterance of a conversation. Our approach therefore can also be applied to communications when there is no active speaking involved.

The perception of finger animation has been studied by Joerg et al. [2010], who showed that even small errors in the synchronization of body and finger motions can be perceived and might change the interpretation of an animation. Hoyet et al. [2012] investigate the perception of finger animations generated with different sets of markers and find that finger motions reconstructed from a reduced set of eight markers per hand are perceived to be similar to motions captured with a set of twenty markers for many types of motions but are perceived to be different for some motions.

Our method to create finger motions searches for adequate short segments of motions in a database and finds an optimal path through the resulting motion graph. It therefore builds on the motion graph literature [Arikian and Forsyth 2002; Kovar et al. 2002; Rose et al. 1998; Lee et al. 2002; McCann and Pollard 2007].

3 Method

The goal of our method is to automatically synthesize finger motions for virtual conversational characters. With finger motions we denote the movements of the phalangeal joints of all five digits of both hands and the carpometacarpal joint of the thumb, while body motions in our case designate the motions of the body joints of the virtual character including the wrist positions and orientations but excluding the finger movements. In its strict anatomical sense the
term finger does not include the thumb [Palastanga et al. 2006]. For simplicity we use it to denote all five digits.

The inputs to our system are the body motion of a virtual gesturing or conversational character and a database of gesturing or conversational motions where both the finger and body motions are present. As an output, our approach generates the movements of the finger joints of both hands (see Figure 2).

The key observation behind our approach is that arm and finger motions are highly correlated. When we see a virtual character without hand animation, we can usually infer what the hand motions must have been. A skilled animator would be able to produce a realistic rendition of the finger motion. To produce realistic finger motions, our method therefore searches the database for body motions that are similar to the body movements of the input motion. We then adapt the associated finger motions to fit the input motion.

Arm and finger motions are correlated for many gestures as we show in Section 4. However, there is no one-to-one mapping between them. Specific arm motions can be associated with several distinct finger motions. A typical example would be rock and scissors from the common hand game Rock-paper-scissors, which have the same arm motion but a different finger motion. Our method can predict plausible, realistic looking finger motions based on the arm motions alone, but we do not aim to reproduce an exact match of the performed motion.

Our algorithm treats the right hand and the left hand independently. It consists of the following steps for each hand: (1) We segment our input motion and our database into meaningful short segments. (2) For each segment, we search for the k most similar body motion segments in our database by computing a segment cost. (3) We then evaluate the quality of the finger motion transitions for all possible combinations of consecutive segments that we previously found (transition cost). We can describe all possible resulting motions in a weighted motion graph where each node represents a segment with its associated cost and each edge corresponds to a transition with the transition cost as weight (see Figure 3). (4) Lastly, we search this graph for the shortest path and (5) process the segments to smoothen the transitions. We describe each part of the algorithm below.

### 3.1 Segmentation

To effectively search for similar body motions, we must segment our input motion and our database into meaningful fragments. A gesture consists of several phases: preparation, where the arm moves away from its rest position, pre-stroke hold, stroke, post-stroke hold, and retraction, where the arm moves back into the rest position [McNeill 1992; Kendon 2004]. Only the stroke phase is obligatory, all other phases are optional. Gestures can be combined into gesture units without returning to a rest pose between gestures.

We found that a simple approach, similar to the method described by Levine et al. [2009], works well at separating our motions into meaningful phases. We split the motion when the speed of the wrist crosses a small threshold, thus separating motion phases (high speed) and hold phases (low speed). We restrict segment length to be no less than 0.33 seconds and no more than 2 seconds. This reduces spurious segmentation due to noise and avoids too long segments.

### 3.2 Finding Similar Segments

For each segment of the input motion, we search the database for the k most similar segments. During the search, we only consider database segments whose length is comparable to the input motion segment (within 1.5 times). This avoids unnaturally fast or slow finger motions.

To compare two segments $S_I$ (the segment from the input motion) and $S_D$ (the segment from the database), we compute the segment cost $c_S$. First, we uniformly scale $S_D$ so that it has the same length as $S_I$. Next, we compute the weighted squared distance of the position and rotation coordinates of the wrist joint in the two segments according to the following formula:

$$c_S = \frac{1}{n} \left( w_P \sum_{t=1}^{n} (P_I(t) - P_D(t))^2 + w_R \sum_{t=1}^{n} (R_I(t) - R_D(t))^2 \right)$$

where $w_P$ and $w_R$ are weighting terms, which ensure that the positions and orientations have the same influence, $t = 1, \ldots, n$ is the frame number, $P_I(t)$ and $P_D(t)$ are the position coordinates of the input motion and of the fragment from the database, and $R_I(t)$ and $R_D(t)$ are the rotation coordinates of the input motion and of the fragment from the database, respectively. We use the position and rotation cost $c_R$ only if both positions and rotations are available.
global rotation coordinates relative to the root (hip) of the character. We use Euler angles for our computation but the method can be used with any angle representation.

3.3 Transition Cost

Not only do we require a good match to the arm motion segments but we also need the transitions between different finger motion segments to be smooth. One very bad or a series of mediocre transitions would make the results look unrealistic.

We compute the transition cost $c_T$ for each set of possible consecutive fragments by comparing the orientations and angular velocities of the fingers at the last frame $A$ of one fragment and the first frame $B$ of the next fragment. We again use squared differences:

$$c_T = \frac{1}{m} \left( \sum_{i=1}^{m} (J_A(i) - J_B(i))^2 + w_W \sum_{i=1}^{m} (W_A(i) - W_B(i))^2 \right)$$

where $J_A(i)$ and $J_B(i)$ are the rotations of the $i$-th degree of freedom of the finger joints in frame $A$ and frame $B$, respectively, while $W_A(i)$ and $W_B(i)$ are the corresponding angular velocities. Again, $w_J$ and $w_W$ are weighting terms to adjust the respective influences.

The finger skeleton consists of $m$ degrees of freedom, for our hand skeleton $m=25$.

3.4 Finding the Best Path

To select the final motion, we compute a weighted graph as shown in Figure 3. The start node of this graph is connected to the $k$ segments from the database that were most similar to the first input motion segment. Then, each segment is connected to the $k$ segments that were most similar to the second input motion segment. So, each row $i$ represents the $k$ best matches for input segment $i$ and each segment of one row ($i$) is connected to each segment of the following row ($i+1$).

The graph is traversed from top to bottom. A weighted sum of the corresponding transition and segment costs is applied to each connection:

$$cost = w_S * c_S + w_T * c_T$$

The weights $w_S$ and $w_T$ adjust the segment and transition costs so that they have the same influence on average.

We find the best path through the weighted graph with Dijkstra’s algorithm.

3.5 Computing the Final Motion

Even though the transition cost was taken into account when selecting the fragments, there will be differences between the finger poses of consecutive fragments. Simple blending between each fragment (every 0.33 – 2 seconds) could result in a very jittery motion. Therefore, for each transition we compute the difference between the finger rotations of the first frame of the new fragment and the last frame of the previous fragment and add this difference as an offset to the finger rotations of the new fragment. Over time this offset can increase. We perform a linear blend only when the mean angular difference or the maximum angular difference exceed a threshold ($5^\circ$ and $20^\circ$, respectively).

4 Analyzing the Similarity Function

As a skilled artist would be able to create plausible finger animations based on body motions, we suggest that it is possible to infer realistic looking hand motions from body motions, even if there might be multiple solutions with substantially different valid finger motions for a specific body motion. We want to determine to which degree finger motions can be inferred from body motions and which parts of the body are most appropriate for predicting finger motions. We suggest to test different parameters and combinations of parameters, notably the positions and orientations of the shoulders, elbows and wrists, to find out which parts of a gesture best determine a gesture domain.

Let’s consider a type of gestures, where all gestures have a very similar finger motion. Ok-gestures, where the thumb and the index form a circle and the other fingers are loosely straight, are a good example. An ok-gesture can be performed at different heights and in different directions depending on who it is addressed to (see Figure 6). Nevertheless, the course of the gesture and its finger movements are always very similar. A good similarity function would, when given an ok-gesture as an input, find other ok-gestures, amongst a larger database of gestures.

Based on this reasoning, we recorded multiple short samples of eight gesture types: attention, big, doubt, shrug, ok, palm presentation (PP), small, turn, and wipe. To decide which gesture types to capture, we annotated several minutes of conversational motions with a method similar to Kipp et al. [2007] and chose gesture types which were present multiple times in those takes and had characteristic finger motions. We recorded variations of each gesture type, ensuring that different paces, heights of the hands, and distances would be represented in each gesture type, but always keeping the typical finger motion. In the stroke phase of the big gesture, for instance, the hands were placed at different heights from abdomen to head and at different distances from close to far (see Figures 5).

We used an equal number of samples from each gesture type, to give all gestures the same influence, leading to 64 gestures in the database (8 repetitions x 8 gesture types). The gestures were performed with one hand only or with both hands, depending on how this gesture would occur naturally. Each gesture was performed in a controlled and systematic way, starting and ending in a rest position with the arms hanging relaxed at the side of the body. We segmented all gestures with the method presented in Section 3.1, typically leading to three segments: preparation, stroke and hold, and retraction. Gestures that were performed slowly might have two stroke/hold segments or two retraction segments. Gestures that were performed more quickly might be split in only two segments. The database consisted of 187 segments.
If we take one of the gestures out of our database and reconstruct it using all the remaining gestures, we can assume that the motion is correct if a gesture of the same type was selected and incorrect if other types were selected. The substantial variations within each gesture type ensure that we are not searching for mere repetitions. To determine which segments of the body are good predictors for appropriate finger motions, we thus used leave-one-out cross-validations. For each parameter, we repeat the following: We iterate over all segments in the database and predict its gesture type based on the remaining gesture segments. The prediction itself is done by finding the most similar gesture motions in the database based solely on the comparison of the parameter we evaluate, using squared differences and dynamic time warping. We compute the percentage of correctly classified segments amongst the k closest suggested segments. In our analysis k=5.

Figure 7 plots the computed probabilities for some of the body segments we tested. When each parameter was considered on its own, the rotation of the palm performed best with 75%. We also included weighted combinations of different parameters and concluded that a weighted combination of the position of the wrist and the orientation of the palm gives the best prediction result. With positions measured in cm and rotations in degrees, the best combination was found to be 0.5*rot+pos.

Not all gesture types were recognized with the same probabilities. A class confusion matrix can be seen in Figure 8. We can see that the gesture type attention was the hardest to recognize. However, even for this type 55% of the suggestions were correct. In 18% of the cases a segment from an ok gesture was suggested instead and in 14% of the cases a segment from a small gesture, which are both gestures with relatively similar arm motions. In the full algorithm, the transitions help to avoid segments with finger motions that are very different to those of the surrounding segments. Therefore, the reconstruction rate of the full motion is likely to be higher.

5 Results

To demonstrate the flexibility of our approach, we motion captured an actress and two actors with optical Vicon motion capture systems consisting of 13 to 18 cameras. In each capture, fifty-six markers were attached to the actor’s body and 22 markers to each hand. The actors were asked to perform a series of gestures and conversations. The resulting data was post-processed and a skeleton was fitted to it. For each hand and for the body, we recorded a series of movements rotating each joint across its full range of motion. We then computed the three skeletons, one for each hand and one for the body, separately to increase their accuracy. The skeletons were then assembled into one. In the following paragraphs we present a series of examples computed with different gesture databases. The resulting animations can be seen in the accompanying video.

5.1 Short Gestures

We synthesized the finger motions of several gestures from the large gesture database presented in Section 4. We chose one of the 64 gestures as an input motion and deleted its finger movements. The remaining 63 gestures formed the database that we used to synthesize new finger motions. In the video we show an ok and a doubt-shrug gesture with motion captured finger movements and with finger motion synthesized with our method as examples of correctly reconstructed motions. We also show an attention gesture as a failure case. In this example, the finger motion of the gesture’s stroke phase was selected from an ok gesture instead of an attention gesture.

In addition to the database that we presented in Section 4, we created a small gesture database using a different actor. That database consisted of seven types of motions, each captured twice in a very similar manner. Motion types were attention, count, no, ok, point, snap, and thumbs-up. In a similar manner as for the large gesture database, we use the body motion of an exemplary gesture as input motion and reconstruct its finger motion with the remaining 13 gestures as a database. The finger motion is synthesized correctly based on the second attention gesture that remained in the database.

We also show that our approach can be used across different characters, at least when the gesture types are similar. We synthesize the finger motions of the attention gesture from the small gesture database using the large gesture database. To account for differences in size between the actors, we scale the position data based on the height of each character. The motion was reconstructed seamlessly as can be seen in the video.

5.2 Conversations, Debates, and Directions

We generated three databases consisting of longer scenes: conversations, debates, and directions. Our goal was to generate different styles and types of gestures. The databases were recorded with different actors and the topics were chosen to lead to different gesture types.

First, for the conversation database, we recorded more than 13 minutes of motions of an actor conversing about multiple casual subjects, such as describing his latest holiday trip, a project he was
working on, or a movie that he recently watched. To ensure the naturalness of the motions, a second person was standing behind the cameras acting as a conversational partner. For motion synthesis, we used sequences of 2-4 minutes as input motions, leaving a database of about 10 minutes. The synthesized finger motions look very realistic, extracts can be seen in the video.

Our second database, debates, consists of 9 minutes of debates from a different actor. We felt that debating would lead to more energetic gestures. We listed a dozen of popular topics from debating clubs, such as creationism or abortion, and let the actor choose subjects. Our five takes were each between 1.5 and 2 minutes in length. To create examples, we excluded a take from the database, and synthesized its finger motions from the remaining database (see video).

Third, for the directions database, we asked an actress to describe how to get to different places, such as the airport or her home. Giving directions involves very specific gestures, for instance, for turning or stopping. Four takes, each 2-3 minutes long, resulted in over 9 minutes of data. We generated examples in the same way as for the previous databases.

5.3 Perceptual Experiment

We tested the realism of our results in a perceptual experiment. Participants were shown short clips of animations with three types of finger motions: "motion captured", "our method", and "no finger motions". The total stimuli consisted of 8 (segments) x 3 (motion types) x 3 (repetitions) = 72 clips. Five segments were chosen from the debates database and three from the large gesture database, including the incorrect reconstruction of the attention gesture shown in the video. We chose a camera perspective focusing on the characters hand space (thigh to neck) for most clips, which is an unusual shot. However, we did not want to show the face as it is not animated, which is especially distracting when the character is talking.

The experiment was fast paced, each clip being between 3 and 5 seconds long. In longer shots, a complete lack of finger motions is usually obvious. The 72 clips were shown in random order. Between each clip participants had 5 seconds to reply.

The participants were asked to pay attention to the finger motions and to rate the realism of each animation on a scale from 1 (very unrealistic) to 5 (very realistic). Six participants were students and postdocs (4m, 2f, aged 18-35) who were recruited through posters on campus. The total experiment took about 20 minutes and was rewarded with $5. Participants took part in two groups of three.

To analyze the results we averaged the three repetitions of each segment for each motion type and participant. A repeated measures ANOVA (within effects motion type and segment) with a Huynh-Feldt correction determined a significant effect of motion type at the 0.1 level: F(2,10)=5.8644, p<0.055 (without the correction p<0.021). A Newman-Keuls post-hoc analysis showed that the animations with the motion types "motion capture" and "our method" were rated significantly more realistic than segments with "no finger motions" (see Figure 9). The differences between the motion captured animations and our method were not significant.

6 Discussion and Future Work

In this paper we present a novel and surprisingly simple approach to generate detailed, lifelike finger motions for gesturing characters. Our method segments the arm movements and, for each resulting motion fragment, scans a database for the k fragments with the most similar wrist motion. It then takes into account the smoothness of the transitions between consecutive segments to compute the final motions.

Finger movements generated with our approach contain an impressive level of detail. Nevertheless, the meaning of a conversation is not taken into account. Could the generated motions therefore be confusing or convey a different connotation than desired? Most parts of an average everyday conversation or of a talk support the stress and the rhythm of a conversation and do not contain specific semantic meaning. We think that one reason our approach generates plausible results for many examples is that the timing, intonation, and even to some extent the meaning of the speech are also represented in the motions and the rhythm of the body, or more specifically of the wrist. Nevertheless, if a particular, semantically meaningful gesture is required our algorithm might produce incorrect results, especially if the wrist motion of that gestures is similar to the wrist motion of other gestures. An example would be playing rock-paper-scissors, where our algorithm could not differentiate between the three gestures as long as the wrist motion is kept the same. Our algorithm would also fail to correctly reconstruct fingerspelling in sign languages as many letters use the same wrist orientation for several letters.

On a more general scale, it is still an unsolved question how much the communication subtleties of finger motions are correlated with the body motions. Furthermore, we would like to know what the optimal way is to exploit this correlation. We showed that a weighted combination of the wrist position and the palm rotation as a predictor for similar motion segments provides the best results amongst the options tested, but many further similarity functions are conceivable. In future work, we aim to find the best combination of predictors on a varied set of gestures.

As all data based approaches, our results highly depend on the available data. A finger motion that is not present in the database can not be synthesized. Our method allows to use a database from one character to augment the body motions of a different character. Based on our experience and as retargetting issues can often be neglected for finger animation, we believe that our approach translates well across characters. Nevertheless, we do not take the stylistic details or cultural backgrounds of those characters into account. Therefore our technique might produce unexpected results if the gesture type present in the database differs too much from the input body movements.

Our method is intended as a post-processing step, which has the advantage that the motion capture session is significantly simplified because only the body needs to be captured, no calibration issues slowing down the process, and no markers or gloves decreasing the agility of the actor’s fingers. Applying our algorithm in post-production also enables the user to actively influence the results. In our algorithm, right and left hand motions are synthesized indepen-
dently to enlarge the variability of our databases. While there are
places in a conversation where both hands carry out the same or
a related motion, a wide range of gestures can be performed with the
right hand, with the left hand, or with both hands in a similar man-
er. In future implementations, we could compute each hand on its
own or both hands together depending on user input, and therefore
also support gestures that require both hands. A more challenging
undertaking would be to expand our algorithm to handle contacts,
which are currently not supported.

We applied our approach to several databases of realistic motion
captured movements. It can also handle databases created by other
means as long as the wrist motion is available. For example,
keyframed animations or stylized motions could also be used. Our
largest exemplary database consists of 13 minutes of conversational
motions. Further tests and databases are necessary to explore the
scability, the potentials and limits of our approach. Finger and
body motion databases are still labor-intensive to create and not ev-
ervybody has the resources to produce them. We hope to contribute
to future research on finger motions and gestures by making our
databases publicly available.

Acknowledgments

Thanks to everybody who contributed in the creation of the ani-
mations, namely Justin Macey for post-processing motion capture
data, Moshe Mahler and Valeria Reznitskaya for modeling the char-
acters, and Brooke for labeling gesture videos. We would also like
to thank the reviewers for their comments. This research has been
supported by the ONR DR-IRIS MURI project grant N00014-09-
1-1052 and the NSF grant IIS-1018486.

References

ARIKAN, O., AND FORSYTH, D. A. 2002. Interactive motion
generation from examples. ACM Transactions on Graphics 21,
3, 483–490.

pose from a cluttered image. In IEEE Computer Vision and Pattern
Recognition (CVPR), vol. 2, 432–439.

Beat: the behavior expression animation toolkit. In Proceedings

CYBERGLOVE SYSTEMS, 2012. CyberGlove II product
website, http://www.cyberglovesystems.com/products/cyberglove-
ii/overview, January.

EL KOURA, G., AND SINGH, K. 2003. Handrix: animating the
human hand. In Proceedings of the ACM SIGGRAPH/Eurogra-

HOYET, L., RYALL, K., MCDONNELL, R., AND O’SULLIVAN,
C. 2012. Sleight of hand: perception of finger motion from
reduced marker sets. In Proceedings of the ACM SIGGRAPH
Symposium on Interactive 3D Graphics and Games, ACM, New
York, NY, USA, I3D ’12, 79–86.

JÖRG, S., HODGINS, J., AND O’SULLIVAN, C. 2010. The percep-
tion of finger motions. In Proceedings of the 7th Symposium on
Applied Perception in Graphics and Visualization (APGV 2010),
ACM, New York, NY, USA, 129–133.

KAHLESZ, F., ZACHMANN, G., AND KLEIN, R. 2004. Visual-
fidelity dataglove calibration. In Proceedings of the Computer

KENNON, A. 2004. Gesture – Visible Action as Utterance. Cam-
bridge University Press, Cambridge, UK.

KIPP, M., NEFF, M., AND ALBRECHT, I. 2007. An annotation
scheme for conversational gestures: How to economically cap-
ture timing and form. In Journal on Language Resources and

KITAGAWA, M., AND WINDSOR, B. 2008. MoCap for Artists:
Workflow and Techniques for Motion Capture. Focal Press.

KOVAR, L., GLEICHER, M., AND PIGHIN, F. 2002. Motion

KRY, P. G., AND PAI, D. K. 2006. Interaction capture and synthe-

LEE, J., CHAI, J., REITSGA, P. S. A., HODGINS, J. K., AND
POLLARD, N. S. 2002. Interactive control of avatars animated
with human motion data. ACM Transactions on Graphics 21, 3
(July), 491–500.

LEVINE, S., THEOBALT, C., AND KOLTUN, V. 2009. Real-time
prosody-driven synthesis of body language. ACM Transactions

LEVINE, S., KRÄHENBÜHL, P., THRUN, S., AND KOLTUN, V.
2010. Gesture controllers. ACM Transactions on Graphics 29,
4, 1–11.

Proceedings of the ACM SIGGRAPH/Eurographics Symposium
on Computer Animation, 163–171.


Automatic splicing for hand and body animations. In Proceed-
ings of the ACM SIGGRAPH/Eurographics Symposium on Com-
puter Animation, 309–316.

McCANN, J., AND POLLARD, N. S. 2007. Responsive characters
from motion fragments. ACM Transactions on Graphics 26, 3
(August).

thought. The University of Chicago Press.

MEASURAND, 2012. ShapeHand product website,
www.shapehand.com/, January.

NEFF, M., AND SEIDEL, H.-P. 2006. Modeling relaxed hand
shape for character animation. In Articulated Motion and De-
formable Objects, F. Perales and R. Fisher, Eds., vol. 4069 of
Lecture Notes in Computer Science. Springer Berlin / Heidel-
berg, 262–270.

Gesture modeling and animation based on a probabilistic re-
creation of speaker style. ACM Transactions on Graphics 27,
1, 1–24.

PALASTANGA, N., FIELD, D., AND SOAMES, R. 2006. Anatomy
and Human Movement – Structure and Function, 5 ed. But-
terworth Heinemann Elsevier.

POLLARD, N. S., AND ZORDAN, V. B. 2005. Physically based
grasping control from example. In Proceedings of the ACM
SIGGRAPH/Eurographics Symposium on Computer Animation,
311–318.


