State of the Art in Hand and Finger Modeling and Animation

Nkenge Wheatland[1], Yingying Wang[2], Huaguang Song[2], Michael Neff [2], Victor Zordan [1], and Sophie Jörg [3]

[1]University of California, Riverside
[2]University of California, Davis
[3]Clemson University

Abstract

The human hand is a complex biological system able to perform numerous tasks with impressive accuracy and dexterity. Gestures furthermore play an important role in our daily interactions, and humans are particularly skilled at perceiving and interpreting detailed signals in communications. Creating believable hand motions for virtual characters is an important and challenging task. Many new methods have been proposed in the Computer Graphics community within the last years, and significant progress has been made towards creating convincing, detailed hand and finger motions. This state of the art report presents a review of the research in the area of hand and finger modeling and animation. Starting with the biological structure of the hand and its implications for how the hand moves, we discuss current methods in motion capturing hands, data-driven and physics-based algorithms to synthesize their motions, and techniques to make the appearance of the hand model surface more realistic. We then focus on areas in which detailed hand motions are crucial such as manipulation and communication. Our report concludes by describing emerging trends and applications for virtual hand animation.

Categories and Subject Descriptors (according to ACM CCS): A.1 [General]: Introductory and Survey—, I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

1. Introduction

Everyday, we use our hands and fingers to perform complex tasks. They can move with delicacy or force, executing a multitude of activities such as writing, eating, playing instruments, handling tools, and communicating (see Figure 1). Roman rhetorician Marcus Fabius Quintilianus wrote:

> As for the hands, without which all action would be crippled and enfeebled, it is scarcely possible to describe the variety of their motions, since they are almost as expressive as words. [Ken04]

We touch, pick up, hold onto, and manipulate objects with our hands and fingers. We also gesture and sign, complementing or replacing linguistic cues. This report summarizes the many research efforts aimed at synthesizing hands and fingers that appear natural as they perform the myriad of behaviors seen in their real-world counterparts.

People are keen observers of hand motion. Jörg et al. [JHO10] showed that small synchronization errors between hand and finger motions can be detected for delays as little as 0.1s and that such errors can alter the interpreta-

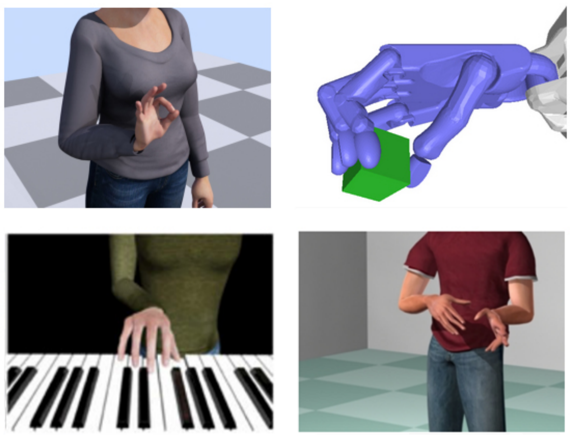

Figure 1: *Examples of hand poses synthesized for various types of motion [JHS12, BL14, ZRHN13, JHO10].*

tion of a scene. Wallbott [Wal98] showed that hand motion contributes to our perception of emotion. Gestures furthermore can convey an individual's personality [NWAW10, NTB*11], and people can be recognized based on their gesture style alone [WBH*08, NKAS08]. Careful and detailed hand animation is thus essential in the creation of convincing virtual characters.

The function of the hand follows from its remarkable structure, comprised of 27 bones, not including the sesamoid bone, in a compact space with an intricate arrangement of muscles and tendons [Nap80]. And so, this report begins with a discussion of hand anatomy and how it has been modeled and simplified in computer animation (Section 2). A diverse set of techniques have been proposed to animate said models, and we organize and highlight these next (Section 3). Specifically, the high bar for animation quality motivates the use of capture techniques to record precise movement. Unfortunately, hands are difficult to capture due in large part to frequent occlusions and changing contacts. We discuss capture technologies along with data-driven algorithms that have been developed to best take advantage of such recordings. In addition, Section 3 covers other approaches used for hand and finger modeling and animation: keyframing, procedural (including physics-based) methods, and approaches to model the hand's surface.

Due to the practical importance of hands, many application-driven techniques have been proposed which often cut across methods and offer hybrid approaches to accomplish the goals of a specific domain. We collate and summarize research in popular applications of hand animation in Section 4. Specifically, significant attention has been paid to synthesizing how hands manipulate objects and their environment, the creation of hand motion in gesture and communication, sign language animation, and motion generation for playing musical instruments.

The report concludes with a discussion of emerging trends and application domains, including virtual reality environments and interfaces. It also summarizes major trends in hand animation research and highlights the strengths and weaknesses of various approaches.

2. Virtual Hand Creation

To discuss the complexities of the many methods used to model hand and finger animations, we must begin with a review of the basic biological structure of the hand. This section describes the key anatomical elements and presents methods for modeling these elements to create virtual hands.

2.1. Anatomy

The key components that comprise the basic structure of most animation models include (a subset of) the bones of the hand and the joints that link those bones together. Naming conventions for bones and joints are adopted from anatomical systems like the one shown in Figure 2. Building upon this basic foundation, the real hand has ligaments that hold the bones and cartilage together and provide the hand skeleton's flexibility while muscles and tendons connect the bones and, through activation, create contractile forces that torque and bend the joints (Figure 3). These structures appear as abstract (simple joint torques) or more explicitly represented depending on the goals and purposes of the hand model. Further details beyond those presented here can be found in anatomy reference books or in work focused on the hands [PS12, Nap80].

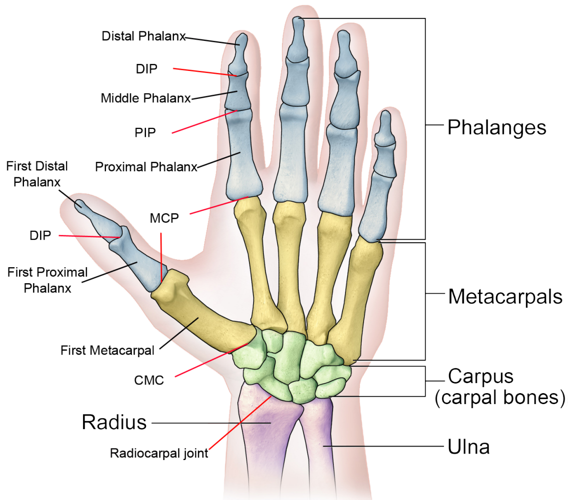

Figure 2: *The bones of the forearm, wrist, and hand [Bla14]. Acronyms: CMC – Carpometacarpal joint, MCP – Metacarpophalangeal joint, PIP – Proximal interphalangeal joint, DIP – Distal interphalangeal joint*

The dexterity of the human hand is derived from the unique configuration of bones, joints, and muscles. Namely, movement comes in the form of joint rotations: *flexion*, bending in the anterior direction (for the hand this means that the fingers form a fist); *extension*, straightening or bending in the posterior direction; *abduction*, movement away from the center of the body (the fingers are spread); and *adduction*, movement toward the center of the body (bringing the fingers together).

Anatomically, the hand has 27 bones: eight bones in the wrist or carpus, five bones in the palm called the *metacarpals*, and three in each finger and two in the thumb known as the *phalanges*. Technically, the word finger refers to digits 2-5, the index, middle, ring, and little fingers, but it is in practice (and in this publication) often used to refer to all five digits including the thumb. The cluster of bones that make up the wrist or carpus can be split into two rows where the proximal row articulates with the head of the two bones of the forearm, the *radius* and the *ulna*, at the *radio-*

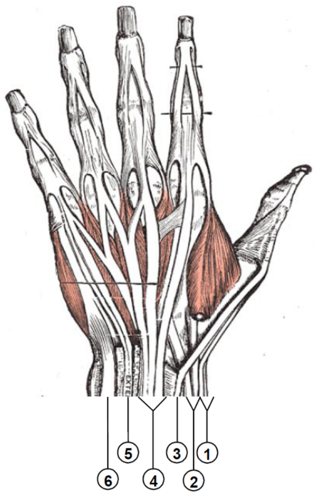

Figure 3: *The muscles of the forearm, wrist, and hand numbers by compartment: 1 – Abductor pollicis longus, Extensor pollicis brevis; 2 – Extensor carpi radialis longus, Extensor carpi radialis brevis; 3 – Extensor pollicis longus; 4 – Extensor indicis, Extensor digitorum communis; 5 – Extensor digiti minimi; 6 – Extensor carpi ulnaris [GL18]*

carpal joint while the distal row articulates with the base of the metacarpals at the *carpometacarpal joints* (CMC). The distal phalanx of the thumb opposes that of the other four fingers. This opposition plays a crucial role in human's ability to perform grasping motions and in dextrous manipulation in general and is rendered possible by the shape of the trapezium, the carpal bone which articulates at the CMC joint with the metacarpal of the thumb. The four fingers have three phalanges, proximal, middle, and distal, while the thumb only has a proximal and distal phalanx. The four fingers can articulate at their three joints: the *metacarpophalangeal joints* (MCP) between the metacarpals and the proximal phalanges, the *proximal interphalangeal joints* (PIP) between the proximal and middle phalanges, and the *distal interphalangeal joints* (DIP) between the middle and distal phalanges. Because the thumb has no middle phalanx, it can only articulate at its MCP and DIP joints. The PIP and DIP joints act primarily as hinge joints and perform flexion/extension and can hyperextend to a small degree. The MCP joints are more mobile and can also perform adduction and abduction and experience *medial* (internal) and *lateral* (external) rotation. Finally, the cupping of the palm, called the palmar arch, occurs between the CMC and MCP joints of the fingers, particularly those of the thumb, ring, and little fingers.

The musclotendon systems in the hand are among the most complex in the body, with connections across several bones in the hand driven by contraction in the forearm. Further, the movement of the palm and fingers is directly related to the flexion/extension and abduction/adduction of the wrist. For example, strong grip is achieved when the wrist is in a neutral pose [PS12]. The muscles that flex and extend the thumb are separate from the muscles responsible for flexing and extending the digits. The *extensor digitorum communis*, a dominant muscle for digit movement, contributes to the coordinated way in which some of our fingers move [PS12]. The index finger has a separate extensor (*extensor indicis*) and the little finger a separate flexor and extensor (*extensor digiti minimi*). These separate muscles give these digits more independence in contrast to the other fingers.

2.2. Hand model representations

Animation researchers have proposed various degrees of complexity for hand models that are appropriate for the needs of various problem domains. Clearly, building a high-resolution anatomical model can be overly complicated and is too computationally expensive for its utility in many applications. Thus, in general, simplifications are made in order to keep models only as complicated as need dictates.

Hands are most often modeled as a relatively small group of articulated rigid links, where the rigid bodies represent the bones of the digits, palm, and sometimes the forearm. While there are researchers that aim at greater anatomical detail [AHS03, TSF05, SKP08], common skeletons usually contain a reduced set of bones. For example, while Tsang et al. [TSF05] build a geometric model that accurately contains all 27 bones of the hand with 2 bones for the forearm, their articulation model only has 16 joints. This choice is also supported by the real anatomy, as many bones of the carpus and the intercarpal joints for example, can be ignored because their movement is considered negligible [AHS03].

Other standard simplifications also appear for the choice of joint degrees of freedom (DOFs). Minimizing the DOFs reduces the joint space which can be helpful in various ways, for example, in physics control problems and optimization or search by limiting the solution space. It is very common to simplify finger joints such as the DIP and PIP joints with single DOF hinge joints. The MCPs are condyloid joints, consisting of the rounded ends of the metacarpals and the concave base of the phalanges. These joints afford all movement except axial (i.e., rotation around its own axis). While there are exceptions to all cases, MCPs are often simplified in virtual hands as universal joints that yield two DOFs (flexion/extension, abduction/adduction). Some models instead take the MCP joints' medial and lateral rotations into account and depict them as ball joints with three DOFs. The CMC joint of the thumb is sometimes represented as having three DOFs because of its ability to perform opposition, depending on the choice of palm model.

A greater amount of variability appears across models in the choice of palm. While it is at times represented as a single rigid body with DOFs added to other joints to compensate [ICL00, MTA*01], a rigid palm is particularly poor for manipulation and grasping tasks. With additional DOFs the palm is able to bend, for example, to create cupping behaviors. Thus for applications involving grasping, two to four links are commonly employed to represent the palm, with the multiple rigid bodies connected by varying joints and/or constraints [PZ05, Liu08, AK13].

Finally, the combination of the radiocarpal, midcarpal, and carpometacarpal joints give the human wrist a wide range of motion, performing flexion/extension, abduction/adduction, and some twisting. At times, the wrist is modeled with six DOFs, to account for rotations as well as translation [ES03]. Other virtual wrist/forearm models either maintain anatomical articulation [TSF05, ZCX12, ZZMC13] or simplify the wrist to fewer DOFs [Liu08, TSF05]. An example skeletal model of the hand is shown in Figure 4.

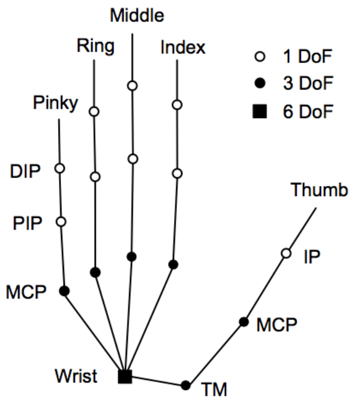

Figure 4: *Skeletal model used by Zhao et al. [ZCX12, ZZMC13] with the degrees of freedom of each joint.*

2.3. Personalized hand skeletons

Personalized skeletal structures closely replicate the kinematic structure of an individual's hand, e.g., the specific locations and orientations of joints. These skeletal models can more precisely reproduce the individual's motions over generic models with general joint position estimations. To create such subject-specific models, processes for determining the centers of rotation (CoR) and the axes of rotation (AoR) for each of the joints have been developed [CLPF05, CP07a, CP07b]. Without a user-specific hand model, data mapping can become problematic, because of fitting (and other) problems associated with using capture systems. For example, small off-axis motion as well as soft tissue deformation can lead to measurement inaccuracies that result in poor mapping of movement for generic hand models (see Section 3.2 for more detail).

Chang and Pollard [CP07a, CP07b] propose methods to determine both the CoR and AoR for joints in the thumb and index finger. Their approach to determining the CoR in the CMC joint of the thumb uses a sphere-fitting method extended to be used with multiple markers and a constrained least-squares cost function. In their method, the CoR of the CMC joint of the thumb is found as the center of the sphere that is common to the trajectories of select markers. Chang and Pollard also use this method and plane-fitting to determine the direction of the AoR [CP07b]. They furthermore propose a technique to identify the directions of the rotational axes of the CMC joint of the thumb [CP08] and determine the range of motion for rotations in individual subjects.

Medical imaging data has also been used as a technique to create specific skeletal hand models. Miyata et al. [MKMK05] take multiple images of the hand in varying poses to create 3D reconstructions and perform measurements of the skeletal structure to produce an accurate kinematic model. Stillfried et al. [SHSvdS14] compare the hand model they create using MRI data to a hand model created using optical motion capture data. They find that, if they fit the same number of DOF to the two different models, there are no significant differences in finger joint translation and rotation in multi-joint hand movement.

2.4. Dimensionality and Redundancy

The hand moves in particular ways due to its anatomy, and while its DOFs create affordances for complex movement, the hand's motion is structured in a manner that suggests order. For example, Somia et al. [SRWG98] found, along with a list of other relationships, that 83% of finger flexion and 80% of finger extension begins in a specific joint (the DIP for the index, middle, and ring fingers and the PIP for the little finger). While this is not surprising given the anatomical structure of the hand (for a discussion see [Zan79]), it suggests that reduction in the complexity of the hand is possible.

Indeed, the motions of the hand have considerable redundancy and reducing the degrees of freedom of the hand simplifies its animation. In an early paper on finger animation, Rijpkema and Girard [RG91] propose the following relationship between the distal and proximal interphalangeal joints:

$$\theta_{DIP} = \frac{2}{3}\theta_{PIP} \tag{1}$$

This equation has been used by several researchers to simplify their animation models [ST94, LWH00, HRMO12a].

Since, many researchers have used techniques to explore and exploit redundancy in hand movement. For example, principal component analysis (PCA) has been shown a valuable technique for studying lower dimensional representations of hand motion [SFS98, BZ04, CGA07]. Braido and Zhang [BZ04] explore finger coordination in both grasping a cylinder, where all four fingers flex at the same time, and in individual finger flexion, where each finger is flexed one at a time. Through PCA, they find that the first two component dimensions explain 98% of the variance in the recorded motion. In another study, Santello et al. [SFS98] conclude that the first two components of PCA from a set of grasping poses account for over 80% of the variance. PCA has also been used to reduce features for recognition and capture techniques [CPMX07, WJZ13].

Jörg et al. [JO09] use a distance metric to study correlations between the different DOFs of the hand. Their approach analyzes which joint rotation DOFs are irrelevant and which are redundant based on motion captured finger motions. To determine irrelevance, they find joints whose rotation ranges are below certain thresholds. They find that out of 50 possible joint rotation curves in two hands, the ranges of 19 are below a threshold of 5°, out of which 11 are below a threshold of 1°. A rotation of 5° is small and a rotation of 1° is barely noticeable. To find redundancy, they examine the root mean squared deviations between pairs of standardized joint rotation curves to determine how accurately one rotation curve can be expressed as a linear transformation of another one. Their results suggest that hand models can be reduced from 50 to 15 DOFs for both hands combined without loosing valuable information.

Hoyet et al. [HRMO12b] investigate the perceived fidelity of finger motions captured with different reduced marker sets. They find that movements captured with a set of eight markers per hand, one on each fingertip, two on the palm, and one on the thumb's CMC joint, is sufficient to be perceived as very similar to movements captured with a set of twenty markers. They recommend to use such a reduced marker set and to reconstruct the motion using inverse kinematics in situations where the accurate finger curvature is not crucial.

3. Animation Techniques

Three main techniques are commonly used in computer animation to create motions: keyframing, motion capture and data-driven methods, and rule-based techniques, which, for example, include physics-based animation and animations based on behavioral rules. These techniques are also combined and hybrid approaches are proposed to provide the desired results. We will define each technique and give an overview of the research that employs them.

3.1. Keyframing

Mimicking a traditional approach used in hand drawn animation, keyframing is one of the earliest methods used in both 2D and 3D computer animation, dating back at least as far as Burtnyk and Wein's use in the early 1970s [BW71, BW76]. The process allows an animator to create motion by specifying the value of attributes (e.g., joint angles) at particular *keyframes*, or points in time. The values of the attributes are then interpolated over time to create continuous animation. Various interpolation techniques give animators control over the resulting animation [Ree81, KB84b, KB84a].

Keyframing allows for the creation of both very realistic and highly stylized motions. It provides excellent control, but can be labor intensive and relies on the skill of the animator to generate quality motion. Keyframing is a convenient abstraction that is used in algorithmic approaches for animation. For example, Neff et al. [NKAS08] use keyframes to represent both hand shape and gesture. Since many gestures have a similar structure (preparation, stroke, and hold phases, see Section 4.2), this approach can lead to a small number of required keyframes. Ip et al. [ICL00] also produce natural 3D hand gesture animations using an anatomical hand model and a series of keyframes driven by static image poses to key the 3D model. In contrast, Shankar et al. [SG06] use actual 2D images as keyframes and interpolate between them to produce 2D gesture animation.

Adamo-Villani [AV08] present a study that examines whether keyframing or motion capture is preferred for animating signing avatars. Twenty clips (10 keyframed, 10 motion captured) are rated by 71 participants. The results show that the keyframed animations are more accurate and legible to the viewers. While the authors give various reasons, mostly related to the limits of motion capture technology, it is also possible that the editing and refinement process of keyframing is responsible for the findings. Jörg et al. [JHO10] on the contrary find that the quality of motion captured finger movements is considered higher than for keyframed animations. However, their study, based on 4 clips in each condition (motion captured, keyframed, random, and no animation) and 24 participants, restricts the number of allowed keyframes and hand poses to account for an animator's limited amount of time. When time is not limited, any type of animation can be created with keyframing. The quality of the result then solely depends on the skills of the animator.

Beyond keyframing, many approaches have been proposed to facilitate the process of animating hands and fingers. These methods rely less on an artist's skills but use algorithms and/or data to synthesize motion. In the remainder of this section we describe such methods, beginning with methods to motion capture hands and fingers, then describing data-driven techniques, and finally giving an overview of physics and rule-based algorithms.

3.2. Motion Capturing Hands

Finger data can be obtained through various forms of motion capture, including marker-based optical, video tracking systems, RGB-Depth (RGB-D) sensors, gloves, and tactile sensors. Menache provides a good overview of common techniques [Men99]. Below, we summarize the main approaches along with recent advances. A comparison of the basic motion capture technologies can be found in Table 1.

Capture Technology	Accuracy	Sources of Error	Capture Volume	Main Advantage	Cost in Money and Time
Marker-based optical	Excellent, although skeleton reconstruction introduces some error	Occlusions, especially for complex handshapes, and marker mislabelings	Small with full marker set, large with reduced set	Accuracy	Expensive in $, time intensive marker attachment and post-processing
Bend-sensor gloves	No spatial position measured, some calibration techniques target finger separation, others just general hand shape, accuracy may be lower than for marker-based optical systems	Cross-coupling between sensors, misalignment of sensors and joints, fewer sensors than hand DOFs	Large	No occlusions, even in large capture volume or for complex hand shapes	Moderate to high in $, calibration can be time consuming, reconstruction is fast
Markerless Optical	Depends on hand shape, better at capturing silhouettes, complex hand shapes are difficult to reconstruct	Occlusions and inaccurate depth estimates	Small	Easy and quick setup, cheap	Cheap in $
Depth Camera	Depends on hand shape, better at capturing silhouettes, complex hand shapes are difficult to reconstruct	Occlusions and sensor noise	Small	Easy and quick setup, cheap	Cheap to moderate in $

Table 1: *Comparison of motion capture technologies for recording hand motions.*

Optical marker-based motion capture. Optical motion capture has become an industry standard for acquiring motion intended for character animation. It allows for the acquisition of natural motion directly from an actor. Marker-based optical motion capture performs triangulation using cameras in order to track the 3D location of markers attached to an actor's body. Generally, an IK problem is then solved to fit a skeleton to these tracked data points and the joint angles of the skeleton can be used to animate a character. A typical system has 4 to 32 cameras that can record between 30 and 2000 samples per second [KW08]. Commercial marker-based optical motion capture systems and companies selling them include Vicon [Vic14], NaturalPoint's OptiTrack [Opt14], Qualisys [Qua14], and PhaseSpace [Pha14].

Marker-based optical motion capture offers excellent positional accuracy if the cameras are correctly calibrated and have a clear view of the markers. It can support a large capture space for full body capture, which permits actors to move freely and multiple subjects to be captured simultaneously. When applied to fingers, marker-based approaches often require a much smaller capture volume. Fingers are small and have a large number of degrees of freedom, requiring many small markers to be placed close to one another; usually 13-20 for a high quality capture. This includes two or more markers on each finger and at least three on the back of the hand [KW08]. An example marker configuration can be seen in Figure 5. In a large space, cameras may not be able to discern these markers, and it is difficult to place sufficient cameras to avoid occlusion, for example when the performer turns the palms up. These problems are alleviated in a small volume, where cameras are brought in close to the actor's hands to capture the motion, isolating the hand motion from that of the full body. Occlusion remains a problem, however, if the actor, for example, curls his fingers to make a fist or performs certain sign language signs. Occlusion is also possible if there are other physical objects in the capture volume, especially if the actor is interacting with them. A substantial amount of post-processing is generally needed to clean the data, addressing marker occlusion and mislabeling.

Figure 5: *Hands outfitted with a fairly comprehensive marker set for optical motion capture. Further markers could be added to capture the motion of additional joints such as the CMC joint.*

Researchers have explored methods for addressing these limitations. A common approach to achieve both full body capture and hand capture is the use of a reduced marker set [CH05, CPMX07, HRMO12a, KWNZ12, WJZ13], which allows for more marker separation and will allow the system to better identify markers correctly [KW08].

Glove-based motion capture. Glove-based systems provide an alternative capture technology. Gloves became popular in the late 1980s as a way for humans to interact with virtual environments, allowing for gesture input that uses the entire hand [SZP89]. Gloves also enable manipulation of objects in virtual environments [FMHR87, WG89]. The MIT-LED glove was one of the first gloves specifically made for tracking the motion of the hand for computer animation [SZ94]. Sturman and Zeltzer [SZ94] and DiPietro et al. [DSD08] have both presented surveys on the different available glove technologies and their applications.

This section will focus on gloves with bend sensors – "sensored gloves" – as they are prevalent in current hand animation research. These gloves feature attached sensors that directly measure hand and finger joint angles. Thomas G. Zimmerman created what is recognized as the first sensored glove in 1982 [DSD08]. The glove used an optical, flex-mounted sensor to measure the bends in fingers [Zim85]. Current gloves are often made of Lycra and the sensors are sewn onto the fabric. Some current sensored glove brands include CyberGlove Systems [Cyb14], DGTech Engineering Solutions [DGT14], Fifth Dimension Technologies (5DT) [Fif14], and Measurand [Mea14]. A pair of CyberGloves is shown in Figure 6. The gloves use different sensors and have different designs and sensor configurations. As a result, some may be better at performing certain tasks than others. Many of the different designs are explained by [Men99] and [DSD08]. The CyberGlove has piezoresistive sensors that convert joint angles into voltages. By contrast, 5DT's Data Glove uses optical-fiber flexor sensors with LED lights attached to one end. When light is returned to the phototransistor on the other end, the intensity of the returned light acts as a measurement for how much a joint it bending [DSD08].

Common design specifications for sensor placement include sensors measuring the following motions:

- flexion/extension of each finger's DIP, PIP, and MCP joints
- flexion/extension of the thumb's IP, MP, and MCP joints
- abduction/adduction of each finger
- wrist flexion and abduction/adduction
- the arch of the palm

Gloves have been used in a range of applications with different accuracy requirements, including sign language [HL10a, LH09], gesture [HEK*07, JLC10], virtual environment interaction [KZK04, MBM*03], robotic tele-operation and object manipulation [FvdSH98, GFTC00, HGL*04]. Sensored gloves are appealing because they can be used in a large space or outdoors, avoid the major problem of occlusion, and are a natural interface for hand data capture. Unfortunately, many gloves also suffer from problems of sensor cross-coupling, where a movement may bend multiple sensors, including some sensors intended to measure a different motion, noise and, to a lesser degree, sensor nonlinearity. As a result, their joint angle accuracy may not be high enough for a detailed finger capture [KZK04]. The gloves need to be accurately calibrated to capture data for each subject and

Figure 6: *A pair of CyberGloves, sensored gloves made by CyberGlove Systems.*

this calibration process may need to be repeated often, for example, between wearings.

Glove Calibration: Most data gloves use linear sensors. The CyberGlove II user guide [Yaz09] claims sensor resolution is approximately 1° with a maximum nonlinearity of 0.6%, with other research [Cha01] reporting mean measurement errors of $1.7^\circ \pm 1.5^\circ$, and better linearity for flexion sensors than for abduction sensors [KHW95]. Further work provides more details on linearity [QWAB89, WGS*90]. Sensored gloves ideally can be calibrated through a linear function mapping sensor input to joint rotation data. The default calibration application provided by CyberGlove [Yaz09] uses two poses, "flat" and "okay", to try to fit a linear function to each DOF. The poses do not involve enough reading changes to calibrate all sensors however, and a manual parameter tuning process is required. This basic two pose method is extended by Huenerfauth and Lu as they develop an efficient and accessible calibration protocol for deaf subjects performing ASL [HL10a, LH09]. They require subjects to perform a minimum set of pre-designed "sign language" poses in proper order, so that each sensor can have two different readings to calibrate its linear mapping function. Data from multiple samplings are used to improve error from thumb cross-coupling in a manual process.

The idealized model of one sensor recording the movement of only one degree of freedom is not accurate in practice, as a single rotation will often change the readings of multiple sensors. This cross-coupling is most frequently reported for the abduction sensors [CGK00, JLC10, SMR11], with some reporting that it could account for 62.4% to 148.3% of the abduction active range [KZK04] and others putting this at 22.7% to 45.7% [WN13a]. Thumb cross-coupling is more complicated as the sensors are not stitched orthogonally to measure the joint rotation. Wang and Neff [WN13a] find that thumb roll could cause 25.4% variance in the thumb abduction sensor readings, a problem also reported in further research [CGK00, SMR11]. Due to cross-coupling, it is not possible to obtain accurate results with a calibration system that always maps one sensor to one joint angle (e.g. using independent linear regression [MBM*03]). For certain joint angles, readings from multiple sensors must be used.

Multiple approaches address cross coupling [KZK04, CGK00, JLC10, SMR11, WN13a]. Kahlesz et al. [KZK04] regard the calibration function for abduction sensors as a density function in 3D space, which takes readings from abduction sensor s_{ABD} together with two neighboring flexion sensors $s_{MCP\_LEFT}, s_{MCP\_RIGHT}$ as inputs. Ground truth data is measured using widgets, but only for the zero-abduction angle. Similarly, Jin et al. collect ground truth data by using fixed widgets and simulate the calibration function for cross-coupled sensors by using natural neighbor interpolation of the collected data points [JLC10]. Chou et al. use a marker based vision system as part of the calibration process [CGK00]. A monocular camera is used to automatically retrieve ground truth by measuring the actual joint rotation and link length as the subject performs a set of hand poses. Multilinear regression is performed to solve for the calibration function for the cross-coupled abduction sensors. Calibration accuracy is restricted by the linear model. Steffen et al. [SMR11] propose different calibration techniques for different cross-coupling types. For finger abduction sensors, they use a zero-abduction surface with parabolic form. For thumb abduction, a parallelogram with non-orthogonal axes is used to bound all the data points. The third cross-coupling effect concerns the fingers' absolute abduction angle, and the proposed method empirically interpolates between all the abduction sensor readings, which allows side-movements of the middle finger.

For gesture or sign language, hand shape accuracy (forward kinematics) may be important while for object manipulation and robotic tele-operation [GFTC00, HGL*04], accuracy of fingertip positions (inverse kinematics) plays a key role. Some calibration approaches have explicitly sought to guarantee fingertip accuracy. Unlike general handshape, fingertip positions depend on the lengths of the fingers.

Griffin et al. [GFTC00] develop individual hand models for each user. They use a linear calibration model for each DOF that includes gain and offset parameters and treat link lengths as additional parameters. During calibration, the subject keeps the pinky and thumb in contact. A Jacobian matrix is used in an optimization process that solves for all model parameters. The system achieves 5.26mm $\pm$1.4mm error. In a related approach, Hu et al. [HGL*04] use two stereo cameras to track actual fingertip positions and a constrained IK solver is used to minimize the errors between the mapped fingertip positions and the measured fingertip positions, with respect to the gain and offset of each sensor calibration function. They report error of less than 5mm. Fischer et al. [FvdSH98] collect ground truth data using a vision tracking system. A feed-forward neural network is constructed for their IK calibration. Their evaluation shows that the resulting position error is typically 0.5mm.

Seeking to provide both accurate hand shape and accurate finger touching, Wang and Neff [WN13a] propose a more flexible calibration by using a linear mean composite Gaussian process regression (LMC-GPR) model. LMC-GPR maps sensor readings to ideal joint rotations in a non-linear way. During training, it uses input from a widget-aided sampling process. Compared to linear calibration, the method can help fix cross-coupling problems and provide better fingertip accuracy for finger touching poses.

Most data gloves have fewer sensors than the number of DOFs on a real human hand. Based on biomechanical observations, the missing finger DIP rotations can be synthesized as proportional to the neighboring PIP rotations [CGK00, WN13a, JLC10] as shown in Equation 1. Jin and Chen [JLC10] also propose a related function to simulate thumb joint rotations when calibrating the 14 sensor 5DT Data Glove.

Recently, alternative glove technologies have emerged that utilize small inertial sensors to track hand and finger motion. Examples include the Synertial's IGS-Gloves [Syn15] and the gloves included in a system from two recently funded Kickstarter projects called Control VR [Con15] and Perception Neuron [Neu15]. Inertial sensors measure the rate of change in orientation or velocity. A limitation is that to calculate position and orientation accurately the output of all of the sensors must be unified and integrated over time [DSD08]. As these systems are in development, research will have to show how they compare to other finger capture systems.

Image and depth sensor based motion capture. Some optical systems use gloves without specific sensors to help track the hand. For example, the Color Glove is a simple cloth glove that has 21 equally sized patches of color [WP09]. The different colors are used to help make each pose distinctive when solving the identification and pose estimation problem. The creation of this glove was inspired by previous works that used color coding to identify different regions of the hand. Dorner [Dor94] used a glove with colored rings on the joints to act as markers that can be easily detected by a video camera. Her gloves were used to detect signs from the ASL alphabet. Theobalt et al. [TAH*04] placed a glove with colored markers on a baseball pitcher's hand and tracked his hand motion using low-cost cameras and a stroboscope.

Recently, new technologies have emerged that can perform hand tracking and detailed hand capture optically in smaller spaces without the use of markers or gloves. These systems commonly use silhouettes or edge detection to determine hand poses.

Wang et al. [WPP11] presents a system that uses two regular video cameras to track hands and identify gesture poses.

The cameras are suspended above the hands. In their system, hands manipulate virtual objects, including the virtual camera in the scene. The system recognizes a select set of hand gestures and uses a pose estimation scheme to identify these gestures from the camera input in real time.

Depth sensor systems have recently been deployed to record human motion. Technologies, such as the *Microsoft* Kinect, have proven capable of full body tracking [SFC*11]. An RGB-D camera like the Kinect captures both color and depth information. Leap Motion [Lea14] has also created a system for free hand tracking. It differs from the Kinect in that it only captures depth information and the motion receiver/controller lies flat on a surface below the hands. Further systems have been developed based on similar techniques [3Ge14, Nim15] even if not all of them have been commercialized. More complex hand configurations with self-occlusions are difficult to accurately recognize with this type of systems. To address this, Tompson et al. [TSLP14] propose a system to accurately recover poses recorded using depth cameras. Their method uses a convolutional neural network to detect the locations of certain 2D and 3D features of the model in real time. The feature locations are used in an inverse kinematics algorithm to retrieve the pose of the hand. Shridar et al. [SOT13] use a different approach to recover poses from depth sensor data. They first filter the data to extract the hand and then perform principal component analysis to resolve the orientation of the palm. A linear support vector machine classifier is used to classify the locations of the fingertips. A pose estimation algorithm is then used to match the fingertip locations to poses in a database with similar fingertip locations.

A downside to these markerless and gloveless systems is that they again require a small capture space. Hand motion must be isolated and free range motion is restricted. To process the recorded data and perform accurate tracking and pose estimation, many approaches employ complex algorithms that are computationally expensive and cannot be run in real time [SMFW04, dLGPF08]. Other approaches provide real time speeds at the expense of resolution, hand pose detail, or the requirement of larger sets of prerecorded gestures [SKSK07, DMR06].

A strategy for improving motion capture systems is to combine approaches. A drawback of RGB-D camera data is that it is often noisy. To solve this problem, Zhao et al. [ZCX12] capture high-quality hand shape data by using a twelve camera Vicon system and a *Microsoft* Kinect, which records RGB-D image data. With this setup they build a large library of poses. The results from the two capture systems are combined by extracting a silhouette of the hand pose from the RGB-D image and then solving a cost function that analyzes how well an input pose matches the data recorded by the two systems. This combined approach allows to capture a multitude of hand poses and provides better results than when only using one capture technique.

3.3. Data-driven Methods

The challenges and significant time to create finger motion notwithstanding, accurately captured finger motions are very convincing and exhibit a high degree of realism. Data-driven techniques provide methods for synthesizing new movements using previously recorded or created motion of any style. They allow for the re-use of motion data, adapting it to new situations.

Data-driven methods have been used to solve a range of problems, such as simultaneously capturing full body and detailed hand movement, synthesizing gestures for conversational characters, or computing parameters for procedural algorithms. Many approaches employ or are inspired by existing data-driven animation methods, for example, dynamic time warping (DTW) and motion graphs, or common data reduction or machine learning models, such as principal component analysis and hidden Markov models, and specifically adapt them to the creation of finger motions or gestures.

Dynamic time warping (DTW) is used to compare two temporal signals or to adapt the timing of one signal to another [BW95]. Majkowska et al. [MZF06] present a technique that relies on DTW to capture detailed finger and body motions. As finger and body motions are difficult to capture simultaneously due to differences in the sizes of the motions and markers, the authors suggest capturing the motion of the body and the hands in two separate sessions, recording the detailed finger motions in a smaller area where the performer remains standing or seated. The positions from four markers on the hand, wrist, and forearm are included in both captures, which allows for a later alignment of the hand and body motions in their three step algorithm. First, movement phases (preparation, stroke, hold, and retraction, further explained in Section 4.2) are matched using DTW based on acceleration and velocity profiles. Then, again with DTW, the frames within the matched phases are aligned to the frames of the full body motion. Finally, the resulting motions are smoothed to fit together seamlessly (see Figure 7).

A class of techniques rely on motion databases. For example, to create finger motions for arbitrary new sequences of body motions, an option is to use a database in which both detailed finger motions and body movements are present. An inherent limitation of this type of approach is that only finger motions that are available in the database can be created and that there is no guarantee that the resulting finger motions correspond to the movements intended by the performer. However, it has the advantage that such a database only needs to be captured once and can then be reused as often as needed.

When separate hand and body databases are used, the challenge is to select and combine the best matching finger motion segments from the database. Jörg et al. [JHS12] use a database to augment the body motions of gesturing virtual characters with plausible, high-quality finger motions. They find that, amongst the tested variables, the best predictor for

Figure 7: *Example of full body animation with detailed hand motion from the splicing method proposed by Majkowska et al. [MZF06].*

consistent finger motions is a combination of the wrist position and rotation. Once the body motion and the database are segmented into phases, the combination of wrist position and orientation is used to select the *k* best matching finger motion segments from the database for each motion segment, adapting shorter and longer segments using DTW. The final sequence of movements is determined by first creating a graph weighted by how well finger and body segments match and how well consecutive finger motions blend into each other and then finding the shortest path through it with Dijkstra's algorithm.

Many further methods use databases as a starting point. Stone et al.'s [SDO*04] database consists of prerecorded speech and arm motions. Based on linguistic and behavioral rules they design a motion graph and find a path through it minimizing an objective function that scores how well adjacent elements match. The result is an animated conversational character with speech and gestures. They also use a time warping approach to fit the motions to the different speech utterances. Levine et al. [LTK09] synthesize the arm motions of conversational characters using speech as input. Their approach uses prerecorded motion capture and audio data of conversations to train the model. Animations are produced by selecting motions from the training based on prosody cues in a live speech signal. A specialized hidden Markov model (HMM) is used to perform the selection and ensure smooth transitions between movements. This method allows the authors to generate hand and body motions for arbitrary audio input provided by a microphone in real time. In further work, Levine et al. create a two layer system to model the connection between prosody and gesture kinematics [LKTK10]. The first layer, the inference layer, infers a belief distribution over a set of states that represent the kinematics of the motion from a training database. The control layer then selects the appropriate gestures based on the inferred distribution. They found that animations generated using this method are preferred over animations generated using the HMM approach.

Other researchers take advantage of the redundancy in hand motions and combine databases with reduced marker sets to synthesize motion. Kang et al. [KWNZ12] and Wheatland et al. [WJZ13] both use a reduced marker set on the hands to capture the hand and finger motion and then use a reference database to reconstruct finger motions for the final animation. The databases contain prerecorded high resolution finger motion similar to the motion being reconstructed, and synthesis is performed by finding the pose in the database that most resembles a low resolution input pose. Wheatland et al. [WJZ13] use principal component analysis (PCA) to select a sparse marker set and to build a regression model. For reconstruction, input marker positions from the reduced set are mapped to the joint angles of the hand through the computed PCA in order to produce the full-resolution hand signs as output.

Data-driven approaches have also been used with glove-based input. Wang and Popović [WP09] propose a system that tracks hand motions in real-time using a glove with a distinctive, colored pattern. For their method, shown in Figure 8, a pose database is built with a large set of prerecorded 3D hand poses and then is sampled to encompass the full hand pose space. A nearest-neighbor algorithm is employed to search for poses in the database that are similar to the query input from the glove, and the most similar poses chosen are blended together to get the estimated result.

Rather than reconstructing motions, some techniques aim to extract particular features from hand data to classify and identify the input [DSD08]. Markov models and neural networks have been used to classify input in multiple gesture recognition systems [MT91, LO96, DM02, MK02]. Using a CyberGlove, Weissmann and Salomon [WS99] explore the question of how to map the angular measurements received from sensored gloves to predefined hand gesture poses. To this aim, they test the performance of different neural network models on set poses. Using training sets comprised of 200 different hand poses, they find that a simply trained back propagation neural network classifies their set of gestures better than a radial basis function neural network. Plancak and Luzanin [PL14] use a low-budget glove, the 5DT Data Glove 5 Ultra, and train a probabilistic neural network to recognize gestures of fully open or fully closed hands. Their method uses clustering algorithms to reduce the training data size and allow for shorter execution times without significant loss in training quality.

Finger motion data is also used as an input to drive animation and several researchers have employed it to animate objects other than hands. Using data-driven approaches and approaches combining glove recordings and simulation, controllers have been developed, for example, to animate biped characters using hand or glove input [DM02, WP09, IIO10, LS12].

One general drawback of data-driven methods is their lack of adaptability to different situations. The smaller the collection of prerecorded motion, the more limiting a pure data-driven approach is. This problem can be solved by adding simulation to the approach. The combination of motion capture data and simulation allows the data to be augmented with a physical model and adapted to new situations. An example is the work of Kry and Pai [KP06], who synthesize hands interacting with different objects. They use motion capture data as a reference motion and add a simulation to generate new hand motions. Ye and Liu [YL12] add detailed manipulation and grasping motion to a full body character by using an algorithm that determines the best hand shape to use based on a set of hand-object contact positions. Inputs to the system include motion capture data of an actor's body, including the movement of the wrist, and the motion of each object that is manipulated by the actor. Multiple contact positions are sampled to find a hand shape that can be reached from the hand's current shape and can match the motion of the wrist and the object. Zhao et al. [ZCX12] synthesize similar interactions combining marker based motion capture data with RGB-D cameras. A database of ten different grip shapes is captured holding a variety of objects. Contact force information is then manually applied to the different grip shapes. Motion captured data has also been used to compute the best parameters for physical models [PZ05]. Physics based techniques are described in more detail in Section 3.4.2.

3.4. Procedural Animation Methods

Procedural systems can help manage the complexity of hand models when data-driven and/or keyframing approaches are infeasible and/or undesirable. To create controllers that produce natural motion, these approaches employ rules derived from a variety of sources that aim to explain how hands should move and function. For example, taking the hand's biomechanics, procedural algorithms can be crafted from rules that facilitate hand and finger coordination from the hand's natural structure [RG91, SRWG98, SFS98, LWH00, BZ04]. Likewise, procedural techniques can utilize application driven rules to constrain movement, for example, collision-free path planning, which leads to a unique set of rules in contrast to the gestural hand movement.

The largest class of procedural methods falls into the category of physically based modeling. Early research in character animation identified the benefits and challenges of simulated hands, as well as the need for control systems to help guide movement [ADH*89, SC92]. And since, many researchers in animation (and robotics) have proposed methods that employ physics and control to animate hands. One advantage is that hand motion can be synthesized without the need for input motion trajectories which relieves many of the problems that come with data capture. In addition, a benefit to using physics is the ability to add disturbances in the form of external forces and perturbations. The latter is key for hand manipulation which is often characterized by complex, purposeful, and incidental contact forces.

3.4.1. Non-physical procedural motion

Many procedural approaches take advantage of the algorithmization of key information related to the domain. Path-planning for manipulation and grasping is an example as motion can be generated within pre-defined kinematic constraints (e.g., hand poses, object position and orientation if grasping is the goal of the animation) to synthesize movement to/from an expected goal [SCSL02, YKH04]. Likewise, application specific desired contact positions, pose selection from pre-existing libraries, and inverse kinematics provide spatial guides that allow direct or search-based trajectory synthesis [HBMTT95, ES03, HL10b, ZRHN13]. Further, temporal cues establish hard and soft timing setpoints to drive hand motion to move in concert with other salient features in an animation [CVB01, NKAS08]. These systems formulate synchronized non-verbal communicative behaviors with timing and pace that support the speech signal. Many examples of procedural animations for hands have appeared in various domains, each exploiting a new set of rules based on the application. In Section 4, we organize and highlight a broad example set by application area.

3.4.2. Physics based techniques

By employing physical rules, hand and finger animation has been generated under a spectrum of settings. Two primary goals for the research in this direction are increased quality through the naturalness of appearance-driven anatomical models [TBHF03, SKP08] and increased generality through responsive interaction, especially for force-based manipulation [PZ05, KP06, Liu09]. Note a survey on the specific topic of simulation for hand animation appears in [SP14].

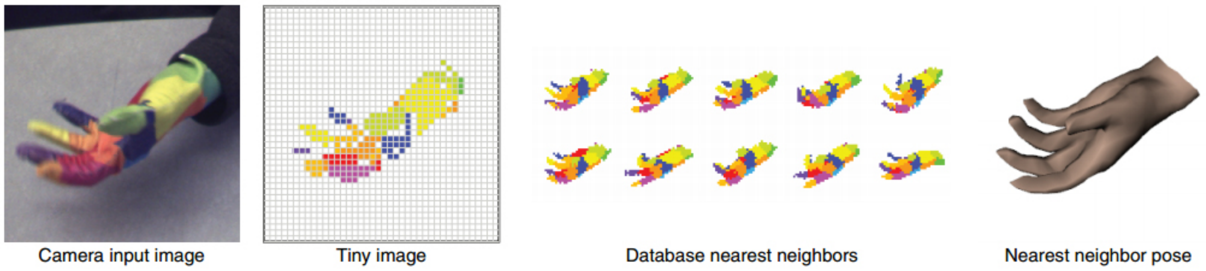

Figure 8: *The pose estimation process proposed by Wang and Popović for use with their colored glove. The original captured image is represented as a normalized tiny image. The image is an input query for a nearest-neighbor search algorithm that returns a corresponding pose from a database [WP09].*

Anatomical hands models. A number of researchers have established the goal of creating natural looking shape and movement by modeling the multi-layered, physical structure of the hand. Many rely on physical simulation of those layered components to add realism based on the established model. An example is Albrecht et al. [AHS03] who create an anatomically driven model of a hand which is animated at the bone level through a set of simplified muscles that torque finger joints, coupled with deformable spring-mass volume models that shape and deform the local area with a skin layer that wraps over the surface to produce the visual rendering of the hand. Tang and Hui [TH09a] also use a spring-mass system to model tendon movement in the hand with the goal of achieving the correct skin deformation on the hand's surface. More sophisticated approaches have been proposed to animate muscle deformation under the surface, using finite volume models for muscles [TBHF03, LST09].

Sueda and colleagues [SKP08] add musculotendon effects by simulating a thin strand model using an algorithmic controller to determine the muscle activations. The simulated strands add the appearance of tendons, visible through the skin, based on keyframed bone animation input. Using a method from the field of biomechanics [DL95], Tsang et al. [TSF05] present an anatomically correct hand with a two-pass forward/inverse simulation that is able to drive hand poses through tendons, as real hands are controlled. One goal of such an approach is to support medical applications and visualization of the hand in action.

Physical interaction synthesis. While physical attributes can add value in a number of fashions, two classes of techniques have been employed to synthesize purposeful force-based movement for the hand. With a few exceptions [ICL00, NS06], the goal of these systems is interaction synthesis that lends itself to manipulation and grasping. The first class of techniques employs forward simulation to enforce physical correctness, and the primary thrust of the work comes in the form of the design of an action controller that drives the hand to perform desired behaviors. The second set uses constrained trajectory optimization to compute physically correct motion based on a series of short-horizon control problems with a set of desired objectives. While abstractly the difference between these two sets of methods is arguable, the approach for solving each is fairly different, and the distinction is helpful for discussing each in this review.

Early on, a main disadvantage with using forward simulation appeared to be the time needed to solve the equations [ADH*89]. However, currently, controller design is the prominent focus in forward simulation techniques. Using a control system, the resulting animation is created by applying generalized forces to the dynamic simulation of the hand along with inertial influences and external disturbances. In most of the published examples to date, the controller follows a hierarchical structure with low-level activation coupled with behavior-level control and sometimes higher-level planning. Neff and Seidel present a clean example of low-level control [NS06]. They use a joint-based proportional-derivative (PD) controller to animate the hand in simple response to gravity. This controller acts as a basic muscle system by generating joint torques to keep the hand close to a static (or desired) rest pose. Tuning gain values can be problematic for such control, although for this particular work the controller values were acquired by matching observations and timing in videos of a person performing the hand motion to be animated.

By layering a state machine over PD-control, Pollard and Zordan [PZ05] create a hierarchical controller that perfoms object-grabbing actions. Along with gravity compensation, a feedforward term, and control parameters extracted from motion capture data, the controller combines passive responsiveness with active control of flexion to grab an object. Andrews and Kry [AK12, AK13] use a similar hierarchy but take it a step further in order to produce more subtle, multifinger manipulations. Rather than extracting poses from a motion capture trajectory, their system computes hand and finger poses for three states (approach, actuate, and release) using a multi-stage optimization. In this fashion, the optimizer is able to search for a small set of joint angles that

Figure 9: *Examples of input grasp poses (top) and poses modified to perform manipulations (bottom) in the physics-based method proposed by Liu [Liu09].*

accomplish the desired manipulation, for example turning a dial, by exploring the effect of the poses across the behavior – without an explicit example or precise control direction.

By recasting the synthesis problem into a series of small, so-called *short-horizon* search problems, physically based animation can be produced that accomplishes a desired outcome. The structural difference between this approach and forward simulation is that the dynamic equations of motion need not be solved explicitly (through forward simulation) but instead can act as constraints that limit the feasible search space to remain physically plausible. For example, in two papers [Liu08,Liu09], Liu describes approaches that synthesize physically correct hand and arm reaching and manipulation from a sequence of short-horizon optimizations.

Liu's two approaches differ somewhat in their specific problem formulations. Namely, the first includes kinematic objectives that, for example, dictate contact of a finger tip, while the second plans for desired contact forces and uses these as constraints in the optimization. Both uphold physical correctness through constraints based on the dynamic equations of motion, but promote smoothness in the multi-stage trajectory differently, through target hand pose [Liu08] or minimum internal torque change [Liu09]. To contrast the effect of these choices, the former casts a problem with conflicting kinematic and dynamic goals to create hand motion, while the latter has the benefit that contact planning leads to force constraints that form a cleaner search space, which is solved with the single objective of smooth joint activation and ultimately results in more robust, less stiff manipulation. Figure 9 shows examples of these synthesized manipulations.

This leads to another important problem in physical interaction synthesis: the modeling of compliance. To mimic real manipulation, we need the hand and fingers to comply or "give" in the presence of sufficiently large forces. The techniques described above deal with this issue by providing various means for keeping the hand from becoming stiff. Jain et al. [JL11] offer an approach to compliance by modeling deformation at the contacts of a simulated manipulation. However, the interaction capture approach of Kry and Pai [KP06], shown in Figure 10, tackles the compliance problem in a more direct and unique way.

To animate hands interacting with a variety of objects, Kry and Pai simultaneously capture markers from the hands and force sensors mounted on the fingertips as real-world manipulations are performed. This allows them to compute both the motion of the joint angles of the hand and possible internal forces that explain the measured contact forces between the hand and manipulated objects. From this "interaction capture", they build controllers with estimated joint compliance, derived from the small time periods before and after the contact. Subsequently, reference motion can be combined with the joint compliant control to synthesize new motions, including those in response to simulated perturbations. The approach also allows to synthesize compliant hands manipulating objects of different shape, size, and surface texture.

3.5. Surface-based models

Along with its movement, the visual rendering of a hand model affects how it is perceived. To make hands look more realistic, researchers have proposed several techniques to improve control, calculation speed, and appearance of the surface representation of the hand, for example, by adding skin wrinkles [YZ05] and palm creases [RNL06]. Realism has been approached by modeling underlying bones, muscles, tendons, and blood distribution that can be seen through the skin [AJK13], in addition to the deformations of the skin layer itself. While we describe several related anatomical modeling techniques in the previous section, surface models for hands have given rise to their own techniques. For example, Kurihara and Miyata [KM04] present a technique to model hand surfaces based on medical images.

As hands move into different configurations, the surface of the hand deforms accordingly [ADH*89]. Early methods for skin deformation propose using a set of operators that affect the skinning of a character based on the configuration of the joints using joint-dependent local deformation [MTT87, MTLT88]. Each deformation operator is responsible for an individual section of the skin surface. In contrast, Gourret and colleagues [GTT89] use finite elements to simulate different skin deformations when the hand is grasping objects. This method also allows them to simulate shape and create the correct contacts between the hand and the object. Moccozet and Magnenat-Thalmann [MMT97] propose a free-form deformation (FFD) model which depends on Delaunay and Dirichlet diagrams to simulate the tissue between the skeleton and the skin. With this approach, they are able to produce wrinkles around the joints and smooth deforma-

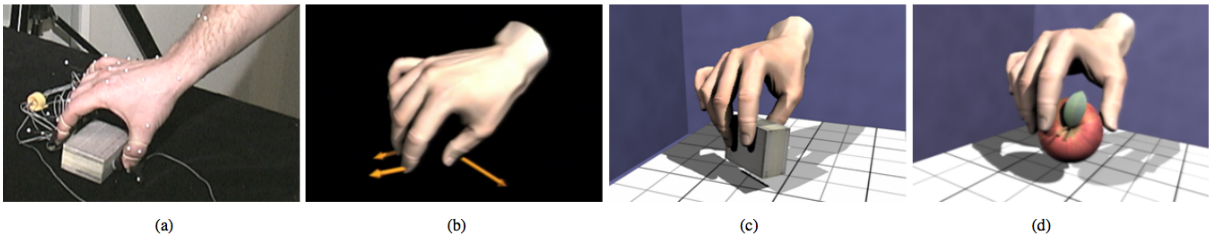

Figure 10: *An example of Kry and Pai's Interaction Capture approach: (a) A grasp is recorded using motion capture markers and fingertip mounted sensors to measure the contact forces; (b) Recorded contact forces are illustrated; (c) A synthesized interaction with a similar object; (d) A synthesized interaction with an object of a different shape [KP06].*

tions across the surface of the hand. The FFD also allows the geometric representation of the skin to respond to contacts.

Kry and colleagues [KJP02] sought to improve the skeletal-subspace deformation skinning technique by Magnenat-Thalmann et al. [MTLT88]. Starting from a simulation of the hand with a finite-element deformable soft-tissue model, they collect a set of simulated sample hand poses and subsequently determine which joints influence displacements in the deformed surface mesh. Principle component analysis is performed on the positions of the selected vertices to acquire a set of *eigen displacements*. By truncating these terms, they find approximate deformations that can be computed quickly. At run-time, each of the vertices contribute to the final skin deformation based on the eigen displacement basis for that vertex and the current bone configuration of the hand.

In a more recent method, Huang and colleagues [HZY*11] create a multiscale surface hand model that has large scale deformations with high resolution details such as wrinkles and creases (Figure 11). To achieve this goal, they capture a set of high precision 3D scans and fit the poses with control points. These poses act as the training input for both the deformation layer and the detail surface layer. When new control points are added, the two layers for the new pose are updated and high resolution hand surfaces are synthesized.

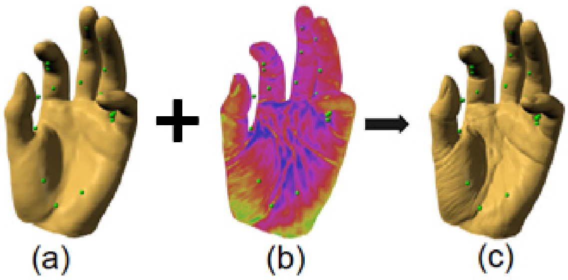

Figure 11: *Huang's method. Training poses inform large scale deformations (a) and high resolution deformation details (b) for the final result (c) [HZY*11].*

4. Applications

4.1. Manipulation

On a daily basis, human hands must perform precision tasks that involve manipulation through grasping and deliberate interaction via contact. It is desirable for virtual characters to manipulate objects in their environment with their hands in a similar manner. Synthesizing hand grasps of arbitrary objects is important but challenging. Grasp synthesis can be difficult because of the large number of possible configurations of the hand, variation in the types of objects to grasp, as well as the choice of grasp that would make a movement natural (e.g. a two-finger pinch or a full hand grasp). Researchers have created classifications for different human grasps and used information regarding the object and the current task to determine the correct grasp choice for robots [CW86, CH90, MKCA03]. While manipulation is an application that researchers in both animation and robotics focus on concurrently, animation techniques must prioritize naturalness in consideration of the shape of the hand as well as the contact necessary.

Before the onset of motion capture, early researchers painstakingly crafted the shape of a grasp manually and placed the hand in a plausible location relative to the object in order to make visually pleasing manipulations [MTLT88]. In this work, the points of contact on both the hand and the object are specified for each grasp. Rijpkema and Girard [RG91] propose a procedural approach that takes into account the shape of the object and a selection of grasps from a library and apply rules to adjust the grasp according to an analysis of the original shape, the current object, and necessary contact forces. They also use a grasp-planning algorithm and a knowledge-based approach similar to those in robotics [TBK87].

Sanso and Thalmann [ST94] propose an automatic system to determine how to grasp (e.g., with two fingers or two hands) and apply joint constraints to make the closing of the hand look realistic. Huang et al. [HBMTT95] offer an alternative method that also takes into account the shape of the object and decides on the grasp strategy. Their heuristic determines the grasp based on object type, size, and hand geometry, and their hand uses virtual sensors to maintain contact. Aydin and Nakajima [AN99] present an approach that allows a virtual character to employ the full body to perform grasping for objects placed at a number of locations, including the ground. They classify the object to be manipulated, estimate the hand shape needed, estimate the posture of the body needed, and then perform the grasp relying on a forward/inverse kinematics scheme.

As motion capture technology became more available, data-driven techniques grew in popularity. For example, Li et al. [LFP07] construct a database of grasp shapes recorded using motion capture. They also construct a database of various objects of different shapes and sizes. To choose the best grasp for each object, a shape-matching algorithm is employed which returns multiple potential matching grasps from the shape database that are then clustered and pruned to find the best grasp. An alternative take on data-driven manipulation is proposed by Ye and Liu [YL12]. Their approach provides a full body (input capture) with realistic and intricate hand configurations for manipulating objects (also recorded). To synthesize hand manipulations, they randomly sample a set of hand-object contact positions to determine feasible contacts based on the present hand configuration. Once contact points are chosen, optimization is performed to move the hand from one configuration to the next.

Hybrid data-driven and physics based systems also began to appear [PZ05, KP06]. For example, using their system, Pollard and Zordan simulate interactions such as the two hands grasping each other in the handshake seen in Figure 12. While physically realistic, these approaches rely heavily on reference examples. In more recent efforts, physics-based synthesis is migrating away from dependence on motion data, for example in Bai and Liu's simulated object rolling [BL14]. This optimization-driven approach controls a virtual robot hand to dextrously maneuver the position and orientation of an object automatically. They create two controllers, one to pick up the object and drop it in the palm and the other to grasp the object and correct its rotating motion. The object rolls with respect to gravity and contact forces associated with the tilt of the palm. To control the tilt angle of the palm, the algorithm relies on principles of mechanical energy.

Virtual Reality applications. An important set of applications of "direct" virtual manipulation appear in the domain of human computer interaction and virtual reality (VR) systems. In many VR systems, virtual hands allow a user to interact with objects in the scene. The VPL DataGloves let users naturally interact with simulated environments [SZ94] and have been used to control touch, grabs, movement, and even throwing virtual objects as well as selecting on-

Figure 12: *A physical controller coupled with motion capture able to perform complex interactions such as a handshake [PZ05].*

screen menu items [KY89, SZP89, FB90]. The VPL software system would execute events, such as grabs, when certain hand poses were recognized. More recent research looks at making VR grasps appear more like the actual user grasps [LCZ*11, AC12]. Beyond non-VR applications, research in this arena focuses on optimizations to keep processes running in real time.

4.2. Communication

Gesture is a key component of nonverbal communication, and an important aspect of communication overall. Gesture animation focuses largely on the hand, considering positioning, timing, and hand shape, and represents an important application of hand animation to support communication. The movement of individual fingers is not always considered and is not a focus of this section. Jörg et al. [JHS12] propose a method to automatically add finger animation to body motions for conversational characters that can be used in combination with approaches where this type of motion detail is not provided.

Human gesture and speech is produced together from what is commonly thought of as a single communicative intent [Ken04, McN92]. Systems that generate conversational characters, such as the SAIBA project [SAI12], tend to follow this idea, combining speech and gesture to produce high level communication. A recognized model for gesture production is the Prep, Stroke, Retraction (PSR) model of gesture phases. The PSR model was first developed by Efron [Efr41], an anthropologist, and later refined by Kendon [Ken80] and others [KVV98]. According to the PSR model, a gesture can be divided into a set of phases as follows:

$$\text{GESTURE} \rightarrow [\ \textbf{preparation}\ ]\ [\ \textbf{hold}\ ]\ \text{STROKE}\ [\ \textbf{hold}\ ][\ \textbf{retraction}\ ] \quad (2)$$

The meaning of the gesture is carried by the *stroke* phase. As such, a gesture should always have a stroke phase, with the other phases being optional, except for independent holds (stroke holds) [KVV98,McN05]. The *preparation* phase places the arm, wrist, hand, and fingers in the proper configuration to begin the stroke [KVV98]. In the *retraction* phase, the arm returns to a rest position. It is generally thought that the *hold* phases exist to synchronize the motion of the gesture with speech [dR00,McN05]. Hold phases could also convey that the speaker maintains a state for a certain length of time. The PSR model provides the basis for many gesture synthesis algorithms.

A large number of researchers have studied how to produce gesture animation. The two key problems are to determine which gestures should be performed in a given situation and to generate appropriate animation of those gestures. Animation techniques have included procedural approaches, data-driven techniques, and physical simulation. Significant attention has been paid to controlling the style of the motion and synchronizing it appropriately with speech.

Modeling the style of gestural movement is necessary in order to create a sense of character and personality. Chi et al. [CCZB00] designed the EMOTE system by using the Effort and Shape components of Laban Movement Analysis (LMA) to define a set of animation control parameters. Effort, for example, consists of four parameters: Weight, Space, Time and Flow. Each parameter has two poles; for example, Weight ranges from Light to Strong. An animator can change the Weight parameter and the resulting animation will be more delicate or more powerful. The system is kinematic with hand tuned mappings between the LMA parameters and spatial and temporal controls. These mappings were validated through a user study.

Hartmann et al. [HMP02] focus on creating believable Embodied Conversational Agents (ECAs), specifically for information delivery. Their approach takes a user inquiry as input and responds with an agent trained in the specified domain of knowledge. They introduce a kinematic animation system, the Gesture Engine. Follow-up work [HMP06] extends this approach to provide parametrized, expressive control of arm gestures. They model the parameters: overall activation, spatial extent, temporal extent, fluidity, power, and repetition.

With an emphasis on creating natural motion, Kopp et al. [KTC04] present a gesture animation framework based on neurophysiological research to control the timing of novel iconic gestures. Iconic gestures focus on visual representations of concrete entities, for example, when describing an object, imitating an action, or giving directions.

Neff and Fiume [NF05] introduce a system that uses editing operations designed based on the arts literature to modify the style of an animation sequence. They automate these style modifications for complete sequences through the use of customizable *character sketches*.

Gibet et al. [GKP04] apply invariant features that should be maintained in gesturing agents, including Fitts' law [Fit54], the two-third power law [VT82], and gesture movement smoothing [FH85, UKS89] following motor control theory, and then give a brief discussion on how these laws can be applied to motion generation and editing.

Physics-based techniques have seen less use in gesture animation than manipulation animation, but several systems support physical simulation for generating the output [NF05, NKAS08, vWRRZ10], often using careful tension control [NF02] to make the character appear more loose or stiff and provide secondary oscillations to the motion. These techniques use proportional-derivative (PD) controllers to provide a simple model of muscle, forward simulation to generate the output, and layered balance controllers to keep the character standing.

Data-driven techniques are popular for gesture animation as they provide high quality, natural motion. The variation space for gestures is very large, so it can be a significant obstacle to capture data for the massive range of feasible interactions. As discussed earlier, data-driven approaches may also offer less control over the motion, particularly if they are limited to playing back previously recorded motion. Motion graphs have been used in several approaches. Stone et al. [SDO*04] present a system that uses a motion graph across combined data of speech and animation. Generating different paths through the motion graph provides different multimodal output sequences with synchronized gesture and speech as feedback for a video game player. Fernandez-Baena et al. [FBMA*14] develop a Gesture Motion Graph (GMG) for generating gesture animation sequences and then use synchrony rules to match the intensity of gestures to the intensity of the speech.

Some recent approaches have applied machine learning techniques to try to generalize gesture models from data. Based on an extension of deep belief networks, Chiu and Marsella [CM11b, CM11a] use hierarchical factored conditional restricted Boltzmann machines (HFCRBMs) [TH09b] (extending [THR06]) to generate gesture sequences from data, triggered by prosody. Later, Chiu and Marsella [CM14] use dynamic Gaussian Process Latent Variable Models (GPLVMs) [WFH08] to learn a low-dimensional embedding of gesture data and find smooth connections between gestures in this space.

The relationship between speech and gestures was often specified with a custom representation language that was paired with an animation system. For example, the Gesture Engine by Hartmann et al. [HMP02] realizes an abstract scripting language for specifying gesture definitions by synthesizing gesturing behavior. Kopp and Wachsmuth [KW02] generate human-like multimodal utterances, gestures, and concurrent speech for a virtual conversational agent that interacts with humans. Later, Kopp and Wachsmuth [KW04] extend their work to develop the Multimodal Utterance Representation Markup Language that is used to specify body and hand gestures, facial expressions and prosodic speech synthesis.

These early specification languages led to the Behavior Markup Language (BML), an XML description language for specifying the verbal and nonverbal behavior of embodied conversational agents [KKM*06, VCC*07]. BML is meant to be independent of any particular system and a BML realizer is an animation engine that can transform BML into character animations. A number of researchers have developed BML realizers, such as Elckerlyc [vWRRZ10], a BML realizer for generating multimodal verbal and nonverbal behavior for virtual humans; SmartBody [TMMK08], a BML realizer that also provides locomotion, steering, object manipulation, lip syncing, and real time gaze control; EMBR [HK10], which supports micro-planning; and Greta [MNBP08], which features significant facial control. BML realizers generally follow a procedural approach and play back either key framed or motion captured examples of gesture, sometimes with parametric variation.

Numerous techniques have been developed to determine which gesture should be performed to accompany a given passage of text. The Behavior Expression Animation Toolkit (BEAT) [CVB01] is an enhanced rule-based text-to-speech system that takes plain text/script as input and uses a set of predefined rules to automatically generate prosody and speech synthesizer intonation, facial animation, and gestures. Stone et al. [SDO*04] use a multimodal data corpus that captures the relationship between speech and gesture. The work of Kipp and colleagues [Kip04, KNKA07] and Neff et al. [NKAS08] uses a statistical model of individual speaker behavior to predict how a particular person will gesture, given input text. The nonverbal behavior generator presented by Lee and Marsella [LM06] is another rule-based tool for automatically generating believable nonverbal behaviors for embodied conversational characters by analyzing syntactic and discourse patterns. Bergmann and Kopp [BK09] propose a data-driven model for integrated language and gesture generation that can account for systematic meaning-form mappings, where speaker preferences are learned from corpus data. Bergmann et al. extend these approaches by also including a cognitive model [BKK13].

Focusing on audio instead of text, Levine et al. predict the timing and type of gesture based on the prosody of the audio signal using first hidden Markov models [LTK09] and then conditional random fields [LKTK10]. Chiu and Marsella follow a similar approach using HFCRBMs [CM11a] and then extend this approach to a two level technique that first predicts the type of gesture from input audio using Conditional Random Fields and then generates the required motion using GPLVMs [CM14]. Fernandez-Baena et al. [FBMA*14] use synchrony rules to match the intensity of gestures to the intensity of the speech. Models based purely on prosody recognize the important correlation between gesture timing and audio changes (e.g. explored in [KdR12, WN13b]), but cannot account for deep semantics. Newer work seeks to address both, for example, the Cerebella system [LM13].

4.3. Sign Language

An important and challenging application for detailed hand and finger animation is depicting sign language. Tools that can produce quality sign language animation can be very useful for members of the deaf community. Over the years, many projects have explored ways to recognize and create hand signs, leading to major innovations in the creation of detailed finger animations. For example, in the 1980s, Kramer and Leifer wanted to build a portable system for the purpose of sign recognition and for sign language to spoken word translation [KL88]. Out of this research came the first CyberGlove [KLG91], which was instrumental in the more recent research on this topic.

Adamo-Villani and Beni [AVB04] created an educational tool to teach people to sign and read finger spelling. They use a realistic hand model with a skeletal deformation system that closely resembles the skeleton of a real hand. Their belief is that realism helps to better identify the shape and position of the hand. The arm and hand are animated using a combination of forward and inverse kinematics. Their tool, which runs in Maya, allows a user to input text and the hand will spell out what was written. They also provide controls to manage the speed of the motion, the rotation of the hand, and the camera angle.

A Chinese sign language recognition and synthesis system is proposed by Gao et al. [GMS*00, GCF*04]. Using a data glove to provide input data, they initially use a fast-match algorithm to find a list of words from their vocabulary that is similar to the input. Then they assign probabilities to the words based on context and search for the most likely word. Their system also captures facial motion to apply it to the signing avatar.

In 2010, Lu and Huenerfauth describe how they create a motion capture ASL corpus [LH10]. They captured body movements and hand signs from native signers using a combination of sensored gloves, motion capture, eye-tracking, and video. Figure 13 shows their capture setup and an example of their animated character. The collected data is then, for example, used to produce inflected verb signs [LH11]. For this type of signs the motion path varies depending on the location in space to which the object and, if present, the subject have been assigned on an horizontal arc-shaped space around the signers body. Huenerfauth and Lu's previous work uses a database created by human signers with the sign language animation tool VCom3D Gesture Builder

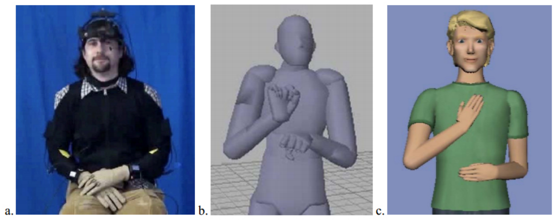

Figure 13: *The technique used by Lu and Huenerfauth to create a motion capture ASL corpus: (a) Motion capture setup consisting of a bodysuit with intertial and magnetic sensors, an acoustical/inertial sensor for the head, two CyberGloves, and an eye-tracker; (b) An animation generated from the motion capture data; (c) An animation of their character Sign Smith performing a sign [LH10].*

[HL10b]. A third-order polynomial model is fitted to each location parameter for each hand, keyframe, and verb. Based on this parameterization, inflected verbs for new subject and object locations can be generated. The same method is then applied to motion captured data [LH11].

Sign language has also been used for evaluation purposes or as a testbed for new methods, for example, in the work of Adamo-Villani [AV08] and Wheatland et al. [WJZ13].

4.4. Instrumental Performance

Playing most musical instruments requires significant manual dexterity. String and keyboard instruments, for example, require complex positioning of the fingers. Research has aimed to automatically recreate the fine motions necessary to animate the playing of instruments such as the violin [KCMT00], guitar [ES03], and piano [KMO*09, ZRHN13].

A significant hurdle when playing musical scores is that fingerings (which finger the player uses to play which notes) are not explicitly stated in the sheet music. Several researchers have addressed this problem using rule-based approaches based on musical teachings to map from notes to fingerings [Say89, PSC*97, Jac01]. A general issue with this type of approach is that the number of rules can become lengthy and unmanageable.

ElKoura and Singh [ES03] combine rule-based and data-driven methods to create a virtual character that is able to read a musical passage written as guitar tablature and to play it on a virtual guitar (Figure 14). They propose a cost-minimization algorithm to determine the fingering of the fretting hand that minimizes objectives such as the required effort. Once the fingering is solved, the exact joint orientations are determined using a k-nearest-neighbors search in a database of motion captured guitar motions to obtain natural hand poses.

Piano playing involves both hands simultaneously playing

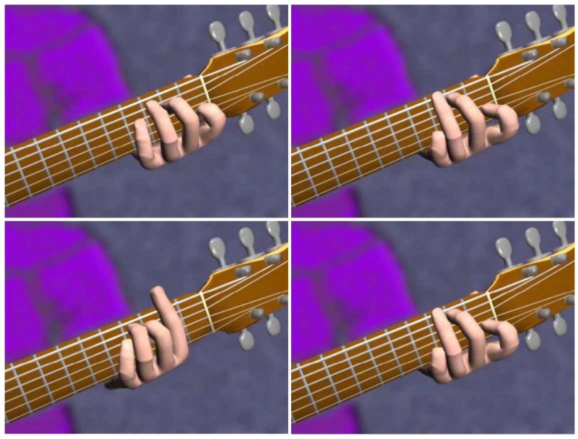

Figure 14: *An example of the Handrix system by ElKoura and Singh performing a C (top left)-G (top right)-F# (bottom left)-G (bottom right) chord progression [ES03].*

different note patterns. Kugimoto et al. [KMO*09] present a system using motion capture data to generate animated piano playing from MIDI files or musical scores. They capture the motions of a piano player's hands and animate the virtual hands based on this data. Similarly to the previous method, their work algorithmically determines the optimal piano fingering. It furthermore uses a function to determine the trajectories of the fingers' control points based on the proposed fingering. Zhu et al. [ZRHN13] also developed a system to animate virtual piano playing based on an input MIDI file. Their kinematic system uses a rule-based method to find the most comfortable fingering for each note or chord. Then a graph-theory based motion planning algorithm is used to determine the optimal key fingerings for the entire musical sequence. Lastly, the position of unused fingers is calculated to make future fingerings more efficient. The system can achieve complicated fingering sequences such as cross-overs where the hand crosses over the thumb. Once the series of poses is generated, the result is optimized to create motion curves based on the performance style.

5. Discussion and Future Trends

This report provided an overview of the research and technologies for creating hand and finger animation for virtual characters, grouping approaches into keyframing, data-driven animation, procedural animation, and surface based animation. Each of these methods has benefits and drawbacks. Keyframing gives animators full control over the final animation, but can be time consuming, and it is difficult to adapt keyframed motion to new situations. Data-driven methods can produce realistic and highly detailed results, but require pre-recorded human motion data similar to the desired result in order to create animations. Capturing elaborate and accurate finger movements is difficult. Optical motion capture technologies require laborious post-processing and occlusions pose a significant problem, while sensored gloves are challenging to calibrate and reconstructions can be inaccurate due to sensor cross-coupling. Accurately determining the size of the hand skeleton is also a hard task when using motion capture techniques. Procedural methods include a wide set of approaches that are driven by algorithms or rules, which can be based on text and speech, psychological principles, anatomical rules, physics-based laws or statistical models. These approaches are often flexible, but when no motion data is used, more complex rules may be required to create high quality, detailed motion. With physics-based algorithms, motions are adaptable to different situations, but might be expensive to compute and can appear unnatural. Surface-based models can be combined with all three animation methods to add details such as wrinkles to the appearance and rendering of a hand model.

We summarize common applications of hand animation under the topics manipulation, communication, sign language, and musical performance. In terms of trends, we note that much manipulation and grasping research takes a procedural approach that includes physical simulation. This is due to the fact that manipulation is highly dependent on the particular task and physical interaction. Thus algorithms that can adapt and generate specific, novel motion perform best. There are numerous successful examples of manipulation research that combine procedural methods with data-driven methods to create more natural-looking animations. In those approaches, the data-driven motion is often used as a first guess and then customized to the particular task.

When it comes to communication, data-driven and rule-based methods dominate. There is a trade-off between the realism of captured motion data and the flexibility of procedural approaches. There are many ways to translate a specific piece of information into gestures or to convey a particular character's personality, and we have yet to fully understand this intricate process. Therefore, the desired motion is often underspecified. Data-driven techniques can add details to the motion in an acceptable way, but we expect that the demand for motion that precisely reflects both the character and the communicative context will increase as quality demands grow. Quality and control can benefit from both new data-driven methods and improved procedural control. Gesturing tasks are less physically constrained, and hence have seen few uses of physics-based models. However, physical simulation can improve the motion quality and adaptability to constraints and is a likely future trend. More and more hybrid solutions span the spectrum from data-driven to procedural techniques, and we expect such hybrid approaches that combine the strengths of each technique to provide an important route forward.

Recent years, have seen an increase in new, more affordable products to track and capture hand motions, including the Microsoft Kinect and the Leap Motion Controller. Even though these technologies cannot capture detailed hand motions in large spaces, they broaden access and grow application areas such as virtual reality, digital games, or assistance and rehabilitation. Some of those applications only track the hands to provide control and not to create animations. Others give a basic or more detailed representation of the hand as feedback to the user. The possibilities and products in that area are currently evolving at a quick pace. For example, in the field of virtual reality, the recent surge in the consumer market due to affordable head mounted displays, like the Oculus Rift or Sony's upcoming Project Morpheus, is leading to new developments in hand tracking.

The modeling and animation of hands and fingers has seen great progress over the past three decades. However, capturing the complex motions of the hands accurately or creating a character that can communicate with people or manipulate objects in a believable way remains a challenge. As the number of applications requiring detailed hand animation continues to increase, the development of new algorithms and techniques for hand and finger animation remains a vibrant research area.

Much progress has been made in solving specific situations. Solutions exist to compute specific grasps, especially single hand grasps in known scenarios, or to create animations for communication in a well defined domain. But more general approaches that could handle a whole range of behaviors are missing. More flexible systems could be created with hybrid approaches, for example, combining prerecorded data with a physics system to increase adaptability. However, even when it comes to specific situations, many types of motions, for example, those involving complex interactions such as lacing a shoe, wringing one's hands, snapping fingers, or using sign language, can currently not be synthesized automatically at high quality. Improvements are required not only in the computation of such motions, but also in generating subtle skin deformations where the fingers interact and in creating wrinkles where necessary. These types of interactions cannot currently be captured accurately. In the motion capture area, we expect new capture methods for the consumer market based on RGB-D cameras to improve in accuracy and gain in popularity. However, the goal of capturing intricate motions easily and precisely in a large capture space together with the body has not been reached yet. Here again, combining different methods might lead to more powerful techniques. Finally, further systematic perceptual studies are needed to understand how we interpret hand and finger models and animations. Such studies would improve our understanding and control of how hand and finger motions express personality and emotions in multimodal communication and would also determine which details in hand and finger modeling and animation are most salient and need further research.

Biographies

Nkenge Wheatland is a Ph.D. student at the University of California, Riverside, where she works with Dr. Victor Zordan in the Riverside Graphics Lab. She is currently a visiting student at Clemson University working with Dr. Jörg. She received her B.S. and M.S. in Computer Science from the University of Maryland, Baltimore County and UC Riverside, respectively. Her current research focus is gesture synthesis for communicative characters with an emphasis on American sign language synthesis.

Yingying Wang is a Ph.D. student at the University of California, Davis, working with Dr. Michael Neff in the Institute for Data Analysis and Visualization (IDAV) lab. Her research focus is mainly on glove calibration for hand motion capture, synthesizing gestures for conversational characters, personality perception of finger motions, gesture perception, and motion retrieval.

Huaguang Song is a Ph.D. student working with Dr. Michael Neff in the Institute for Data Analysis and Visualization group at the University of California, Davis. He received his B.S. degree in Computer Science and his M.S. degree in Engineering Science from the University of the Pacific in 2010 and 2012, respectively. His current research interests are focused on character animation including procedural and data-driven gesture generation with embodied conversational agents.

Michael Neff is an Associate Professor in Computer Science and Cinema & Technocultural Studies at the University of California, Davis where he directs the Motion Lab, an interdisciplinary research effort in character animation and embodied input. He holds a Ph.D. from the University of Toronto and is also a Certified Laban Movement Analyst. His interests include character animation tools, especially modeling expressive movement, physics-based animation, gesture and applying performing arts knowledge to animation. He also currently serves as co-Director for the Program in Cinema and Technocultural Studies.

Victor Zordan is an Associate Professor at University of California Riverside. His research interests center around games and special effects, especially animation for characters with an emphasis on controllability. Victor has developed a host of techniques that merge motion capture and dynamic simulation to create flexible and responsive behavior for humanoids. He holds a Ph.D. from Georgia Tech.

Sophie Jörg is an Assistant Professor in Computer Science at Clemson University. Her research revolves around virtual characters, perception, motion capture, and digital games with a focus on data-driven character animation and motion perception. She holds a Ph.D. from Trinity College Dublin, Ireland, and was a visiting researcher at Carnegie Mellon University and Disney Research, Pittsburgh.

References

[3Ge14] 3GEAR SYSTEMS: Motion capture system website, http://threegearsystems.blogspot.com/, http://www.threegear.com/, August 2014. 9

[AC12] ALEOTTI J., CASELLI S.: Grasp programming by demonstration in virtual reality with automatic environment reconstruction. *Virtual Reality 16*, 2 (2012), 87–104. 15

[ADH*89] ARNALDI B., DUMONT G., HÉGRON G., MAGNENAT-THALMANN N., THALMANN D.: Animation control with dynamics. In *State-of-the-art in Computer Animation*, Magnenat-Thalmann N., Thalmann D., (Eds.). Springer-Verlag, 1989, pp. 113–123. 11, 12, 13

[AHS03] ALBRECHT I., HABER J., SEIDEL H.-P.: Construction and animation of anatomically based human hand models. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2003), pp. 98–109. 3, 12

[AJK13] ANDREWS S., JARVIS M., KRY P. G.: Data-driven fingertip appearance for interactive hand simulation. In *Proceedings of Motion on Games* (2013), MIG '13, ACM, pp. 155:177–155:186. 13

[AK12] ANDREWS S., KRY P. G.: Policies for goal directed multi-finger manipulation. In *The 9th Workshop on Virtual Reality Interaction and Physical Simulation (VRIPHYS)* (2012), pp. 137–145. 12

[AK13] ANDREWS S., KRY P. G.: Goal directed multi-finger manipulation: Control policies and analysis. *Computers & Graphics 37*, 7 (2013), 830–839. 4, 12

[AN99] AYDIN Y., NAKAJIMA M.: Database guided computer animation of human grasping using forward and inverse kinematics. *Computers & Graphics 23*, 1 (1999), 145–154. 15

[AV08] ADAMO-VILLANI N.: 3D rendering of American sign language finger-spelling: A comparative study of two animation techniques. *5th International Conference on Computer Instructional Technologies* (2008), 808–812. 5, 18

[AVB04] ADAMO-VILLANI N., BENI G.: Automated finger spelling by highly realistic 3D animation. *British Journal of Educational Technology 35*, 3 (2004), 345–362. 17

[BK09] BERGMANN K., KOPP S.: Increasing the expressiveness of virtual agents: Autonomous generation of speech and gesture for spatial description tasks. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1* (2009), AAMAS '09, pp. 361–368. 17

[BKK13] BERGMANN K., KAHL S., KOPP S.: Modeling the semantic coordination of speech and gesture under cognitive and linguistic constraints. In *Proceedings of the 13th International Conference on Intelligent Virtual Agents* (2013), Springer, pp. 203–216. 17

[BL14] BAI Y., LIU C. K.: Dexterous manipulation using both palm and fingers. In *IEEE International Conference on Robotics and Automation* (2014). 1, 15

[Bla14] BLAUSEN MEDICAL COMMUNICATIONS: Hand and wrist bones, http://commons.wikimedia.org/wiki/file:blausen_0440_handbones.png, September 2014. Edits made for this publication. 2

[BW71] BURTNYK N., WEIN M.: Computer-generated key frame animation. *Society of Motion Picture and Television Engineers 80*, 3 (1971), 149–153. 5

[BW76] BURTNYK N., WEIN M.: Interactive skeleton techniques for enhancing motion dynamics in key frame animation. *Communications of the ACM 19*, 10 (Oct. 1976), 564–569. 5

[BW95] Bruderlin A., Williams L.: Motion signal processing. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques* (1995), SIGGRAPH '95, ACM, pp. 97–104. 9

[BZ04] Braido P., Zhang X.: Quantitative analysis of finger motion coordination in hand manipulative and gestic acts. *Human Movement Science 22*, 6 (2004), 661–678. 4, 11

[CCZB00] Chi D., Costa M., Zhao L., Badler N.: The emote model for effort and shape. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (2000), SIGGRAPH '00, ACM, pp. 173–182. 16

[CGA07] Ciocarlie M., Goldfeder C., Allen P.: Dimensionality reduction for hand-independent dexterous robotic grasping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2007), pp. 3270–3275. 4

[CGK00] Chou T., Gadd A., Knott D.: Hand-eye: A vision based approach to data glove calibration. *Proceedings of Human Interface Technologies* (2000). 7, 8

[CH90] Cutkosky M. R., Howe R. D.: Human grasp choice and robotic grasp analysis. In *Dextrous Robot Hands*, Venkataraman S. T., Iberall T., (Eds.). Springer-Verlag New York, Inc., 1990, ch. 1, pp. 5–31. 14

[CH05] Chai J., Hodgins J. K.: Performance animation from low-dimensional control signals. *ACM Transactions on Graphics 24*, 3 (July 2005), 686–696. 6

[Cha01] Chao A.: *Uni-dimensional evaluation of Wristsystem and CyberGlove*. Master's thesis, Department of Industrial and Manufacturing Engineering, Wichita State University, Los Alamitos, CA, 2001. 7

[CLPF05] Cerveri P., Lopomo N., Pedotti A., Ferrigno G.: Derivation of centers and axes of rotation for wrist and fingers in a hand kinematic model: Methods and reliability results. *Annals of Biomedical Engineering 33*, 3 (2005), 402–412. 4

[CM11a] Chiu C.-C., Marsella S.: How to train your avatar: A data driven approach to gesture generation. In *Proceedings of the 10th International Conference on Intelligent Virtual Agents* (2011), IVA'11, Springer-Verlag, pp. 127–140. 16, 17

[CM11b] Chiu C.-C., Marsella S.: A style controller for generating virtual human behaviors. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 3* (2011), International Foundation for Autonomous Agents and Multiagent Systems, pp. 1023–1030. 16

[CM14] Chiu C.-C., Marsella S.: Gesture generation with low-dimensional embeddings. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems* (2014), AAMAS '14, pp. 781–788. 16, 17

[Con15] Control VR: Motion capture system website, http://controlvr.com/, January 2015. 8

[CP07a] Chang L. Y., Pollard N. S.: Constrained least-squares optimization for robust estimation of center of rotation. *Journal of Biomechanics 40*, 6 (2007), 1392–1400. 4

[CP07b] Chang L. Y., Pollard N. S.: Robust estimation of dominant axis of rotation. *Journal of Biomechanics 40*, 12 (2007), 2707–2715. 4

[CP08] Chang L., Pollard N.: Method for determining kinematic parameters of the in vivo thumb carpometacarpal joint. *IEEE Transactions on Biomedical Engineering 55*, 7 (July 2008), 1897–1906. 4

[CPMX07] Chang L. Y., Pollard N., Mitchell T., Xing E. P.: Feature selection for grasp recognition from optical markers. In *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2007), pp. 2944–2950. 5, 6

[CVB01] Cassell J., Vilhjálmsson H. H., Bickmore T.: Beat: the behavior expression animation toolkit. In *ACM SIGGRAPH 2001* (2001), pp. 477–486. 11, 17

[CW86] Cutkosky M., Wright P.: Modeling manufacturing grips and correlations with the design of robotic hands. In *IEEE International Conference on Robotics and Automation. Proceedings* (Apr 1986), vol. 3, pp. 1533–1539. 14

[Cyb14] CyberGlove Systems: Motion capture system website, http://www.cyberglovesystems.com/products/cyberglove-ii/overview, August 2014. 7

[DGT14] DGTech Engineering Solutions: DG5 VHand description website, http://www.dg-tech.it/vhand/eng/index.html, August 2014. 7

[DL95] Delp S. L., Loan J.: A graphics-based software system to develop and analyze models of musculoskeletal structures. *Computers in Biology and Medicine 25*, 1 (1995), 21–34. 12

[dLGPF08] de La Gorce M., Paragios N., Fleet D.: Model-based hand tracking with texture, shading and self-occlusions. In *IEEE Conference on Computer Vision and Pattern Recognition* (June 2008), pp. 1–8. 9

[DM02] Damasio F. W., Musse S. R.: Animating virtual humans using hand postures. In *SIBGRAPI* (2002), IEEE Computer Society, p. 437. 10, 11

[DMR06] Dhawale P., Masoodian M., Rogers B.: Bare-hand 3D gesture input to interactive systems. In *Proceedings of the 7th ACM SIGCHI New Zealand Chapter's International Conference on Computer-human Interaction: Design Centered HCI* (2006), CHINZ '06, ACM, pp. 25–32. 9

[Dor94] Dorner B.: *Chasing the Colour Glove [microform] : Visual Hand Tracking*. Canadian theses. Thesis (M.Sc.)–Simon Fraser University, 1994. 8

[dR00] de Ruiter J.: The production of gesture and speech. In *Language and Gesture: Window into Thought and Action*, McNeill D., (Ed.). Cambridge University Press, Cambridge, England; New York, NY, 2000, pp. 284–311. 16

[DSD08] Dipietro L., Sabatini A., Dario P.: A survey of glove-based systems and their applications. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 38*, 4 (July 2008), 461–482. 7, 8, 10

[Efr41] Efron D.: *Gesture and environment*. King's Crown Press, 1941. 15

[ES03] ElKoura G., Singh K.: Handrix: animating the human hand. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2003), pp. 110–119. 4, 11, 18

[FB90] Feiner S. K., Beshers C.: Visualizing n-dimensional virtual worlds with n-vision. *Computer Graphics 24*, 2 (Feb. 1990), 37–38. 15

[FBMA*14] Fernández-Baena A., Montaño R., Antonijoan M., Roversi A., Miralles D., Alías F.: Gesture synthesis adapted to speech emphasis. *Speech Communication 57* (2014), 331–350. 16, 17

[FH85] Flash T., Hogans N.: The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of neuroscience 5*, 7 (1985), 1688–1703. 16

[Fif14] Fifth Dimension Technologies: 5DT Data Glove product description website, http://www.5dt.com/products/pdataglovemri.html, August 2014. 7

[Fit54] Fitts P. M.: The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology 47*, 6 (1954), 381 – 391. 16

[FMHR87] FISHER S. S., MCGREEVY M., HUMPHRIES J., ROBINETT W.: Virtual environment display system. In *Proceedings of the 1986 Workshop on Interactive 3D Graphics* (1987), I3D '86, ACM, pp. 77–87. 7

[FvdSH98] FISCHER M., VAN DER SMAGT P., HIRZINGER G.: Learning techniques in a dataglove based telemanipulation system for the dlr hand. In *IEEE International Conference on Robotics and Automation* (1998), vol. 2, IEEE, pp. 1603–1608. 7, 8

[GCF*04] GAO W., CHEN Y., FANG G., YANG C., JIANG D., GE C., WANG C.: HandTalker II: A Chinese sign language recognition and synthesis system. In *ICARCV'04* (2004), pp. 759–764. 17

[GFTC00] GRIFFIN W., FINDLEY R., TURNER M., CUTKOSKY M.: Calibration and mapping of a human hand for dexterous telemanipulation. In *ASME IMECE 2000 Symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems* (2000), pp. 1–8. 7, 8

[GKP04] GIBET S., KAMP J.-F., POIRIER F.: Gesture analysis: Invariant laws in movement. In *Gesture-Based Communication in Human-Computer Interaction*, Camurri A., Volpe G., (Eds.), vol. 2915 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2004, pp. 1–9. 16

[GL18] GRAY H., LEWIS W. H.: *Anatomy of the human body*, 20 ed. Philadelphia, Lea & Febiger, 1918. 3

[GMS*00] GAO W., MA J., SHAN S., CHEN X., ZENG W., ZHANG H., YAN J., WANG J.: Handtalker: A multimodal dialog system using sign language and 3-d virtual human. In *Advances in Multimodal Interfaces – ICMI 2000*, Tan T., Shi Y., Gao W., (Eds.), vol. 1948 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2000, pp. 564–571. 17

[GTT89] GOURRET J.-P., THALMANN N. M., THALMANN D.: Simulation of object and human skin formations in a grasping task. *Computer Graphics 23*, 3 (July 1989), 21–30. 13

[HBMTT95] HUANG Z., BOULIC R., MAGNENAT THALMANN N., THALMANN D.: Multi-sensor approach for grasping and 3D interaction. In *Proceedings Computer Graphics International '95, Academic Press* (1995), pp. 235–254. 11, 15

[HEK*07] HÜLSKEN F., ECKES C., KUCK R., UNTERBERG J., JÖRG S.: Modeling and animating virtual humans. *International Journal of Virtual Reality (IJVR) 6*, 4 (December 2007), 11–20. 7

[HGL*04] HU H., GAO X., LI J., WANG J., LIU H.: Calibrating human hand for teleoperating the hit/dlr hand. In *IEEE International Conference on Robotics and Automation* (2004), vol. 5, IEEE, pp. 4571–4576. 7, 8

[HK10] HELOIR A., KIPP M.: Real-time animation of interactive agents: Specification and realization. *Applied Artificial Intelligence 24*, 6 (July 2010), 510–529. 17

[HL10a] HUENERFAUTH M., LU P.: Accurate and accessible motion-capture glove calibration for sign language data collection. *ACM Transactions on Accessible Computing 3*, 1 (Sept. 2010), 2:1–2:32. 7

[HL10b] HUENERFAUTH M., LU P.: Modeling and synthesizing spatially inflected verbs for American sign language animations. In *Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility* (2010), ASSETS '10, ACM, pp. 99–106. 11, 18

[HMP02] HARTMANN B., MANCINI M., PELACHAUD C.: Formational parameters and adaptive prototype instantiation for MPEG-4 compliant gesture synthesis. In *Proceedings of the Computer Animation* (2002), CA '02, IEEE Computer Society, pp. 111–119. 16

[HMP06] HARTMANN B., MANCINI M., PELACHAUD C.: Implementing expressive gesture synthesis for embodied conversational agents. In *Proceedings of the 6th International Conference on Gesture in Human-Computer Interaction and Simulation* (2006), GW'05, Springer-Verlag, pp. 188–199. 16

[HRMO12a] HOYET L., RYALL K., MCDONNELL R., O'SULLIVAN C.: Sleight of hand: perception of finger motion from reduced marker sets. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2012), I3D '12, pp. 79–86. 4, 6

[HRMO12b] HOYET L., RYALL K., MCDONNELL R., O'SULLIVAN C.: Sleight of hand: Perception of finger motion from reduced marker sets. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (2012), I3D '12, ACM, pp. 79–86. 5

[HZY*11] HUANG H., ZHAO L., YIN K., QI Y., YU Y., TONG X.: Controllable hand deformation from sparse examples with rich details. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2011), SCA '11, ACM, pp. 73–82. 14

[ICL00] IP H. H. S., CHAN S. C. S., LAM M. S. W.: Hand gesture animation from static postures using an anatomy-based model. In *Proceedings of the International Conference on Computer Graphics* (2000), CGI '00, IEEE Computer Society, pp. 29–. 3, 5, 12

[IIO10] ISROZAIDI N., ISMAIL N., OSHITA M.: Data glove-based interface for real-time character motion control. In *ACM SIGGRAPH ASIA 2010 Posters* (2010), SA '10, ACM, pp. 5:1–5:1. 11

[Jac01] JACOBS J. P.: Refinements to the ergonomic model for keyboard fingering of parncutt, sloboda, clarke, raekallio, and desain. *Music Perception: An Interdisciplinary Journal 18*, 4 (2001), pp. 505–511. 18

[JHO10] JÖRG S., HODGINS J., O'SULLIVAN C.: The perception of finger motions. In *Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization* (2010), APGV '10, ACM, pp. 129–133. 1, 5

[JHS12] JÖRG S., HODGINS J., SAFONOVA A.: Data-driven finger motion synthesis for gesturing characters. *ACM Transactions on Graphics 31*, 6 (Nov. 2012), 189:1–189:7. 1, 9, 15

[JL11] JAIN S., LIU C. K.: Controlling physics-based characters using soft contacts. *ACM Transactions on Graphics 30* (Dec. 2011), 163:1–163:10. 13

[JLC10] JIN S., LI Y., CHEN W.: A novel dataglove calibration method. In *5th International Conference on Computer Science and Education* (2010), IEEE, pp. 1825–1829. 7, 8

[JO09] JÖRG S., O'SULLIVAN C.: Exploring the dimensionality of finger motion. In *Proceedings of the 9th Eurographics Ireland Workshop (EGIE 2009)* (December 2009), pp. 1–11. 5

[KB84a] KOCHANEK D. H. U., BARTELS R. H.: Interpolating splines for keyframe animation. In *Graphics Interface '84* (May 1984), pp. 41–42. 5

[KB84b] KOCHANEK D. H. U., BARTELS R. H.: Interpolating splines with local tension, continuity, and bias control. *Computer Graphics 18*, 3 (Jan. 1984), 33–41. 5

[KCMT00] KIM J., CORDIER F., MAGNENAT-THALMANN N.: Neural network-based violinist's hand animation. In *Computer Graphics International* (2000), pp. 37–41. 18

[KdR12] KIRCHHOF C., DE RUITER J.: On the audiovisual integration of speech and gesture. *Presented at the 5th Conference of the International Society for Gesture Studies (ISGS)* (2012). 17

[Ken80] KENDON A.: Gesticulation and speech: Two aspects of the process of utterance. *The Relationship of Verbal and Nonverbal Communication 25* (1980), 207–227. 15

[Ken04] KENDON A.: *Gesture – Visible Action as Utterance*. Cambridge University Press, Cambridge, UK, 2004. 1, 15

[KHW95] KESSLER G. D., HODGES L. F., WALKER N.: Evaluation of the CyberGlove as a whole-hand input device. *ACM Transactions on Computer-Human Interaction 2*, 4 (1995), 263–283. 7

[Kip04] KIPP M.: *Gesture Generation by Imitation - From Human Behavior to Computer Character Animation*. PhD thesis, Saarland University, Boca Raton, Florida, December 2004. 17

[KJP02] KRY P. G., JAMES D. L., PAI D. K.: Eigenskin: Real time large deformation character skinning in hardware. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2002), SCA '02, ACM, pp. 153–159. 14

[KKM*06] KOPP S., KRENN B., MARSELLA S., MARSHALL A. N., PELACHAUD C., PIRKER H., THÓRISSON K. R., VILHJÁLMSSON H.: Towards a common framework for multimodal generation: The behavior markup language. In *Proceedings of the 6th International Conference on Intelligent Virtual Agents* (2006), IVA'06, Springer, pp. 205–217. 17

[KL88] KRAMER J., LEIFER L.: The talking glove. *ACM SIGCAPH Computers and the Physically Handicapped*, 39 (April 1988), 12–16. 17

[KLG91] KRAMER J., LINDENER P., GEORGE W.: Communication system for deaf, deaf-blind, or non-vocal individuals using instrumented glove, Sept. 10 1991. US Patent 5,047,952. 17

[KM04] KURIHARA T., MIYATA N.: Modeling deformable human hands from medical images. In *Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2004), SCA '04, Eurographics Association, pp. 355–363. 13

[KMO*09] KUGIMOTO N., MIYAZONO R., OMORI K., FUJIMURA T., FURUYA S., KATAYOSE H., MIWA H., NAGATA N.: CG animation for piano performance. In *SIGGRAPH '09: Posters* (2009), SIGGRAPH '09, ACM, pp. 3:1–3:1. 18

[KNKA07] KIPP M., NEFF M., KIPP K. H., ALBRECHT I.: Towards natural gesture synthesis: Evaluating gesture units in a data-driven approach to gesture synthesis. In *Proceedings of the 7th International Conference on Intelligent Virtual Agents* (2007), pp. 15–28. 17

[KP06] KRY P. G., PAI D. K.: Interaction capture and synthesis. *ACM Transactions on Graphics 25*, 3 (2006), 872–880. 11, 13, 14, 15

[KTC04] KOPP S., TEPPER P., CASSELL J.: Towards integrated microplanning of language and iconic gesture for multimodal output. In *Proceedings of the 6th International Conference on Multimodal Interfaces* (2004), ICMI '04, ACM, pp. 97–104. 16

[KVV98] KITA S., VAN GIJN I., VAN DER HULST H.: Movement phase in signs and co-speech gestures, and their transcriptions by human coders. In *Proceedings of the International Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction* (1998), Springer-Verlag, pp. 23–35. 15, 16

[KW02] KOPP S., WACHSMUTH I.: Model-based animation of co-verbal gesture. In *Proceedings of Computer Animation* (June 2002), IEEE, pp. 252–257. 16

[KW04] KOPP S., WACHSMUTH I.: Synthesizing multimodal utterances for conversational agents: Research articles. *Computer Animation and Virtual Worlds 15*, 1 (Mar. 2004), 39–52. 17

[KW08] KITAGAWA M., WINDSOR B.: *MoCap for Artists: Workflow and Techniques for Motion Capture*. Focal Press, 2008. 6

[KWNZ12] KANG C., WHEATLAND N., NEFF M., ZORDAN V.: Automatic hand-over animation for free-hand motions from low resolution input. In *Motion in Games*, vol. 7660 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 244–253. 6, 10

[KY89] KAUFMAN A., YAGEL R.: Tools for interaction in three dimensions. In *Proceedings of the Third International Conference on Human-computer Interaction, Vol.1 on Work with Computers: Organizational, Management, Stress and Health Aspects* (1989), Elsevier Science Inc., pp. 468–475. 15

[KZK04] KAHLESZ F., ZACHMANN G., KLEIN R.: Visual-fidelity dataglove calibration. In *Computer Graphics International* (2004), pp. 403–410. 7, 8

[LCZ*11] LI Q., CHEN Y., ZHANG Y., LIU Y., ZHAO D.: Grasp gesture recognition based on rough sets theory in ar system. *International Journal of Digital Content Technology and its Applications 5*, 1 (2011), 226–234. 15

[Lea14] LEAP MOTION: Motion capture system website, http://www.leapmotion.com/, August 2014. 9

[LFP07] LI Y., FU J. L., POLLARD N. S.: Data-driven grasp synthesis using shape matching and task-based pruning. *IEEE Transactions on Visualization and Computer Graphics 13*, 4 (2007), 732–747. 15

[LH09] LU P., HUENERFAUTH M.: Accessible motion-capture glove calibration protocol for recording sign language data from deaf subjects. In *Proceedings of the 11th International ACM SIGACCESS Conference on Computers and Accessibility* (2009), ACM, pp. 83–90. 7

[LH10] LU P., HUENERFAUTH M.: Collecting a motion-capture corpus of American sign language for data-driven generation research. In *Proceedings of the NAACL HLT 2010 Workshop on Speech and Language Processing for Assistive Technologies* (2010), SLPAT '10, Association for Computational Linguistics, pp. 89–97. 17, 18

[LH11] LU P., HUENERFAUTH M.: Synthesizing American sign language spatially inflected verbs from motion-capture data. *Second International Workshop on Sign Language Translation and Avatar Technology (SLTAT), in conjunction with ASSETS* (2011). 17, 18

[Liu08] LIU C. K.: Synthesis of interactive hand manipulation. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008), pp. 163–171. 4, 13

[Liu09] LIU C. K.: Dextrous manipulation from a grasping pose. *ACM Transactions on Graphics 28*, 3 (2009), 3:1–3:6. 11, 13

[LKTK10] LEVINE S., KRÄHENBÜHL P., THRUN S., KOLTUN V.: Gesture controllers. *ACM Transactions on Graphics 29*, 4 (2010), 1–11. 10, 17

[LM06] LEE J., MARSELLA S.: Nonverbal behavior generator for embodied conversational agents. In *Proceedings of the 6th International Conference on Intelligent Virtual Agents* (2006), IVA'06, Springer-Verlag, pp. 243–255. 17

[LM13] LHOMMET M., MARSELLA S. C.: Gesture with meaning. In *Proceedings of the 13th International Conference on Intelligent Virtual Agents* (2013), Springer, pp. 303–312. 17

[LO96] LIANG R.-H., OUHYOUNG M.: A sign language recognition system using hidden markov model and context sensitive search. In *Proceedings of the ACM International Symposium on Virtual Reality and Softwre Technology* (1996), pp. 59–66. 10

[LS12] LOCKWOOD N., SINGH K.: Finger walking: Motion editing with contact-based hand performance. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2012), SCA '12, Eurographics Association, pp. 43–52. 11

[LST09] LEE S.-H., SIFAKIS E., TERZOPOULOS D.: Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics 28*, 4 (Sept. 2009), 99:1–99:17. 12

[LTK09] LEVINE S., THEOBALT C., KOLTUN V.: Real-time prosody-driven synthesis of body language. *ACM Transactions on Graphics 28*, 5 (Dec. 2009), 172:1–172:10. 10, 17

[LWH00] LIN J., WU Y., HUANG T. S.: Modeling the constraints of human hand motion. In *Proceedings of the Workshop on Human Motion (HUMO'00)* (2000), HUMO '00, IEEE Computer Society, pp. 121–126. 4, 11

[MBM*03] MENON A. S., BARNES B., MILLS R., BRUYNS C. D., TWOMBLY A., TWOMBLY E., SMITH J., MONTGOMERY K., BOYLE R.: Using registration, calibration, and robotics to build a more accurate virtual reality simulation for astronaut training and telemedicine. In *Proceedings of the 11th International Conference in Central Europe on Computer Graphics, Visualization, and Computer (WSCGo03)* (2003). 7, 8

[McN92] MCNEILL D.: *Hand and Mind: what gestures reveal about thought*. The University of Chicago Press, 1992. 15

[McN05] MCNEILL D.: *Gesture and thought*. University of Chicago Press, 2005. 16

[Mea14] MEASURAND: www.shapehand.com, August 2014. 7

[Men99] MENACHE A.: *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999. 5, 7

[MK02] MEHDI S., KHAN Y.: Sign language recognition using sensor gloves. In *Proceedings of the 9th International Conference on Neural Information Processing* (Nov 2002), vol. 5, pp. 2204–2206. 10

[MKCA03] MILLER A. T., KNOOP S., CHRISTENSEN H. I., ALLEN P. K.: Automatic grasp planning using shape primitives. In *In Proceedings of the 2003 IEEE International Conference on Robotics and Automation* (2003), pp. 1824–1829. 14

[MKMK05] MIYATA N., KOUCH M., MOCHIMARU M., KURIHARA T.: Finger joint kinematics from MR images. In *IEEE/RSJ International Conference on Intelligent Robots and Systems* (Aug 2005), pp. 2750–2755. 4

[MMT97] MOCCOZET L., MAGNENAT-THALMANN N.: Dirichlet free-form deformations and their application to hand simulation. In *Computer Animation '97* (Jun 1997), pp. 93–102. 13

[MNBP08] MANCINI M., NIEWIADOMSKI R., BEVACQUA E., PELACHAUD C.: Greta: a saiba compliant eca system. In *Troisième Workshop sur les Agents Conversationnels Animés* (Paris, France, November 2008), WACA '08. 17

[MT91] MURAKAMI K., TAGUCHI H.: Gesture recognition using recurrent neural networks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (1991), CHI '91, ACM, pp. 237–242. 10

[MTA*01] MCDONALD J., TORO J., ALKOBY K., BERTHIAUME A., CARTER R., CHOMWONG P., CHRISTOPHER J., DAVIDSON M. J., FURST J., KONIE B., LANCASTER G., ROYCHOUDHURI L., SEDGWICK E., TOMURO N., WOLFE R.: An improved articulated model of the human hand. *The Visual Computer 17*, 3 (2001), 158–166. 3

[MTLT88] MAGNENAT-THALMANN N., LAPERRIRE R., THALMANN D.: Joint-dependent local deformations for hand animation and object grasping. In *In Proceedings on Graphics Interface* (1988), pp. 26–33. 13, 14

[MTT87] MAGNENAT-THALMANN N., THALMANN D.: The direction of synthetic actors in the film rendez-vous a montreal. *IEEE Computer Graphics and Applications 7*, 12 (Dec. 1987), 9–19. 13

[MZF06] MAJKOWSKA A., ZORDAN V. B., FALOUTSOS P.: Automatic splicing for hand and body animations. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2006), pp. 309–316. 9, 10

[Nap80] NAPIER J.: *Hands*. New York: Pantheon Books, Princeton, NJ, 1980. 2

[Neu15] NEURON P.: Motion capture system website, http://perceptionmocap.com/, Janurary 2015. 8

[NF02] NEFF M., FIUME E.: Modeling tension and relaxation for computer animation. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2002), SCA '02, ACM, pp. 81–88. 16

[NF05] NEFF M., FIUME E.: AER: Aesthetic exploration and refinement for expressive character animation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2005), SCA '05, ACM, pp. 161–170. 16

[Nim15] NIMBLE VR: Motion capture system website, http://nimblevr.com//, January 2015. 9

[NKAS08] NEFF M., KIPP M., ALBRECHT I., SEIDEL H.-P.: Gesture modeling and animation based on a probabilistic recreation of speaker style. *ACM Transactions on Graphics 27*, 1 (2008), 1–24. 2, 5, 11, 16, 17

[NS06] NEFF M., SEIDEL H.-P.: Modeling relaxed hand shape for character animation. In *Articulated Motion and Deformable Objects*, vol. 4069 of *Lecture Notes in Computer Science*. Springer, 2006, pp. 262–270. 12

[NTB*11] NEFF M., TOOTHMAN N., BOWMANI R., TREE J. E. F., WALKER M. A.: Don't scratch! Self-adaptors reflect emotional stability. In *Proceedings of the 10th International Conference on Intelligent Virtual Agents* (2011), Springer, pp. 398–411. 2

[NWAW10] NEFF M., WANG Y., ABBOTT R., WALKER M.: Evaluating the effect of gesture and language on personality perception in conversational agents. In *Proceedings of the 10th International Conference on Intelligent Virtual Agents* (2010), Springer, pp. 222–235. 2

[Opt14] OPTITRACK: Motion capture system website, https://www.naturalpoint.com/optitrack/, July 2014. 6

[Pha14] PHASESPACE: Motion capture system website, http://www.phasespace.com/, August 2014. 6

[PL14] PLANCAK M., LUZANIN O.: Hand gesture recognition using low-budget data glove and cluster-trained probabilistic neural network. *Assembly Automation 34*, 1 (2014), 94–105. 11

[PS12] PALASTANGA N., SOAMES R.: *Anatomy and Human Movement – Structure and Function*, 6 ed. Butterworth Heinemann Elsevier, 2012. 2, 3

[PSC*97] PARNCUTT R., SLOBODA J. A., CLARKE E. F., RAEKALLIO M., DESAIN P.: An ergonomic model of keyboard fingering for melodic fragments. *Music Perception: An Interdisciplinary Journal 14*, 4 (1997), 341–382. 18

[PZ05] POLLARD N. S., ZORDAN V. B.: Physically based grasping control from example. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2005), pp. 311–318. 4, 11, 12, 15

[Qua14] QUALISYS: Motion capture system website, http://www.qualisys.com/, August 2014. 6

[QWAB89] QUAM D. L., WILLIAMS G. B., AGNEW J. R., BROWNE P. C.: An experimental determination of human hand accuracy with a dataglove. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (1989), vol. 33, SAGE Publications, pp. 315–319. 7

[Ree81] REEVES W. T.: Inbetweening for computer animation utilizing moving point constraints. *Computer Graphics 15*, 3 (Aug. 1981), 263–269. 5

[RG91] RIJPKEMA H., GIRARD M.: Computer animation of knowledge-based human grasping. In *Proceedings of the 18th Annual Conference on Computer graphics and Interactive Techniques* (1991), SIGGRAPH '91, ACM, pp. 339–348. 4, 11, 14

[RNL06] RHEE T., NEUMANN U., LEWIS J. P.: Human hand modeling from surface anatomy. In *Proceedings of the 2006 Symposium on Interactive 3D Graphics and Games* (2006), I3D '06, ACM, pp. 27–34. 13

[SAI12] SAIBA: Working group website, 2012. http://www.mindmakers.org/projects/saiba/wiki. 15

[Say89] SAYEGH S. I.: Fingering for string instruments with the optimum path paradigm. *Computer Music Journal 13*, 3 (1989), pp. 76–84. 18

[SC92] STEWART A. J., CREMER J. F.: Beyond keyframing: An algorithmic approach to animation. In *Proceedings of the Conference on Graphics Interface '92* (1992), Morgan Kaufmann Publishers Inc., pp. 273–281. 11

[SCSL02] SIMÉON T., CORTES J., SAHBANI A., LAUMOND J. P.: A manipulation planner for pick and place operations under continuous grasps and placements. In *In Proceedings of the IEEE International Conference on Robotics and Automation* (2002), pp. 2022–2027. 11

[SDO*04] STONE M., DECARLO D., OH I., RODRIGUEZ C., STERE A., LEES A., BREGLER C.: Speaking with hands: Creating animated conversational characters from recordings of human performance. *ACM Transactions on Graphics 23*, 3 (2004), 506–513. 10, 16, 17

[SFC*11] SHOTTON J., FITZGIBBON A., COOK M., SHARP T., FINOCCHIO M., MOORE R., KIPMAN A., BLAKE A.: Real-time human pose recognition in parts from single depth images. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition* (2011), CVPR '11, IEEE, pp. 1297–1304. 9

[SFS98] SANTELLO M., FLANDERS M., SOECHTING J. F.: Postural hand synergies for tool use. *The Journal of Neuroscience 18*, 23 (1998), 10105–10115. 4, 5, 11

[SG06] SHANKAR V., GHOSH D.: Dynamic hand gesture synthesis and animation using image morphing technique. *IET Conference Proceedings* (January 2006), 543–548(5). 5

[SHSvdS14] STILLFRIED G., HILLENBRAND U., SETTLES M., VAN DER SMAGT P.: MRI-based skeletal hand movement model. In *The Human Hand as an Inspiration for Robot Hand Development*, Balasubramanian R., Santos V. J., (Eds.), vol. 95 of *Springer Tracts in Advanced Robotics*. Springer International Publishing, 2014, pp. 49–75. 4

[SKP08] SUEDA S., KAUFMAN A., PAI D. K.: Musculotendon simulation for hand animation. *ACM Transactions on Graphics 27*, 3 (Aug. 2008), 3:1–3:8. 3, 11, 12

[SKSK07] SCHLATTMANN M., KAHLESZ F., SARLETTE R., KLEIN R.: Markerless 4 gestures 6 DOF real-time visual tracking of the human hand with automatic initialization. *Computer Graphics Forum 26*, 3 (Sept. 2007), 467–476. 9

[SMFW04] SUDDERTH E. B., M M. I., FREEMAN W. T., WILLSKY A. S.: Distributed occlusion reasoning for tracking with nonparametric belief propagation. In *In Neural Information Processing Systems* (2004), MIT Press, pp. 1369–1376. 9

[SMR11] STEFFEN J., MAYCOCK J., RITTER H.: Robust dataglove mapping for recording human hand postures. *Intelligent Robotics and Applications* (2011), 34–45. 7, 8

[SOT13] SRIDHAR S., OULASVIRTA A., THEOBALT C.: Interactive markerless articulated hand motion tracking using RGB and depth data. In *IEEE International Conference on Computer Vision* (2013), IEEE, pp. 2456–2463. 9

[SP14] SUEDA S., PAI D.: Hand simulation models in computer graphics. *Journal of the Society of Biomechanisms 38*, 1 (2014). 11

[SRWG98] SOMIA N., RASH G., WACHOWIAK M., GUPTA A.: The initiation and sequence of digital joint motion: A three-dimensional motion analysis. *The Journal of Hand Surgery: Journal of the British Society for Surgery of the Hand 23*, 6 (Dec. 1998), 792–795. 4, 11

[ST94] SANSO R. M., THALMANN D.: A hand control and automatic grasping system for synthetic actors. In *Computer Graphics Forum* (1994), vol. 13, pp. 167–77. 4, 14

[Syn15] SYNERTIAL: Motion capture system website, http://www.synertial.com/products/glove-7-sensor/, January 2015. 8

[SZ94] STURMAN D. J., ZELTZER D.: A survey of glove-based input. *IEEE Computer Graphics and Applications 14*, 1 (1994), 30–39. 7, 15

[SZP89] STURMAN D. J., ZELTZER D., PIEPER S.: Hands-on interaction with virtual environments. In *Proceedings of the 2nd Annual ACM SIGGRAPH Symposium on User Interface Software and Technology* (1989), UIST '89, ACM, pp. 19–24. 7, 15

[TAH*04] THEOBALT C., ALBRECHT I., HABER J., MAGNOR M., SEIDEL H.-P.: Pitching a baseball: Tracking high-speed motion with multi-exposure images. *ACM Transactions on Graphics 23*, 3 (August 2004), 540–547. 8

[TBHF03] TERAN J., BLEMKER S., HING V. N. T., FEDKIW R.: Finite volume methods for the simulation of skeletal muscle. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2003), SCA '03, Eurographics Association, pp. 68–74. 11, 12

[TBK87] TOMOVIC R., BEKEY G. A., KARPLUS W. J.: A strategy for grasp synthesis with multifingered robot hands. In *IEEE International Conference on Robotics and Automation* (Mar 1987), vol. 4, pp. 83–89. 14

[TH09a] TANG Y.-M., HUI K.-C.: Simulating tendon motion with axial mass-spring system. *Computers & Graphics 33*, 2 (2009), 162 – 172. 12

[TH09b] TAYLOR G. W., HINTON G. E.: Factored conditional restricted boltzmann machines for modeling motion style. In *Proceedings of the 26th Annual International Conference on Machine Learning* (2009), ACM, pp. 1025–1032. 16

[THR06] TAYLOR G. W., HINTON G. E., ROWEIS S. T.: Modeling human motion using binary latent variables. In *Advances in neural information processing systems* (2006), pp. 1345–1352. 16

[TMMK08] THIEBAUX M., MARSELLA S., MARSHALL A. N., KALLMANN M.: Smartbody: Behavior realization for embodied conversational agents. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 1* (Richland, SC, 2008), AAMAS '08, pp. 151–158. 17

[TSF05] TSANG W., SINGH K., FIUME E.: Helping hand: An anatomically accurate inverse dynamics solution for unconstrained hand motion. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2005), pp. 319–328. 3, 4, 12

[TSLP14] TOMPSON J., STEIN M., LECUN Y., PERLIN K.: Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transaction on Graphics 33*, 5 (2014), 169:1–169:10. 9

[UKS89] UNO Y., KAWATO M., SUZUKI R.: Formation and control of optimal trajectory in human multijoint arm movement - minimum torque-change model. *Biological Cybernetics 61*, 2 (1989), 89–101. 16

[VCC*07] VILHJÁLMSSON H., CANTELMO N., CASSELL J., E. CHAFAI N., KIPP M., KOPP S., MANCINI M., MARSELLA S., MARSHALL A. N., PELACHAUD C., RUTTKAY Z., THÓRISSON K. R., WELBERGEN H., WERF R. J.: The behavior markup language: Recent developments and challenges. In *Proceedings of the 7th International Conference on Intelligent Virtual Agents* (2007), IVA '07, Springer-Verlag, pp. 99–111. 17

[Vic14] VICON: Motion capture system website, http://www.vicon.com, July 2014. 6

[VT82] VIVIANI P., TERZUOLO C.: Trajectory determines movement dynamics. *Neuroscience 7*, 2 (1982), 431 – 437. 16

[vWRRZ10] VAN WELBERGEN H., REIDSMA D., RUTTKAY Z. M., ZWIERS J.: Elckerlyc - A BML Realizer for continuous, multimodal interaction with a virtual human. *Journal on Multimodal User Interfaces 3*, 4 (August 2010), 271–284. 16, 17

[Wal98] WALLBOTT H. G.: Bodily expression of emotion. *European Journal of Social Psychology 28*, 6 (1998), 879–896. 2

[WBH*08] WILLIAMS G., BREGLER C., HACKNEY P., ROSENTHAL S., MCDOWALL I., SMOLSKIY K.: *Body Signature Recognition*. Tech. Rep. TR-2008-915, New York University, 2008. 2

[WFH08] WANG J. M., FLEET D. J., HERTZMANN A.: Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence 30*, 2 (2008), 283–298. 16

[WG89] WEIMER D., GANAPATHY S. K.: A synthetic visual environment with hand gesturing and voice input. *SIGCHI Bulletin 20*, SI (Mar. 1989), 235–240. 7

[WGS*90] WISE S., GARDNER W., SABELMAN E., VALAINIS E., WONG Y., GLASS K., DRACE J., ROSEN J. M.: Evaluation of a fiber optic glove for semi-automated goniometric measurements. *Journal of Rehabilitation Research and Development 27*, 4 (1990), 411–424. 7

[WJZ13] WHEATLAND N., JÖRG S., ZORDAN V.: Automatic hand-over animation using principle component analysis. In *Proceedings of Motion on Games* (2013), MIG '13, ACM, pp. 175:197–175:202. 5, 6, 10, 18

[WN13a] WANG Y., NEFF M.: Data-driven glove calibration for hand motion capture. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2013), SCA '13, ACM, pp. 15–24. 7, 8

[WN13b] WANG Y., NEFF M.: The influence of prosody on the requirements for gesture-text alignment. In *Proceedings of the 13th International Conference on Intelligent Virtual Agents* (2013), Springer, pp. 180–188. 17

[WP09] WANG R. Y., POPOVIĆ J.: Real-time hand-tracking with a color glove. *ACM Transactions on Graphics 28*, 3 (2009), 1–8. 8, 10, 11, 12

[WPP11] WANG R. Y., PARIS S., POPOVIĆ J.: 6D hands: markerless hand-tracking for computer aided design. In *ACM Symposium on User Interface Software and Technology* (2011), pp. 549–558. 8

[WS99] WEISSMANN J., SALOMON R.: Gesture recognition for virtual reality applications using data gloves and neural networks. In *International Joint Conference on Neural Networks* (1999), vol. 3, pp. 2043–2046 vol.3. 10

[Yaz09] YAZADI F.: Cyberglove systems: CyberGlove II wireless data glove user guide. CyberGlove Systems LLC, 2009. 7

[YKH04] YAMANE K., KUFFNER J. J., HODGINS J. K.: Synthesizing animations of human manipulation tasks. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 532–539. 11

[YL12] YE Y., LIU C. K.: Synthesis of detailed hand manipulations using contact sampling. *ACM Transactions on Graphics 31*, 4 (2012). 11, 15

[YZ05] YANG X. S., ZHANG J. J.: Modelling and animating hand wrinkles. In *Proceedings of the 5th International Conference on Computational Science - Volume Part II* (2005), ICCS'05, Springer-Verlag, pp. 199–206. 13

[Zan79] ZANCOLLI E.: *Structural and dynamic bases of hand surgery*, 2 ed. Lippincott, Philadelphia, Pennsylvania, 1979. 4

[ZCX12] ZHAO W., CHAI J., XU Y.-Q.: Combining marker-based mocap and RGB-D camera for acquiring high-fidelity hand motion data. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2012). 4, 9, 11

[Zim85] ZIMMERMAN T.: Optical flex sensor, Sept. 17 1985. US Patent 4,542,291. 7

[ZRHN13] ZHU Y., RAMAKRISHNAN A. S., HAMANN B., NEFF M.: A system for automatic animation of piano performances. *Computer Animation and Virtual Worlds 24*, 5 (2013), 445–457. 1, 11, 18

[ZZMC13] ZHAO W., ZHANG J., MIN J., CHAI J.: Robust realtime physics-based motion control for human grasping. *ACM Transactions on Graphics 32*, 6 (Nov. 2013), 207:1–207:12. 4