Teaching American Sign Language in Mixed Reality

QIJIA SHAO, Department of Computer Science, Dartmouth College
AMY SNIFFEN, Department of Computer Science, Dartmouth College
JULIEN BLANCHET, Department of Computer Science, Dartmouth College
MEGAN E. HILLIS, Department of Psychological and Brain Sciences, Dartmouth College
XINYU SHI, Department of Informatics, Xiamen University
THEMISTOKLIS K. HARIS, Department of Computer Science, Dartmouth College
JASON LIU, Department of Education, Dartmouth College
JASON LAMBERTON, Motion Light Lab, Gallaudet University
MELISSA MALZKUHN, Motion Light Lab, Gallaudet University
LORNA C. QUANDT, Educational Neuroscience Program, Gallaudet University
JAMES MAHONEY, Department of Computer Science, Dartmouth College
DAVID J. M. KRAEMER, Department of Education, Dartmouth College
XIA ZHOU, Department of Computer Science, Dartmouth College
DEVIN BALKCOM, Department of Computer Science, Dartmouth College

This paper presents a holistic system to scale up the teaching and learning of vocabulary words of American Sign Language (ASL). The system leverages the most recent mixed-reality technology to allow the user to perceive her own hands in an immersive learning environment with first- and third-person views for motion demonstration and practice. Precise motion sensing is used to record and evaluate motion, providing real-time feedback tailored to the specific learner. As part of this evaluation, learner motions are matched to features derived from the Hamburg Notation System (HNS) developed by sign-language linguists. We develop a prototype to evaluate the efficacy of mixed-reality-based interactive motion teaching. Results with 60 participants show a statistically significant improvement in learning ASL signs when using our system, in comparison to traditional desktop-based, non-interactive learning. We expect this approach to ultimately allow teaching and guided practice of thousands of signs.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing systems and tools**; • **Computer systems organization → Embedded systems**.

Additional Key Words and Phrases: Motion teaching, American Sign Language, Mixed reality.

Authors' addresses: Qijia Shao, Department of Computer Science, Dartmouth College, Qijia.Shao.GR@dartmouth.edu; Amy Sniffen, Department of Computer Science, Dartmouth College; Julien Blanchet, Department of Computer Science, Dartmouth College; Megan E. Hillis, Department of Psychological and Brain Sciences, Dartmouth College; Xinyu Shi, Department of Informatics, Xiamen University; Themistoklis K. Haris, Department of Computer Science, Dartmouth College; Jason Liu, Department of Education, Dartmouth College; Jason Lamberton, Motion Light Lab, Gallaudet University; Melissa Malzkuhn, Motion Light Lab, Gallaudet University; Lorna C. Quandt, Educational Neuroscience Program, Gallaudet University; James Mahoney, Department of Computer Science, Dartmouth College; David J. M. Kraemer, Department of Education, Dartmouth College; Xia Zhou, Department of Computer Science, Dartmouth College; Devin Balkcom, Department of Computer Science, Dartmouth College.

2474-9567/2020/12-ART152 $15.00
https://doi.org/10.1145/3432211

ACM Reference Format:
Qijia Shao, Amy Sniffen, Julien Blanchet, Megan E. Hillis, Xinyu Shi, Themistoklis K. Haris, Jason Liu, Jason Lamberton, Melissa Malzkuhn, Lorna C. Quandt, James Mahoney, David J. M. Kraemer, Xia Zhou, and Devin Balkcom. 2020. Teaching American Sign Language in Mixed Reality. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4, Article 152 (December 2020), 27 pages. https://doi.org/10.1145/3432211

1 INTRODUCTION

Teaching motion is hard. Precise or rapid body motion is hard to observe, explain, and execute. The marvelous human body affords greater control and has more degrees of freedom than any humanoid robot, with adaptable learning capabilities and wonderful sensing ability. However, our interfaces to the human body are limited. Manipulation of the body by another human intrudes on the privacy of the learner, voice is typically imprecise for communicating physical motion, and un-augmented vision requires the learner to interpret and remap an external motion onto their own body.

Intensive human coaching can be effective, but coaches are often in short supply. How can motion demonstration, sensing, and feedback be technologically augmented effectively? Online learning platforms such as Khan Academy, Coursera, and Codecademy provide effective examples of the benefits of teaching at scale, but remain limited by the communication medium. They are unable to extend population-scale teaching beyond the computer screen and keyboard into the physical world for learning of physical motion tasks.

Sign language serves as a concrete challenge for automated assistance for teaching human motion. In American Sign Language (ASL), individual signs involve coordinated finger and hand motions to communicate smoothly and naturally; learning these motions is necessarily time-intensive. Expert teachers are not widely available, and standard learning videos and online ASL courses are not tailored to individual learners and lack real-time interactive feedback.

In this paper, we set out to address the challenge of providing automated assistance for teaching vocabulary words[1] in ASL. We leverage the latest mixed reality (MR) technology to create an immersive learning environment, where learners can perceive their own hands while following third- and first-person demonstrations to practice ASL sign motions. The learning environment is also interactive, augmented with continuous sensing that monitors the motion of an individual learner. This allows the system to provide tailored, real-time feedback to correct any motion errors on the fly and boost learning efficiency. We choose MR over traditional desktop displays because of the immersiveness of the MR environment. The head-mounted display of MR also allows the learning environment to be readily available in front of a learner's eyes and viewed from the right perspective. The MR display can potentially extend the system to teach a wider range of motions in diverse scenarios where a desktop display is inconvenient to carry or ineffective for teaching due to a fixed point of view. We choose MR over virtual reality (VR) because MR allows the learner to perceive her own hands, which is beneficial for learning motions. We did not choose augmented reality (AR) in this study because of the sub-optimal user experience offered by current AR technology (e.g., limited field-of-view, semi-translucent rendered objects).

Developing such a system presents challenges on two fronts. *First*, on the front of teaching methodology, a key challenge lies in designing a scalable teaching approach to deal with the large volume of ASL signs. Approaches that rely on a limited database of hard-coded signs cannot be easily expanded to new signs. Additionally, it is hard to identify relevant features of ASL signs and determine the accuracy of the motion. Although machine learning classifiers can be used to extract features and recognize whole signs, extracted features which are meaningful to algorithms often turn out to be hard to interpret for learners. Whole-sign classifiers are sensitive to overfitting of a training dataset with limited signs and are difficult to scale up to new signs. *Second*, on the system front, scaling up ASL teaching requires the system to be lightweight and portable while being capable of robustly and accurately sensing hand and finger motion to provide real-time feedback. Existing motion sensing techniques,

[1]We focus on teaching individual signs in this work as the first step, we plan to teach ASL grammar and fully signed sentences as future work.

however, either require heavy instrumentation of the environment (*e.g.*, setting up multiple cameras), or suffer from occlusion that can occur regularly during the performing of ASL signs. Current techniques are also unable to sense all types of motion related to ASL signs, including both fine-grained finger motion and coarse-grained hand motion and its position relative to the head.

We overcome these challenges as follows. (1) To provide useful human-interpretable feedback for a wide variety of signs, we build on Hamburg Notation System (HNS) [19], a generalized notation system focused on the fundamental components of sign language. Much as the international phonetic alphabet factors spoken words into phonetic units independent of specific languages, HNS factors individual signs into a reduced set of fundamental components describing physical movement. Focusing on the small set of HNS primitive features allows the system to expand the ASL dictionary indefinitely. These features are also intuitive and easily interpretable, providing a basis for computing effective feedback during learning and practice. (2) We design a portable motion sensing system as a custom-built glove that senses the HNS features while eliminating issues of occlusion. We embed lightweight, thin flex and force sensors into the glove for sensing the flexion and extension of finger joints and computing joint angles, as well as the contact of adjacent fingers and finger tips. Exploiting the front-facing camera of the MR headset, we also add visual tags to the glove to facilitate the tracking of coarse-grained hand motion and its relative position to the headset. (3) We develop systematic algorithmic solutions to translate raw sensing data into HNS features and compute effective feedback. We specifically address the large number of possible configurations arising from various combinations of base handshapes and modifiers. Instead of building the translation on the level of the final handshape, we propose a layered approach that sequentially identifies modifiers and base handshapes to significantly reduce the translation overhead. Qualitative, descriptive feedback is then presented based on the discrepancy at the level of HNS features for users to effectively understand.

We build a system prototype as a proof-of-concept to evaluate the efficacy of MR-based interactive motion teaching. Our prototype consists of an MR headset – specifically, an HTC VivePro headset [23] with an attached Zed Mini Camera [54] – and an in-house custom built glove system for motion sensing. We conduct a user study with 60 novice users, aiming to teach them twelve ASL signs. These users are divided into four groups of equal size. One group learns ASL signs via our MR system, while the other three learn the same lesson via a desktop version. We obtain the following key findings when comparing our system to video-based, non-interactive learning:

- A mixed-design ANOVA reveals a significant main effect of learning group on performance ($F_{3,56} = 29.05$, $p < 0.001$; η^2=0.609), indicating a statistically significant difference in teaching effectiveness between MR-based teaching and desktop-based, non-interactive teaching.
- Post-hoc t-tests show that our system is more helpful in teaching both trajectory following and detailed handshapes.
- The immersive learning environment contributes more to learning the orientation and movement, and real-time feedback contributes substantially to learning the handshape.
- Self-reported results indicate that the learners are much more engaged when learning in the MR environment than with the desktop.

Contributions. We summarize our contributions as follows.

- We proposed a new methodology that exploits the HNS representation of ASL to decompose ASL signs into a small set of primitives for teaching. This bottom-up approach sets a fundamental departure from prior works on ASL classification, which all commonly deal with ASL signs in a top-down manner and offer poor scalability by requiring extensive training data to cover a large number of signs.
- To the best of our knowledge, the ASL-HNS dictionary, which we have built based on online videos and other ASL resources, is the first-of-its-kind and will be of use to other researchers whose work requires a standardized method for documenting ASL as a means of studying the language itself, such as regional dialects or gestural

variation between fluent signers. As our system develops further, the potential for other uses in the future will expand even more. We released the ASL-HNS dictionary to the research community for broader use[2].
- We designed a portable sensing system as a custom-built glove that senses the HNS features while eliminating issues of occlusion. Systematic algorithmic solutions were also developed to translate raw sensing data into HNS features and compute effective feedback.
- We have built a system prototype using off-the-shelf hardware components and existing MR technology and designed a learning lesson as a proof of concept.
- We demonstrated the efficacy of MR-based motion teaching via a user study with 60 novice ASL learners and analyzed individual contributions of the immersiveness and interactivity provided by our system by comparing it to desktop counterparts.

2 RELATED WORK

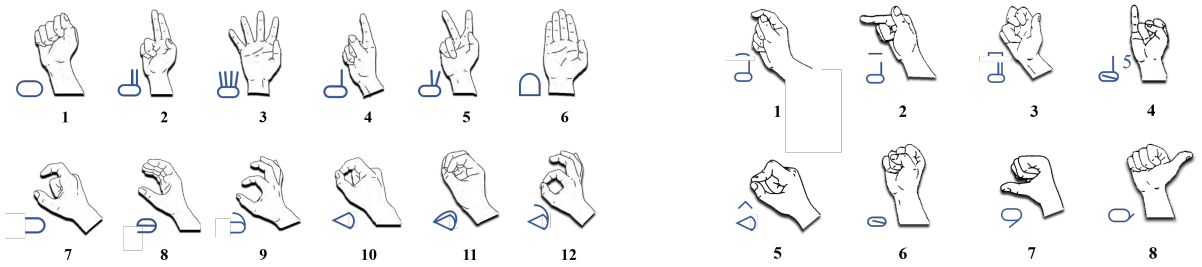
Sign Language Recognition. Most existing hardware and software systems for working with ASL have focused on recognition and translation. We categorize existing sign language recognition work based on whether they used on-body sensors.

1) *w/ on-body sensors*: Leap Motion has been widely used to perform sign language recognition [8, 9, 11, 31, 39, 40]: Du et al. [11] achieved a 99.42% accuracy rate when recognizing 10 digits with an SVM classifier. Mapari et al. proposed an ASL recognition system with the ability to recognize 32 letters and digits ('J', 'Z', '2', and '6' are excluded from the system) with an accuracy rate of 90%. Chong et al. [8] presented a system that can recognize all 26 letters and 10 digits with an accuracy rate of 72.79% for an SVM and 88.79% for the DNN (deep neural network). These systems are all based on Leap Motion, which limits their potential of non-line-of-sight recognition, while our system solves this problem with our sensing gloves. Hou et al. [22] proposed a real-time ASL recognition system with a smartwatch and a smartphone. The system shows the ability to recognize 103 words separately with a detection rate of 99.2% and at the sentence level with a word error rate of 1.04% on average. Zhu et at. [68] presented an epidermal-iontronic sensing (EIS) system based on a wearable device that is worn on finger joints for to recognize 35 ASL fingerspellings. The recognition accuracy was 99.6% within-user, and was 76.1% across users. In contrast to these systems, our ASL teaching system is built on HNS and focuses on a small set of HNS primitive features, allowing our system the potential to recognize an indefinite number of ASL signs with extremely low training overhead.

2) *w/o on-body sensors*: The other category of works aims to recognize ASL in a non-intrusive way (with no wearable device). There is some work using WiFi signals to recognize ASL: 5 hand gestures in [52], 9 digit postures in [35], 25 hand/finger gestures in [43] and 276 hand gestures in [38]. SignFi [38] feeds the WiFi signals' channel state information to a Convolutional Neural Network (CNN) and is able to recognize 150 sign gestures with an 86.66% accuracy rate. 60 GHz millimeter-wave (mmWave) has recently emerged as a viable method for ASL recognition. mmASL[51] is a mmWave-based recognition system and reports an ability of recognizing 50 ASL signs with an 87% average accuracy. While innovative, these approaches to sensing ASL without on-body sensors rely on neural network learning algorithms to recognize whole ASL signs. This entails a daunting training overhead to expand to a large volume of ASL signs, and relies on features that may be difficult for learners to interpret. Our HNS-based system provides intuitive and easily interpretable features, which are the basis for computing effective feedback during learning and practice.

Motion Teaching. The other related line of work is motion teaching. While some existing works leverage AR/VR technology to build a learning environment, other works teach motion using only a screen or their self-designed systems. We next review the representative works in each category.

[2]https://github.com/QijiaShao/ASL-HNS-dictionary



(a) Base handshapes (b) Modifiers

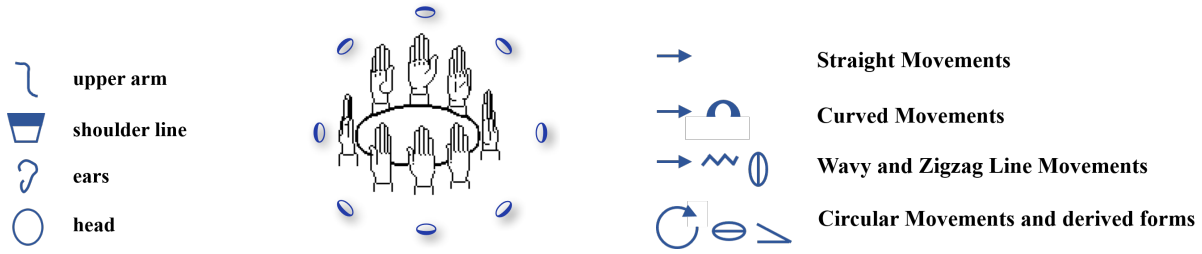
Fig. 1. Handshape feature can be further divided into base handshapes and modifiers [19].

1)*w/o AR/VR:* Motion teaching has been done for the specific application of dancing [3, 30]. [30] focused on using a motion capture system to evaluate a user's dance performance by comparing their performance to the examples. Since the comparison is done offline after the user has finished dancing, this system does not provide real-time feedback to the user. A more general training system, YouMove, uses recorded data to train users by a large-scale augmented reality mirror [3]. During training, YouMove specifies the key frames for the movement and provide real-time feedback on a user's errors. Tagami et.al [55] proposed a motion teaching system to transmit a teacher's motion to learners directly by using motion capture and an assistive suit. Since both systems used the Kinect as the sensing system to capture the body, occlusion limits the potential to sense movements with higher DoF. [55] also requires a teacher to be present with the learner at the same time. In contrast to those systems, our system leverages lightweight, wearable sensors to sense finger movement with high DoF and an MR environment to teach learners ASL without the need of a professional signer. Xia et al.[66] created a feedback mechanism involving a row of actuators that push the user's fingers up and down to cover or uncover holes on a flute in accordance with notes of a musical phrase. Our work differs from work in this category by using MR technology to create an immersive learning environment with first- and third-person motion demonstration, practice, and real-time feedback.

2)*w/ AR/VR:* Recent work [15, 41, 49] embeds VR/AR technologies into neurosurgey resident training and clinical enhancements by simulating a real-world environment for users to experience, allowing training in a low-risk environment. Feedback components of these systems are fairly limited. Traditional stroke rehabilitation lacks timely feedback of a patient's actions and cannot offer effective advice to improve training. Oagaz et. al [46] proposed a system that uses the patient's motion information collected from smart insoles in their shoes to provide the input for a VR application that performs the corresponding exercise animations. Pei et. al [48] used the Kinect for motion capture and achieved the real-time rehabilitation motion feedback by customized skeleton modeling and virtual character constructing in Unity3D. The feedback they provided, however, was limited to the binary right or wrong information or scores of the trainee's motions. In this paper, our system provides both high-level right or wrong feedback and detailed instructions guiding users to perform a correct sign in real-time.

3 HAMBURG NOTATION SYSTEM

In order to provide useful feedback to a learner, the system must evaluate the accuracy of the learner's motion. This raises a key question: accurate compared to what? There may be significant variability in how different experts perform the same sign, for reasons ranging from hand geometry, to the mood the speaker intends to communicate. Fortunately, several decades of research and development have been applied to the issue of notating sign language into a written format; these notation systems provide a useful way of evaluating the correctness of components of a sign.



(a) (b) (c)

Fig. 2. Examples of (a) location, (b) orientation, (c) movement features and their corresponding HNS symbols [19].

HNS Background. *Hamburg Notation System* (HNS) is the collaborative effort of researchers across several nations to develop a nomenclature that can effectively describe all sign languages in written form. HNS provides a means to fractionate individual sign language signs into a reduced set of fundamental component elements that describe physical movements [19]. Thus, each sign in ASL, or any sign language, can be described in terms of 5 component features:

- *Handshape*: HNS defines twelve basic handshapes (Fig. 1a) as the basis of ASL signs. In addition to the base handshape, signs may require finger modifiers. Modifiers target specific fingers to transform base handshapes into signs [18]. There are 8 modifiers (Fig. 1b), with two classes: thumb modifiers and finger modifiers [19]. Thumb modifiers specify which way the thumb should point during a sign. Finger modifiers indicate whether the fingers should be straight, bent, hooked, or flattened. [17].
- *Location*: Hand location is defined by two components, the location of the hand within the frontal plane, and on the z-axis with respect to the body. If one or both of these components are not specified, the hand is assumed to be located in the "neutral signing space", which is at a "natural" distance in front of the torso [18]. Additionally, for two-handed signs, the positions of the hands in relation to each other may also be described. There are seven possible locations in the frontal plane: head, mouth, pairy head, trunk, upper arm, lower arm, and lower extremities. Fig. 2a shows examples of symbols for signs that are performed at the center of the upper arm, the shoulder line, the ears, and the head.
- *Orientation*: The orientation of the hand is also defined by the combination of two components: extended finger direction and palm orientation [19]. The former describes the orientation of the knuckles with respect to the wrist, while the latter describes the orientation of the palm [18]. HNS provides symbols for both components in increments of 45 ° [17]. Fig. 2b shows all twelve possible palm orientations.
- *Movements*: Path movements (changes in hand position) can be performed in straight, curved or zigzagged lines, with direction defined in increments of 45 ° [19]. In-place movements (changes in hand posture) can also be performed in sequence or in parallel with path movements. Diacritic symbols describe the size and speed of motion [17]. Examples of the most common movements - straight, curved, wavy/zigzag, and circular - are shown in Fig. 2c.
- *Non-manual features*: HNS provides coding schemes for a number of non-manual tiers including facial expression, head or body movements such as shoulder shrugging, eye gaze, and mouth position [19].

We focus on teaching the first four features in this study.

HNS-based Teaching. With the use of HNS, our approach for implementing the feedback component becomes clearer. First, we need a dictionary of ASL signs translated into HNS. Such dictionaries exist for other sign languages, including German sign language and British sign language, but to date no publicly available HNS dictionary exists for ASL. Given that ASL is the most widely used sign language, developing this dictionary

would have broader implications for research beyond this project. Therefore, both for our own method and for the broader utility of developing this resource, we have spent considerable time and effort translating ASL into HNS from online video dictionaries and other resources. We will discuss how we build this ASL-HNS dictionary in Section 5.2 in more detail.

Once we have ASL signs translated into HNS, the second task is to translate the motion of the user, as recorded by the sensors of the gloves and the headset, into HNS. Based on joint angles, hand orientation, and hand position over time, we then translate recorded motion into HNS primitive features. Completing this task will also have broader implications for the field, as new fluent ASL signers will be able to contribute to the development of the public ASL-HNS dictionary simply by participating in a recording session in which they perform several signs naturally while wearing our devices. Thus, the dictionary can both capture natural variance between signers, and can also grow exponentially with the help of crowd sourcing, rather than hard-coding each individual sign.

Finally, we need to connect the first two steps by comparing the user's motions to the dictionary entry for the sign the user is attempting. By comparing the difference between the user's input and the dictionary entry-both in the space of HNS features-we can then provide clear and interpretable feedback to the user about what parts of the sign are incorrect and how they can correct them. This feedback will eventually be displayed visually through the headset, and possibly also in haptic form through actuators on the gloves (more detail discussed in Section 8). This step would close the loop, allowing the user to attempt to perform a given sign and then to receive feedback in real time.

HNS Benefits. As our overarching goal is scalability of the teaching approach, HNS has several advantages over alternative approaches. Instead of relying on a limited database of hard-coded signs, an HNS dictionary can expand indefinitely and the system will still be able to evaluate the accuracy of new signs because they are always simple linear combinations of existing HNS primitive features. Another advantage–already mentioned but worth repeating–is that the dictionary can be expanded by the contribution of fluent signers, even when they are not familiar with HNS or the linguistic study of sign language. This allows for a natural development of this resource that can inform broader study of ASL, and which can potentially serve the wider sign language community. Moreover, this system can apply to other sign languages besides ASL for which HNS dictionaries already exist. Finally, and perhaps most importantly, relying on HNS to determine the relevant features of sign languages provides a way for us to leverage the amassed knowledge of linguistic researchers who have codified this system of sign language notation specifically to facilitate research by focusing only on the relevant level of detail to accurately describe sign language "phonology" [19]. In other words, rather than relying on our own intuitions about which features of the hand movements are relevant, or a machine learning classifier operating at the level of whole signs–which would be sensitive to overfitting of a training dataset and would be difficult to add additional signs to when we want to expand the dictionary database [2]–we instead rely on this existing codification system designed to precisely translate sign language movements into a written notation [19].

4 SENSING HNS FEATURES

One component of the teaching system is a portable method for sensing HNS primitive features, which includes handshapes, orientation, location, and movements. While the problem of sensing hand configurations is not new, we found that existing approaches fall short in one way or another for this application. For example, one might mount vision-based solutions (e.g., Leap Motion [37] and Kinect [44]) on the VR headset to reconstruct handshapes and track hand motion [53, 67]. However, these techniques are fundamentally vulnerable to occlusion among fingers and between two hands, which frequently occur as users perform ASL signs. Existing wearable systems eliminate the occlusion problem, but are unable to provide information on relative hand positions and overall motion. Recent work has studied the use of ambient wireless signals (e.g., Wi-Fi [35, 43, 52]) to differentiate

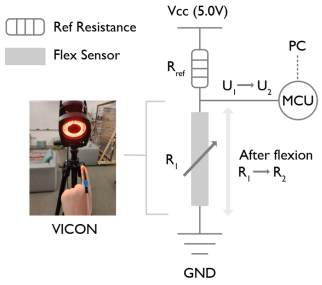
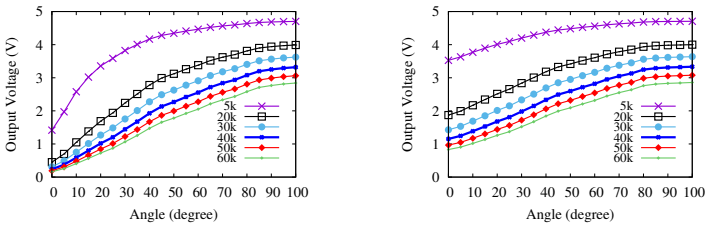


Fig. 3. Measurement platform configuration.



(a) 1-inch flex sensor (b) 2-inch flex sensor

Fig. 4. The change of output voltage as the sensor is being bent with different reference resistors.

signs. These systems, however, only provide recognition information for a limited set of handshapes and typically fail in providing the fine-grained information needed for handshape reconstruction.

To address these challenges, we combine a wearable system with vision-based tracking. The wearable system is realized as a pair of augmented gloves embedded with flexible, lightweight sensors, dedicated to reconstructing fine-grained handshapes. We add visual markers to the glove wrists and exploit the front-facing camera built into VR headsets to track coarse-grained features including orientation, location, and hand movements.

4.1 Sensing Handshapes

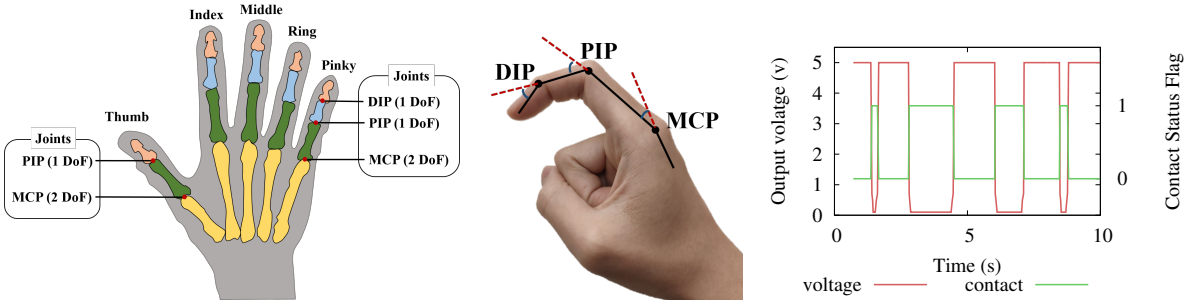
To reconstruct the handshape features, we propose a pair of sensing gloves that track finger joint angles and finger contact information.

Finger Joint Angles. To sense finger joint angles, we leverage flex sensors produced by Flexpoint [26] that detect flexion/extension in one dimension. The sensor is a single-layer, thin, flexible piece of material coated with a proprietary polymer-based ink and can be used as a potentiometer. When the sensor flexes, micro-cracks occur on the polymer coating. These cracks make the conductive particles move away from each other, increasing resistance. The sensors return to their original resistance when released, even after repeated flexing. We use flex sensors with two different lengths, 1-inch and 2-inches. To systematically apply different levels of flexion and obtain the ground truth, we use the Vicon System [57] to measure the real-time flexion angle. Fig. 3 illustrates the overall setup.

Fig. 4 plots the output voltage after a low-pass filter as the 1-inch and 2-inch flex sensors bend along one direction with different reference resistors. We also plot the flex angle measured by the Vicon System to show the levels of flexion. We make two observations here: *First*, as the flexion angle increases, the change of output voltage is monotonic. *Second*, the linearity of the curve is related to the value of the reference resistor. We have two guiding principles to choose our reference resistor value: 1) In order to make the calibration process easier for scalability, we need resistance value that leads closest to a linear curve, and 2) A larger output range will improve the sensing resolution. Once the reference resistor is chosen, the output voltage vs. flexion angle can be approximated as a linear curve. Therefore, we have the following equation:

$$V = k * \theta + b, \tag{1}$$

where V and θ denote the current output voltage and the flexion angle, respectively, and k, b are the calibration parameters that are calculated during the calibration process. We use two data points, one at 0 °and another at 90 °, which correspond to a flat handshape and a fist handshape respectively, in each curve to calculate the two parameters k and b. Then we calculate the Root Mean Squared Error (RMSE) and average error of the linear approximation. We observe that the average reconstruction error with $50k\Omega$ reference resistor for a 1-inch flex sensor is 1.4°. Therefore, it is reasonable to use a linear approximation for the relationship between the bending



(a) Hand anatomy. (b) Three joint angles. (c) Detection of contact.

Fig. 5. (a) Four fingers have 3 joints with 4 DoF while thumb has 2 joints with 3 DoF, resulting into $4 \times 4 + 3 = 19$ DoF in total. (b) DIP and PIP are highly dependent. (c) Compare the output voltage to a threshold to obtain the contact information.

angle and output voltages. We used the same method to select the resistor value for 2-inch sensors, and found that $30k\Omega$ is effective for 2-inch sensors.

To reconstruct a handshape, we need to sense sufficiently many values to completely constrain the joint angles. In order to reduce the DoF we need to sense from 19 to 15, we leverage a kinematic model of the finger. As shown in Fig. 5a, four fingers have two 1-DoF revolute joints (i.e., the proximal interphalangeal joint (PIP) and distal interphalangeal joint (DIP)) and one 2-DoF spherical joint called the metacarpophalangeal joint (MCP). The thumb only contains the PIP and MCP. The movement of these joints is not entirely independent. Specifically, the DIP and PIP joint angles are highly correlated due to the interaction of tendons attached to middle and distal phalanges [6]. Prior work [29] experimentally found that the interdependency between the DIP and PIP joint can be expressed by the equation: $\theta_{DIP} = 0.84 * \theta_{PIP}$. Leveraging this interdependency, we reduce the DoF to 15 while still effectively reconstructing the handshape. Fig. 5b illustrates the three joint angles we defined for the index finger. For simplicity, we refer to the three kinds of joint angles as the PIP, DIP, and MCP. Furthermore, we define the MCP joint constraints of the 2-DoF joints as the MCP joint angle and lateral movement between adjacent fingers. We attach a 1-inch flex sensor to the DIPs and a 2-inch flex sensor to the MCPs on each finger for sensing the 10 joint angles.

Finger Contact. We require lateral contact information to constrain the remaining 5 DoF of the hand. However, HNS symbols capture only contact or no-contact between adjacent fingers, and do not measure precise distances between fingers, suggesting that precise inter-finger distance is not critical in ASL. We therefore require only a robust binary test as to whether adjacent fingers touch each other. Similarly, HNS notation indicates that fingertip-to-fingertip contact between the thumb and other four fingers is important.

To sense contact information, we use a force sensor from Interlink Electronics[28]. The force sensor consists of two membranes separated by a small air gap. When force is applied to the force sensor, the conductive ink shorts the traces together with a resistance that depends on the value of the applied force. For force sensors, the *actuation force or turn-on threshold* is defined as the force required to bring the sensor from an open circuit to below 100 $k\Omega$ resistance. The actuation force of the Model 400 Short Trail Force Sensor is 0.2 N, smaller than the natural contact force between fingers which ranges from 0.26 N to 2.04 N [34]. Therefore, we can use this sensor to obtain the binary contact information between two adjacent fingers and between the thumb and other four fingers. For simple force-to-voltage conversion, the force sensor is cascaded with a reference resistor in a voltage divider circuit and the output voltage on the reference resistor is described by the following equation:

$$V_{FS} = \frac{U_{power} * R_{FS}}{R_{FS} + R_{ref}}, \tag{2}$$

where V_{FS} is the output voltage on the force sensor, U_{power} is the supply voltage, and R_{FS}, R_{ref} are the resistances of the force sensor and reference resistor, respectively. We compare V_{FS} to a threshold, U_{touch}, to determine whether there is contact. Fig. 5c shows an example of using this algorithm to get the contact information, where 1 indicates contact and 0 indicates finger separation. With the joint angles and finger contact information known, we have all the constraints needed to reconstruct the handshape of a 19 DoF hand.

4.2 Sensing Remaining HNS Features

The second sensing component aims to sense the remaining three HNS features (i.e., orientation, location, and movement). Since these three features are all related to hand motion, we need to track the hand in real-time. Adding IMU sensors to the hand would be the most lightweight method and would not severely influence the user's movement; however, orientation and relative position features both require known landmarks. Since our system is portable and users will constantly move during the learning process, multiple IMU sensors would need to be securely fixed to the user. Stability is also difficult to guarantee during motion, which would decrease the sensing performance. Finally, to calculate the hand position from IMU data, we would additionally need to measure the length of each user's arms. This would introduce errors for the overall sensing performance.

Since we already have an MR headset to provide the immersive learning environment, we explore the possibility of leveraging its front-facing camera to sense the remaining three HNS features. After setting the MR headset as a landmark, we 1) augment our sensing gloves with ARTags [12] to obtain the real-time coordinates of the hand relative to the headset and 2) compute the location and movement features. With a specific configuration of ARTags, we are able to retrieve the hand orientation features. We next describe our augmentation of the gloves.

ARTag. ARTag is a fiducial marker system widely used in Augmented Reality (AR) and robotics applications for position and orientation tracking [12]. ARTag markers are bitonal planar patterns with black or white square borders, containing a unique ID encoded with robust digital techniques (e.g., checksums, forward error correction). Using an edge-based approach which links edge pixels into segments and groups them into "quads," quadrilateral contours located in an image or a video can be extracted accurately and quickly. After quad detection, digital processing is applied to identify the tag's ID [12]. With prior knowledge of the actual size and ID of each ARTag marker, we can estimate their positions and orientations in real-time with respect to a camera based on their size and distortion in the image[13].

Computation of Features. The ARTags are arranged on a small cuboid box which also contains the circuit board for the gloves. As shown in Fig. 6, the box has four surfaces (i.e., A, B, C, and D) that may appear in the camera's field of view.

After testing several kinds of configurations, we find that two ARTags on the two rectangular shape surfaces (B and D) and one ARTag on the other two square shape surfaces (A and C) have good robustness and accuracy. This configuration guarantees that regardless of the hand's orientation, there will always be at least one ARTag in the camera's field of view which enables continuous position tracking. In our tracking algorithm, we approximate the hand's real-time position as the position of the ARTags. When we successfully obtain several ARTag positions, we compute the average of these positions as the current hand position. Finally, once the real-time hand coordinates relative to the camera are known, we can easily obtain the relative location feature by calculating the distance (d) between the user's forehead and their hands. We also use the IMU readings from the IMU headset to

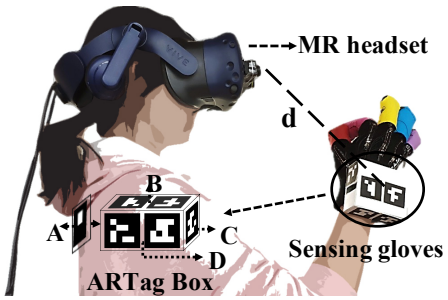


Fig. 6. Illustration of the sensing system. We place ARTags onto 4 surfaces: A, B, C, D, which sense the real-time location and orientation of the hand relative to the MR headset. The distance (d) between user's hands and MR headset is then computed.

cancel out the movement of MR headset. We use the $(time, location)$ pair to record the changing of the hand position, which represents the movement feature. We simplify the orientation feature to which box surface is facing the user. Specifically, we have four surfaces on the box that are possible to appear in the camera's field of view so we have four potential orientations. When we detect multiple ARTags on two different surfaces, we define the orientation as the inter-orientation between them. There are only three possible ways to view two surfaces at the same time (i.e, A and B, B and C, B and D), so in total we have seven orientations. Since we place different ARTags on the four surfaces of the box, we can retrieve the different tag IDs during the detection phase. These IDs indicate which tag is in the field of view of the camera, so we are able to obtain the orientation feature.

Accuracy and Robustness. In order to test the hand sensing accuracy and robustness, we conduct a experiment with one participant. The participant wears the HTC VIVEPro and the sensing glove with Vicon markers attached to both devices. She then moves her hand in arbitrary positions while keeping it in the camera's field of view. We evaluate the ARTag-based sensing accuracy by the deviation of the coordinates. The deviation of the coordinates measures the differences in the hand trajectory tracking that is generated by our ARTag-based sensing and the Vicon system (represented by the palm coordinates in x, y, and z axis.) Fig. 7 shows the deviation of the coordinates in each axis. Overall, the tracking error of x, y, z axis are 2.5cm, 2.9cm, 3.6cm, respectively. This sensing accuracy is sufficient enough to support our applications and we discuss how we leverage the hand tracking in Section 6. During the experiment, we also found that as long as the user put her hand in the camera's field of view, we can always detect the ARTag marker's ID and obtain the orientation information.

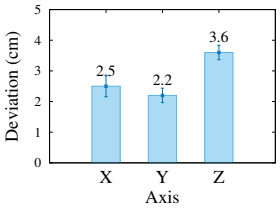


Fig. 7. Coordinates deviation between our ARTag-based sensing and Vicon system.

5 COMPUTING REAL-TIME FEEDBACK

Armed with the sensing data (i.e., finger joint angles and contact, hand location and orientation), we now discuss how we compute and present the real-time feedback. Since the most challenging feature for a user to learn is proper handshape, we focus on providing detailed feedback on the handshape and more general feedback for the other HNS features.

5.1 Sensing Data to HNS

As mentioned in Section 3, with the use of HNS, we can cover all ASL signs with relatively low overhead. To provide feedback on the HNS feature level, we need to first convert the sensing data into the four HNS features. Our methods to calculate the orientation, location, and movement features are discussed in Section 4.2. Next, we will illustrate how to convert the joint angles and finger contact sensing data into the handshape feature.

The handshape feature can be divided into two sub-features: base handshape and modifier. There are two groups of modifiers: five finger modifiers ("extended", "flattened", "bent", "hooked", and "hitch hiker") and three thumb modifiers ("thumb out", "thumb under", and "thumb across"). Since the modifier for each finger can be different, the final handshape is a combination of one base handshape and finger modifiers (up to 5). With 12 base handshapes and 8 modifiers (which are divided into two groups - finger and thumb) in HNS, the number of handshape features is 180 (i.e.,$12 \times 3 \times 5^4$). Clearly, directly mapping sensing data (i.e., joint angles, finger contact) to these possible handshapes entails an expensive training overhead.

To reduce this overhead, we propose a layered approach that sequentially extracts sub-features from sensing data. We observe that because of the effect of modifiers, the base handshape may no longer be easily identifiable from the final handshape. Additionally, extracting modifiers is relatively easy because they have distinctive shapes that can be defined by their angle measurements. We start by extracting the sub-feature of modifier(s) from the sensing data. We then revert the effect of modifier(s) to recover the base handshape by setting the

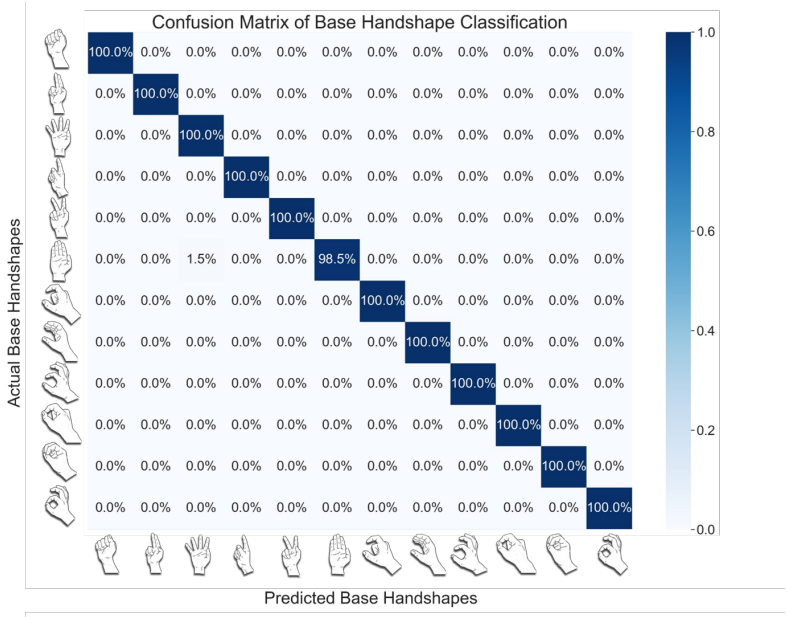
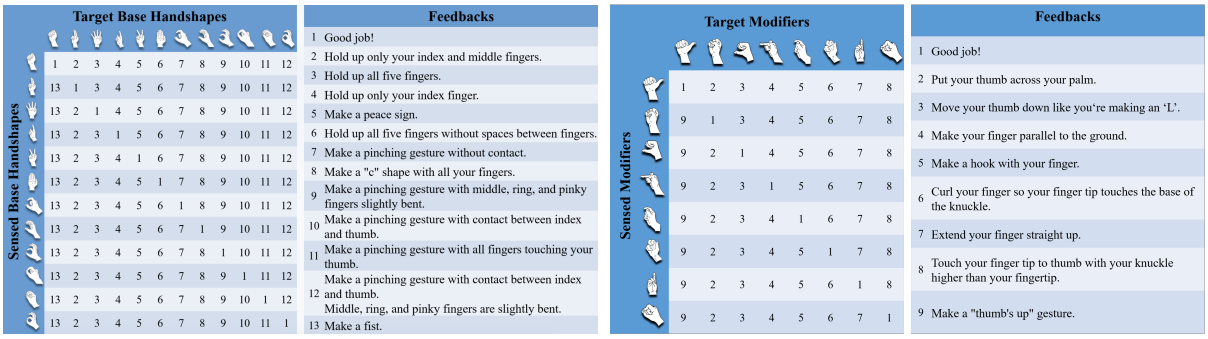


Fig. 8. Classification results of the base handshapes using KNN algorithm.

angles of the fingers with modifier(s) back to a neutral handshape. There are only twelve possibilities for base handshape, which is much easier to classify than the handshape with every possible modifier. This layered approach essentially reduces the search space from 180 to 35 (i.e., $12 + 3 + 5 \times 4$).

Extracting Modifers. To extract the sub-feature of modifiers, we apply a range-based method because all modifiers can be described by the flexion of individual fingers and thus easily identified by comparing the flexion angle along an angular range. Specifically, we consider the MCP and PIP flexion angles to extract finger modifiers. To determine the MCP and PIP angular ranges for each modifier, we recruited ten representative participants with different palm sizes to perform the eight modifiers while wearing our sensing glove. The MCP and PIP ranges are then set for each modifier by the maximum and minimum MCP and PIP angles measured by the glove across all participants performing this modifier. Because we already know the sign the user is performing, we can narrow down the modifiers to examine. For the signs without any modifiers, we can bypass the modifier extraction process. Since the two ranges used to extract the modifiers do not overlap in each group, the overall accuracy with this range-based method is nearly 100%. Once modifiers are detected, we then replace the modifier angles or contact with the values of the neutral handshape. The neutral handshape is chosen based on the definitions of the base handshape in HNS documentation [17].

Obtaining Base Handshapes. We use the k-nearest neighbors (kNN) algorithm [2] to classify the data into 12 base handshape classes. We have ten participants learn the twelve base handshapes for several hours with the guidance of an HNS expert. We then let them perform the twelve base handshapes while wearing our sensing gloves. Each user performs each base handshape ten times (5 seconds each time), during which we collect the joint angles and finger contact data. The total number of data points per user is $12 \times 10 \times 5 \times 40$ with a 40-Hz output frame rate of the sensing gloves. We then manually label them from 1-12 to indicate the twelve base handshapes. We use this dataset as the training and testing data for our kNN classifier. The kNN classifier was built using the Python Scikit-learn package [47]. Since the training phase of the kNN algorithm consists only of storing the feature vectors and class labels of the training samples, the overall training overhead is low. We perform



(a) (b)

Fig. 9. Transition instructions between (a) sensed and target base handshapes (b) sensed and target modifiers.

leave-one-out cross-validation: choose nine of the ten users' data as the training set and the last one as the testing set and repeat the process 10 times. Then we use the overall accuracy and overall F-score as the performance metrics. For binary classification, the F-score is calculated as $2 \times (precision \times recall)/(precision + recall)$. In our study, we calculate the overall F-score by counting the total true positives, false negatives, and false positives for all the classes. The overall accuracy value is 0.9989 and the F-score is 0.9988. Fig. 8 shows the performance of each base handshape and the entries have been normalized for each class. We can see that we can accurately classify the base handshapes. Notably, "Finger2345" was not 100% predicted and misclassified as "Finger2345Spread." This is most likely due to the user not pressing their fingers together while performing the "Finger2345" handshape.

5.2 Real-time Feedback

ASL-HNS Dictionary. We provide feedback based on the discrepancy between the sensed handshape and the target handshape. To determine the target handshape feature, we build an HNS dictionary describing the HNS features for ASL signs. First, we recruited three people who already have some knowledge of ASL. They spent five months learning HNS and translating the 56 ASL signs documented in the ASL Signbank [20] into their corresponding HNS features. Next, we convert the translated HNS features into animation using the Signing Gesture Markup Language commands [10] and JASigning software [61] to make sure we the HNS features match the target sign. After this verification step, we add the correct HNS features into our dictionary. Each entry of the dictionary contains the base handshape, finger modifiers, palm orientation, location, and any movement associated with the sign. We can then find the target base handshape and modifiers from our HNS dictionary.

Feedback For Handshape. A straightforward way to give feedback for handshape is to calculate the difference between the target finger angles and the sensed finger angles. We can then use text to provide quantitative feedback such as "bend/unbend the index finger 30 °." We conduct a pilot study with this method and find this kind of feedback is too overwhelming for users to effectively understand.

We found that descriptions such as "make a thumbs-up gesture" or "make a pinching gesture" were more effective than providing quantitative feedback for joint angles. We choose this method because we want to emulate the way a coach for a sport might teach a complex motion by breaking it into smaller motions the athlete already knows how to perform. Building on this idea, instead of telling the users to bend/unbend their fingers to a certain angle, we pre-define the transition instructions between the sensed modifier/base handshape and the target modifier/base handshape.

We empirically determined these transition instructions by first asking our HNS experts to formulate several descriptions for each modifier/base handshape pair. Then, they show users different descriptions of each

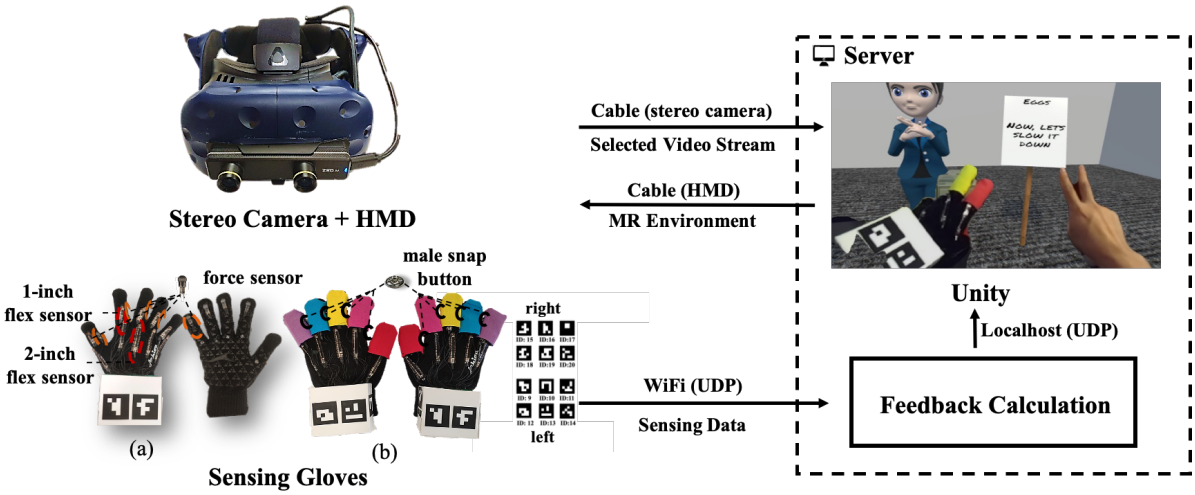


Fig. 10. System Architecture. Sensing gloves: (a) Two types of sensors on the gloves. There are five 1-inch flex sensors and five 2-inch flex sensors. We also attach 8 force sensors to the fingertips and lateral finger parts. We have 18 channels of data on one glove. (b) Final look of our gloves. Each finger is covered with different color finger cots for better cropping performance. Each glove is augmented by 6 ARTag markers.

modifier/base handshape pair and select the one that user felt provided the clearest explanation. The chart used for base handshape to select these transition instructions is shown in Fig. 9a. If, for example, the user performs "HamPinch12" but should have performed "HamPinchAll," our feedback would be "Make a pinching gesture with all fingers touching your thumb." The same feedback chart for modifiers is shown in Fig. 9b. We will discuss the forms we use to present this textual feedback in the following implementation section.

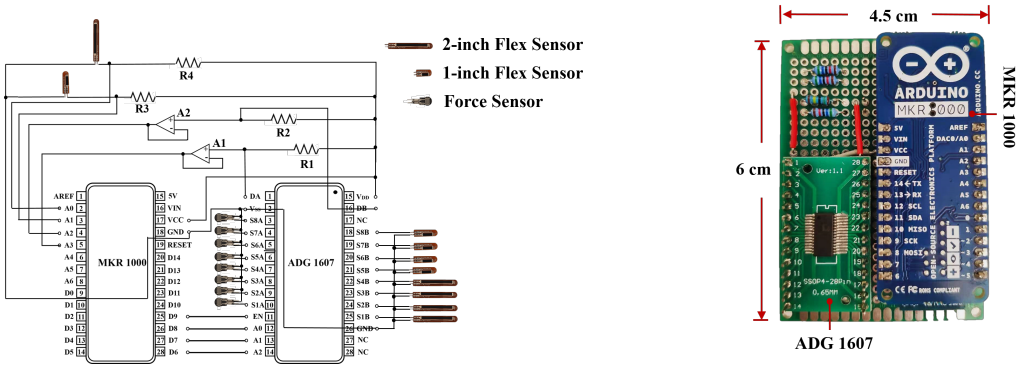
Feedback For Remaining Features. For the location and orientation features, since they are static and relatively simple to learn when the user can see the correct first-person hands ahead of them in the MR environment, we only highlight the correct locations in the MR learning environment according to our HNS dictionary. This helps guide the users to where they should move their hands while performing each sign. Using the real-time location and orientation of the user's hand, we can provide feedback about whether or not the user's hands are at the target location and orientation. Currently, we assume the user can correctly follow the expert's hand movements and leave the movement feedback for future work. We will discuss how we design an MR learning lesson for users to learn and follow with sufficient details in the following implementation section.

6 PROTOTYPE IMPLEMENTATION

The hardware of our ASL teaching system consists of a VR headset, a 3D stereo camera providing the immersive mixed-reality learning environment, a pair of ARTag-augmented sensing gloves to sense the HNS features, and a computer to power the experience (see Fig. 10). We combined this hardware into an integrated learning experience using Unity3D game development tools [58]. We created lessons within the virtual environment featuring a teaching avatar, first-person and third-person demonstrations, and an informational whiteboard. These lessons were designed to provide both implicit and explicit feedback over several repetitions at slow speeds.

6.1 Hardware Implementation

MR Setup. We realized an MR environment by combining a VR headset (HTC VIVEPro) and a 3D stereo camera (ZED Mini) attached to the front of the headset as shown in Fig. 10. The ZED Mini supports a wide field of view,



(a) Sensing glove circuit design. (b) Fabricated circuit board.

Fig. 11. The overall circuit design and real circuit board. 10 flex sensors and 8 force sensors are connected to the Micro-controller through the multiplexer.

at 90°(H) × 60°(V) × 110°(D), which is important for a high-quality MR experience [54]. Only the portions of the video stream that are within arm's reach (primarily the user's hands) are displayed - the rest of the user's visual field is a pure VR environment (our method for doing this is described in section 6.2).

Sensing Gloves. To fabricate our custom-built gloves, we augment a pair of off-the-shelf gloves made of high-quality acrylic fibers [24] because of their better support for embedded sensors while guaranteeing the flexibility for finger movement. As shown in Fig. 10, we sew five 1-inch flex sensors and five 2-inch flex sensors [26] to the PIP and MCP joints, respectively. We also place eight force sensors [28] on the fingertips and lateral finger components to obtain contact information.

The flex and force sensors are connected to a micro-controller (MKR1000 board [25]), which digitizes signals from these sensors. Fig. 11 shows the circuit design and the circuit board in our prototype.

We attach an ARTag box ($6.5cm \times 7cm \times 3cm$) on the wrist of each glove, where the ARTag box is fabricated as a cardboard box[27] with ARTags printed on each surface. We use the ArUco Library [16] to generate 12 ARTag markers with IDs ranging from 9 to 20, allowing us to detect and identify the different markers. These markers are printed on $3cm \times 3cm$ pieces of white paper. Since our gloves are black, we fabricate ten finger wraps with different colors using sport kinesiology tape [50] for better cropping performance. We also attach a male snap button to the lateral side of each finger wrap to make the force sensors more sensitive to side contact. Fig. 10 shows the final implementation of our sensing gloves.

6.2 Software Design

MR Environment. We set up a virtual classroom where users can see their hands utilizing selective video passthrough from our 3D stereo camera. This was achieved by modifying a forward-lighting shader to take into account the computed depth buffer information and only render video passthrough for pixels that were determined to be within an arm's length of the HMD. To decrease these cropping artifacts, we choose fabrics for the sensing glove with high surface contrast and ran the experiment in a well-lit room.

The lesson was implemented in a single scene within Unity. The most important objects within the scene include: 1) a third-person teacher avatar (Fig. 13c), positioned 1.5 meters in front of the user. 2) a first-person avatar (Fig. 13d), a copy of the teacher avatar positioned just in front of the user, but altered to have a translucent suit and to only render the arms; 3) a whiteboard for real-time feedback (Fig. 13b), implemented as a rectangle

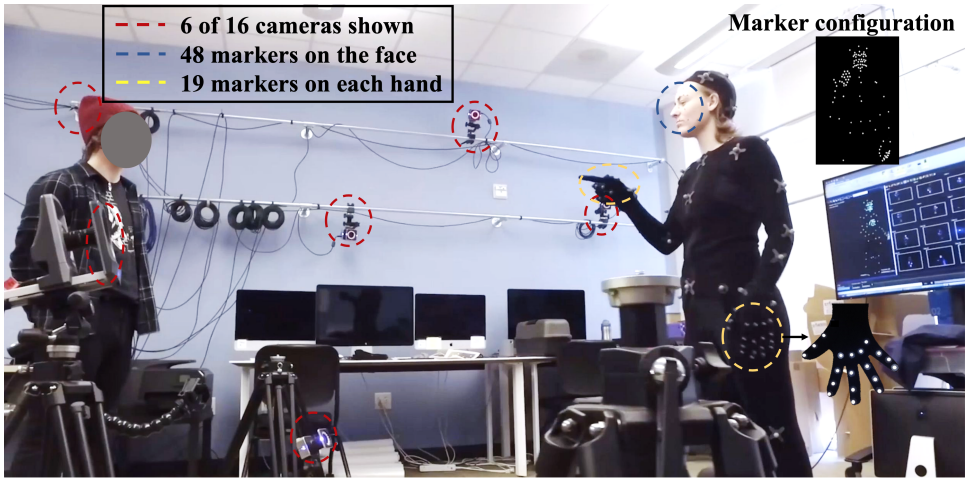


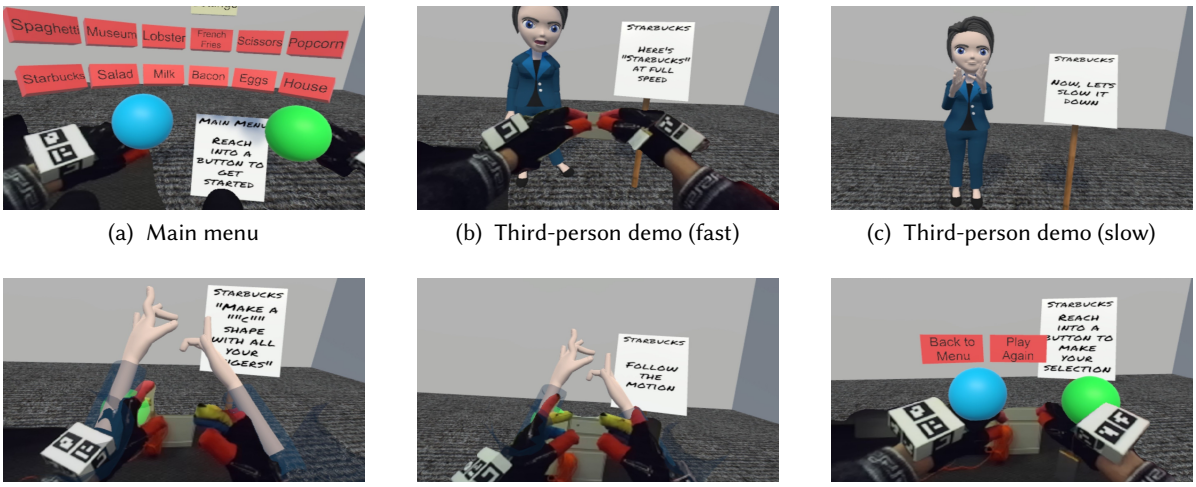
Fig. 12. A 16-camera motion capture Vicon system. 120 markers are placed on a native deaf signer: 19 markers on each hand, 48 markers on the face, and 34 markers on the upper body.

and cylinder with text meshes on top; and 4) menu buttons (Fig. 13a). Lesson selection buttons are generated programmatically as the environment is initialized.

Digital Modeling. The third-person teacher avatar was created by recording motion capture data and mapping it onto a 3D humanoid model. The avatar was modeled in Maya [5] and then imported into Motionbuilder [4] for the animation. This was done by creating a skeletal mapping for the model and matching the skeleton of the motion capture data to the corresponding points on the model. As shown in Fig. 12, a 16-camera Vicon system with 8 MX Series [60] and 8 Vero Series cameras [59] was used for motion capture recording. Markers were placed on 120 locations on the signer's body, with labeling done in Vicon Blade [45]. One deaf signer acted as the model for the motion capture recording session. She signed short scripts of introductory ASL content, in the role of an ASL teacher. The 3D virtual human design went through several iterations to ensure she was aesthetically appealing and likely to be accepted by users. The virtual human's geometry was rigged to the motion capture skeleton, to ensure the avatar would accurately represent the recorded movements.

Learning Lesson Design. Our teaching environment is designed for a seated learner and consists of a virtual teacher avatar, a whiteboard, and disembodied first-person arms. Learners can interact with our MR environment by reaching into buttons to select (hand position is detected using our ARTag-based tracking, as described in Section 4). The lesson starts when the learner selects a sign they want to learn (Fig. 13a):

- The first step is a teacher demonstration, where the teacher avatar performs the sign three times at full speed(Fig. 13b) and then another three times at 30% speed (Fig. 13c). No explicit instruction or feedback is given to the learner during either of these steps, although the learner's ability to see their own hands and compare them to the avatar's can be considered a form of implicit feedback.
- Then the lesson transitions to a first-person instructional mode, where the teacher avatar is hidden and a cropped first-person avatar is presented(Fig. 13d). For single-handed signs, only one arm is shown. These arms are oriented to match the learner's orientation - therefore the learner does not need to perform a mental rotation to understand how the avatar's arms and hands map to their own. The avatar's arms are translucent, but the hands are opaque. This allows the learner to see how the first person avatar's shoulder, elbow, and wrist are configured without losing sight of the hands. Below and in front of the avatar's hands are position targets(as discussed in Section 5.2), visualized as translucent green and blue spheres.



(a) Main menu (b) Third-person demo (fast) (c) Third-person demo (slow)

(d) First-person demo (static) (e) First-person demo (motion) (f) Completion screen

Fig. 13. Lesson Sequences. Users interact with MR environment and navigate themselves by reaching their hands into specific buttons and highlighted areas.

At this point, the system waits for the learner to achieve the correct static handshape before continuing. Feedback is delivered as text on the whiteboard (as discussed in Section 5.2). The participant is able to see the target handshape, attempt to replicate it, and receive corrective feedback simultaneously. There is no time limit to this lesson step, however, there is a "skip" button that learners can use if they become frustrated while attempting to achieve the target handshape.

- Once the learner achieves the correct handshape or selects the skip button, the lesson continues and plays the sign animation on the first person hands at 30% speed (Fig. 13e). The motion is repeated three times, and the learner is asked to follow the motion. After the learner performs the sign, the lesson automatically advances to a completion screen (Fig. 13f) which congratulates the learner and offers a choice to either repeat this sign or return to the main menu.

7 EVALUATION

Given that the sensing component of our system has been evaluated in Section 4, in this section we focus on the system's end performance for teaching. We conducted a user study with four user groups, where one of the groups used our MR system and the other three used alternative systems with a desktop screen. We chose a set of 12 signs as the first ASL learning targets: spaghetti, museum, lobster, french fries, scissors, popcorn, Starbucks, salad, milk, bacon, eggs, and house. We set the number of signs as 12 in order to match the average number of signs taught in an in-person 30-min ASL class [65]. These signs can easily be performed while seated and their movements occur in the physical neutral space in front of the participant, allowing them to observe both their own hands and those of a virtual avatar within the same field of view. We compared the performance of the three groups based on the 3 HNS features (the location feature was neutral for these 12 signs). We also collected participants' comments through a questionnaire and an interview to evaluate the effectiveness of our system.

7.1 Participants and Procedure

We recruited 60 participants with various backgrounds from our local institution of ages 18 – 33 (average = 24.2) years. All participants are right-handed. 55 participants have no prior ASL experience. 5 participants have seen

the ASL alphabet before but reported that they did not remember it. We randomly divided them into four groups which include an *MR group* using our MR system, an *interactive desktop group* using the desktop version of our system (without MR immersiveness), an *non-interactive desktop group* using the non-interactive desktop version of our system (without MR immersiveness and feedback), and a *Signbank video group* watching modified online videos from Signbank [20]. We set up and calibrated the various systems as follows for each user group:

1) MR Group. For participants in this group, they used our MR system. We first instructed them to put on our sensing gloves and then calibrated the gloves. The calibration was done by asking the wearer to perform an open hand and a closed fist, which takes roughly one minute. Then, as shown in Fig. 14, participants were instructed to wear the HTC VIVEPro Headset and experience the learning lesson in the MR environment.

Fig. 14. Setup of MR group.

2) Interactive Desktop (ID) Group. To isolate and test the impact of MR immersion on sign learning, we created a variant of our system that displays on a standard 2D monitor instead of the HMD of the MR system. The user still wears the sensing gloves and receives feedback in the ID system (Fig. 15)–other than using a mouse to select on screen buttons, the rest of the experience is identical.

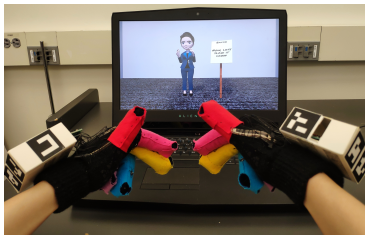


Fig. 15. Setup of ID group.

3) Non-interactive Desktop (NID) Group. This baseline group is to verify the effectiveness of the real-time feedback. The NID group use the same desktop verison system but without real-time feedback and they do not wear the sensing gloves. Participants were seated a comfortable distance away from the monitor, and instructed to use the mouse to navigate the lesson.No other instruction or feedback was provided at any point, and participants were not required to perform a sign correctly before moving on to the next lesson.

4) Signbank Video (SV) Group. Since the easiest and most common way to learn ASL without an instructor is through watching online videos, we modified the videos that were freely available on the Internet and constructed a Signbank video lesson. Videos of fluent signers performing each of the same 12 signs used in the MR lesson were downloaded from the Signbank website [20] and formatted using Microsoft Powerpoint. Participants were seated a comfortable distance away from the monitor, and instructed to press a keyboard key to navigate the lesson. Each video was played at full speed for three repetitions, then at 30% speed for another three repetitions. The participant was not instructed to follow along during this phase. Finally, the video was played at 30% speed for another three repetitions. During this section, the text was displayed on the screen encouraging the participant to try to copy the signer in the video. No other instruction or feedback was provided at any point, and participants were not required to perform the sign correctly before moving on to the next lesson.

7.2 Results based on HNS Experts

The same HNS experts who helped build our dictionary have designed detailed evaluation metrics for all 12 signs. To avoid bias, we hired another three HNS experts who are unaware of the goal and procedure of our experiments to evaluate the performance of all four groups. Each sign was evaluated based on 3 HNS features: handshape, orientation, and movement. Each feature is rated on a scale of 1 – 10.

1) Rubric for Handshape. The rubric for handshape addresses whether the base handshape is correct, whether the modifier is correct, and whether both hands are in the proper handshape. The signs in the system were either one-handed signs or symmetric two handed-signs. Points were deducted from the handshape if the participant used the opposite hand of the one they were viewing for one-handed signs and points were deducted if the handshapes were not symmetric for two-handed signs. The handshape was weighted the heaviest, as it is the most important part of the sign and the other pieces were given point values accordingly (e.g., for the sign ‘Lobster’ , a

participant can earn 4 points for "having the open 'c' shape" and 3 points for "no spaces between fingers" and 2 points for "thumb opposite" and 1 point for "both hands in shape", respectively).

2) Rubric for Orientation. The rubric for orientation considers palm, finger, and thumb direction. Additionally, this metric includes whether the sign is performed in the right location relative to the body. Points are deducted if any of the directions are incorrect, with the palm direction weighted the heaviest (e.g., for the sign 'Lobster' , a participant can earn 4 points for "fingers pointing out", 3 points for "palms facing each other", 2 points for "thumb pointing out", 1 point for "hands are at the center of body", respectively).

3) Rubric for Movement. The movement rubric considers the overall motion of the sign and whether the handshapes remain either constant during the sign or change depending on the sign. The starting and ending locations are also considered. The high-level motion is weighted the most and the relative handshape was assigned a point value accordingly (e.g., for the sign 'Lobster' , a participant can earn 4 points for "fingers bend in" and 3 points for "fingers touch thumb" and 3 points for "repeat motion" , respectively).

Overall Performance. We first averaged scores given by the three evaluators and used the mean value to represent each participant's performance for each of the three features: handshape, orientation, and movement.

- *Performance analysis between groups.* A mixed-design ANOVA reveals a significant main effect of learning group on performance ($F_{3,56}$=29.05, p<0.001; η^2=0.609), indicating that the four groups did not learn the signs equally well. Mauchly's Test of Sphericity indicated that the assumption of sphericity had not been violated (χ^2=5.339, p=0.069) , indicating that the four groups did not have significantly unequal variances. A post-hoc Tukey test reveals that the MR group significantly outperformed the other three groups across all three features (p<0.001). No other groups were found to differ significantly in their overall scores.

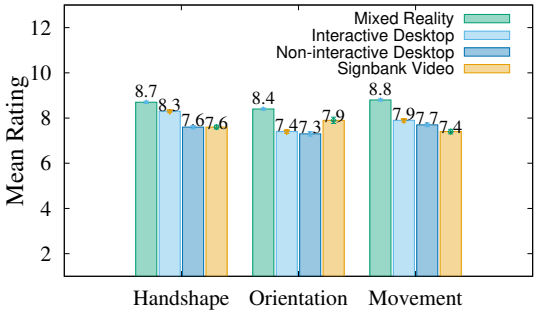


Fig. 16. The averaged evaluators' rating of all participants for the three features.

- *Performance analysis within groups.* There was no single feature that exhibited a significant effect on performance, which shows that all three features contribute equally to the overall performance. There was a significant interaction between group and features ($F_{6,112}$=29.05, p<0.001; η^2=0.359), indicating that not all groups performed proportionately well on each feature. To investigate which differences between groups are driving this interaction, a second repeated-measures ANOVA was performed. The effect of feature was significant in the MR group ($F_{2,118}$=53.52, p<0.001; η^2=0.640) and the ID group ($F_{2,118}$=13.82, p<0.001; η^2=0.773), but not the NID group or the SV group. This indicates that MR and ID groups did not perform equally well on each feature. In both cases they performed slightly better on handshape and movement.

We conducted several post-hoc t-tests to verify the effectiveness of each design element (i.e., immersive environment, real-time feedback, and first-person demonstration).

Immersive Environment. We compare the performance bewteen the MR group and the ID group to verify the contribution of the immersive environment. The MR group scored significantly higher than the ID group on handshape (t(28)=6.05, p<0.001; d=2.25), orientation (t(28)=6.43, p<0.001; d=2.35), and movement (t(28)=6.07, p<0.001; d=2.21) (See Fig. 16). With the immersive environment, the MR group can adjust and find a most suitable view of the hands for themselves, making full use of the first-person demonstration. The ID group only has a font facing view, where sometimes fingers block each other. The ID group also had difficulty figuring out the correct start and end position of the sign. Hand position tracking and the color highlighted hand locations as

shown in Fig. 13f in the immersive environment substantially helped the MR group to make sure their hands are in the right positions.

Real-time Feedback. The ID group outperformed the NID group in handshape ($t(28)$=4.91, p<0.001; d=1.79), but no significant difference in orientation or movement were found. Since the only different between ID and NID group is the real-time feedback, this result indicates that the detailed handshape feedback is very helpful. As shown in Fig. 16, the general feedback helps a little bit on learning the orentation and movement but not significantly.

First-person Demonstration. Performance bewteen the NID group and the SV group tests the contribution of the first-person demonstration. Although most participants agree that the first-person demonstration is helpful, the SV group scored significantly higher than the NID group on orientation (t(28)=3.52, p<0.001; d=1.29), with no major differences in handshape or movement (See Fig. 16). It is more difficult to copy a motion of an avatar than copying a real person because the avatar hands are not rigged exactly the same as a human hand. This shows that without feedback and MR immersiveness, the first-person demonstration is not as helpful as learning from a human teacher.

Most Common Apporach. We compare the performance bewteen the MR group and the SV group to show the effectiveness of the proposed MR system over the most common used approach to learn ASL without an instructor–online videos. Combined with all the advantages that our design elements bring, the proposed MR system scored significantly higher than the SV group on all three features: handshape ($t(28)$=9.86, p<0.001; d=3.53), orientation ($t(28)$=4.43, p<0.001; d=1.61), and movement ($t(19.02)$=6.98, p<0.001; d=2.55).

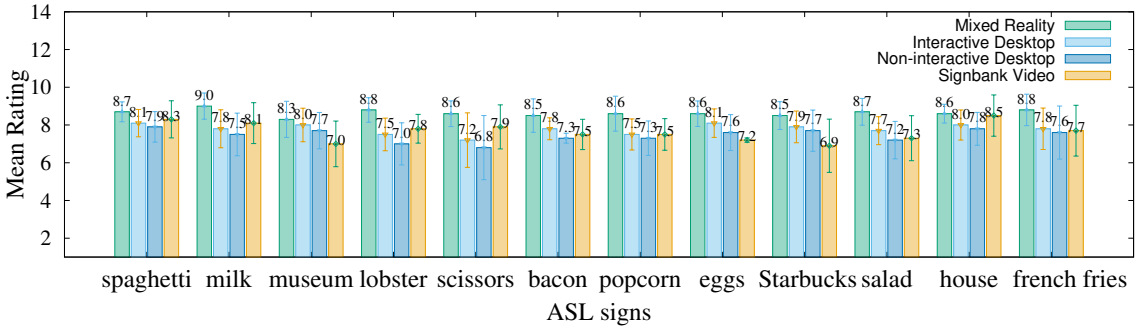


Fig. 17. The mean rating of all participants for 12 ASL signs in all three groups.

Performance across ASL Signs. To examine the performance across different signs, we took the mean value of the three features among 15 participants in each group as the sign performance. Overall, the MR group achieved the highest rating in all 12 signs. Although the SV group performed well on signs with easy shapes and motions, such as the signs for "spaghetti" and "house" (Fig. 17), they had a difficult time with signs that included more complicated motion trajectories, such as the signs for "bacon", "salad", and "popcorn". With the three desktop-version systems, participants had a more difficult time with signs that included arm movements, such as "bacon", "milk", "salad", "french fries", and "museum" as shown in Fig. 17. Evaluators reported that participants using the desktop-version systems tended to exaggerate the movements or perform them in an incorrect plane or with the incorrect hand. Evaluators also noted that signs such as "Starbucks" which have more difficult handshapes gave the NID and SV group a lot of difficulties. These results demonstrate the potential of using our system to advance the teaching of more complex ASL content such as grammar and sentences. This is also the potential reason that the improvement in the performance of the MR group is rather small as signs we select are relatively easy.

Interrater Reliability. Interrater reliability is a statistical measure of agreement between raters. It ranges from -1 to 1, where 1 is perfect agreement and 0 means ratings are unrelated while -1 is perfect disagreement. In our study, we evaluated the reliability of the three evaluators using two different metrics.

First, we calculated Krippendorff's alpha coefficient, a statistical measure of agreement between two or more raters evaluating the same content [33]. Krippendorff's alpha was calculated for z-scored ratings of each sign for each participant, averaged across all three features (handshape, orientation, and movement), using the online utility ReCal [14] with a confidence interval of 95%. Krippendorff's alpha in the present study was 0.59.

Next, we calculated intraclass correlation coefficient (ICC), a modification of Pearson correlation coefficient which reflects both correlation and agreement[42]. ICC form (3,3) was calculated for the same z-scored ratings averaged across the feature dimensions (handshape, orientation, and movement) using SPSS version 26.0. ICC in the present study was 0.57, which falls into the range of "moderate" reliability [32].

Given that individual idiosyncrasy in sign performance, which is common even among fluent signers (comparable to different accents in the same spoken language), likely contributed to the noise in the ratings, these results indicate acceptable levels of interrater reliability.

7.3 Results of Participant Self-reports

Since "learning effectiveness" is a subjective concept that varies from person to person, we also asked the participants to fill out questionnaire. Additionally, we conducted 1-minute interviews to gather input. The questionnaire asks participants of all four groups to rate on the standard 5 point Likert scale [36] : (1) Strongly Disagree, (2) Disagree, (3) Neutral, (4) Agree, and (5) Strongly Agree to the following statements:

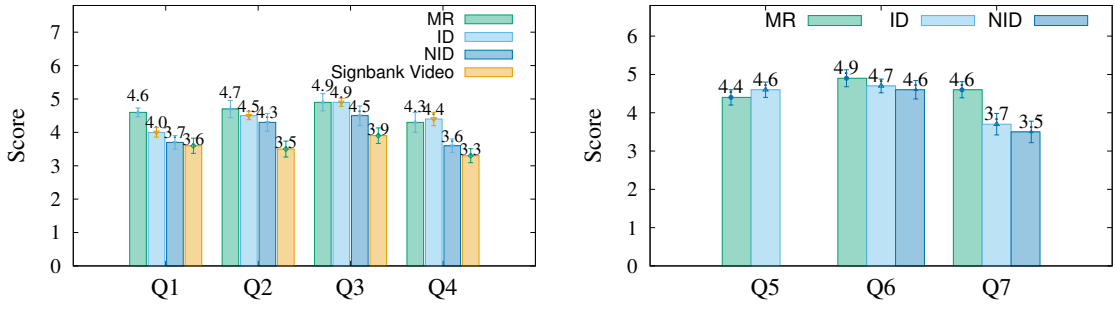
- Q1: I found this lesson to be engaging.
- Q2: I think I would learn a lot from using this lesson to practice ASL.
- Q3: This lesson would be helpful for people who are just beginning to learn ASL.
- Q4: This lesson would be helpful for people who already know some ASL.

For the MR, ID, and NID group, we asked them to answer rate two additional statements:

- Q6: I found the first-person hand demonstration very helpful.
- Q7: The immersive Mixed Reality environment/animated demonstration was very beneficial.

For the MR and ID group, we asked them to rate one additional statement:

- Q5: I found the feedback given during the lesson very helpful.



(a) Average score for question 1-4.

(b) Average score for question 5-7.

Fig. 18. Overall Likert scale responses for all four learning systems.

Engagement. As shown in Fig. 18a, the questionnaire feedback indicates that participants were very enthusiastic about our MR teaching lesson, rating their engagement (Q1) an average of 4.60 out of 5. The evaluators observed that participants in the three desktop systems were attempted to repeat all previously learned signs, but later abadoned trying to trying to perform them all.

Usefulness. Questions 2-4 focus on the usefulness of each system. The MR and ID system outperformed the other two systems, indicating that immersive environment and real-time feedback did improve their overall learning. Specifically, our MR learning lesson achieved 4.93 out of 5 for question 3 (Fig. 18a), which means 14 out of 15 participants strongly agree that our MR learning lesson would be helpful for people who are just beginning to learn ASL. Since our original goal was to create a system for helping people with no experience learn ASL, this was an encouraging result.

We asked the MR, ID, and NID groups two additional questions (Q6 and Q7) in order to evaluate the effectiveness of first-person hand demonstration and immersive MR environment in our learning lesson. As shown in Fig. 18b, participants in all three groups reported that first-person hand demonstration is very helpful. During our study, one thing that struck our evaluators was that in the SV groups, a lot of participants performed one-handed signs with their left hand, even though all participants were right-handed. Generally, when signers perform a one-handed sign they use their dominant hand. The participants may have switched hands because they were subconsciously "mirroring" the movements they saw in the online videos, which do not have the first-person view. This observation helps to validate the effectiveness of the first-person hand demonstration.

The immersive MR environment achieved an average score of 4.6 on usefulness, while the animated demonstration only achieved 3.7 and 3.5 in ID and NID group. This shows that the immersive MR environment did improve participants' learning experience. Participants in MR and ID groups also gave an average score of 4.4 and 4.6 out of 5 for the real-time feedback, indicating its usefulness.

User Suggestions. User feedback also points to room for improvement.

- There were 7 out of 15 participants who mentioned that the avatar's fingers sometimes appear in odd angles and even intersect the palm of the other hand, which is creepy and impossible to follow. Some participants skipped signs that had those problems. This is due to the imperfect match between the avatar model and the motion capture data. Even though we used 16 cameras to capture both macro body movement and micro finger movement, it was challenging to have perfect mocap data for every single frame. We leave the fine-tuning of each frame of the animation as future work.
- Two participants encountered a problem that the system failed to recognize their correct signing and kept giving the same feedback, which left them stuck at the handshape stage for a long time. This problem frustrated some participants and caused them to skip more frequently than other participants as they grew tired of attempting to acheive the target handshape. This indicates that a smooth learning experience guaranteed by the robust sensing components is crucial for a participant's engagement and effectiveness of learning.
- Some participants mentioned they would like to take a test later on to validate their learning. Several participants mentioned they would like to have a 'talking' stage, where they can apply what they learned to communicate. We leave this as one of the limitations and future work. We discuss it in detail in Section 8.

Overall Evaluation. We have an open question at the end of the questionnaire asking for the participant's overall evaluation of the assigned systems and suggestions on how to improve the systems.

MR Group. 80% of participants in the MR group used the world "helpful", which highlighted the usefulness of our system. Additionally, 70% of participants used words like "amazing", "interesting", and "fun" to describe the experience. This is an encouraging and somewhat expected result, as the immerserive learning environment provides an effective and innovative way to improve engagement during the learning process.

NID and SV Groups. More than 80% of participants in both video groups commented that they have no sense of whether they performed a sign correctly, and that they think feedback on the correctness of a sign or how to correct a mistake would be very helpful. This is an expected problem of motion learning with no instructors, which validated the necessity of the real-time feedback in our MR system.

To authentically capture the 'voice' of the participants' remarks, we also report some comments verbatim (including any errors):

1) ***MR Group:*** *"I thought this tool was* ***amazing****. The starting requirement where you have to place the blue boxes together was finicky, and I would have liked it to be less strict about the exact hand gesture to start the motion. Overall, I would not majorly change anything."(****P03****)*

2) ***NID Group:*** *"Feedback on the white board is very helpful for correcting my handshape. I hope there could be some feedback for the trajectory."(****P54****)*

3) ***NID Group:*** *"The animated woman was kind of scary. Also multiple perspectives of the teacher hands are helpful.* ***Feedback*** *would also be helpful."(****P37****)*

4) ***SV Group:*** *"It was kind of helpful for me to learn signs but I have no idea if I am right. "(****P27****)*

8 LIMITATIONS AND FUTURE WORK

User Studies. We recognize that our current user study is limited by the small user population and short-term learning experience with a limited number of ASL signs. We will expand our user base to a group with more diverse age and background. Furthermore, since ASL is a language people use to communicate, we will add more evaluation schemes related to communication, such as peer-evaluation between participants to validate the effectiveness of our system. We can ask participants if they can recognize others' gestures as ASL signs. One of the limitations of the user study is the lack of a long-term retention test. We will conduct longer-term studies to determine whether our system helps improve learning durability (i.e., whether the user would remember the ASL sign longer when compared to traditional learning methods).

Baseline Comparison. Instructor-based learning, given that learning with an instructor, albeit unscalable, is still the most effective way of learning ASL. We did not include such comparison because of the limited content of our current lesson. We plan to add the comparison after we design an MR lesson that is more comparable to offline learning content.

Sign Animation. To animate the teaching avatar and provide third-person sign demonstration, we have relied on collecting motion capture data from native signers and manually mapping the data onto our teacher avatar. In the future, it is possible to develop an application similar to the web avatar created at the University of East Anglia [61] so we can automatically generate the animation for a given sign. This application would take the HNS from the dictionary and translate it into an animation of the sign on the avatar, so we do not have to map a human skeleton onto a digital avatar. Another possibility is to generate the 3D skeleton data from existing sign language videos, which is similar to the method proposed by Heike et al. in [7]. These approaches would scale the number of signs we can teach tenfold.

Additional Learning Content. The lesson design we implemented is focused on the introduction of new sign vocabulary, but this is only one step towards the learning of ASL. With additional content, our learning environment could foster a more complete acquisition of this new language. Some potential next steps could be: 1) *Non-manual Features:* Develop new content and system capabilities for teaching the coordinated use of non-manual features, which are essential for ASL grammar. 2) *Standard Curriculum:* Adapt publically-available ASL curriculum [1] for the virtual reality environment. 3) *Simulation Games:* Place the user in a realistic environment and ask them to perform certain tasks using sign language skills they've been learning. 4) *Daily Lessons:* Keep

track of the user's progress and generate lessons based on a user's learning history, which would encourage users to practice sign language as a daily habit.

Tactile Feedback. Our current feedback is mainly visual through the MR environment. Researchers have proved that tactile feedback benefits navigational and positioning tasks [63]. Given that our system is capable of sensing movement features, we can use tactile feedback to provide precise, instant instruction to correct a user's movements. We can use different levels of tactile feedback to correct a user depending on how much their performed hand trajectory deviates from the target trajectory. Since we have a pair of sensing gloves, it is possible to add tactile feedback to a user's hands. To guarantee the freedom of finger movement, we need to develop a method that will sense and give tactile feedback without many additional hardware components. Materials such as flexible electrostatic transducers (FET) [56] have the potential to both sense and give tactile feedback and are worth exploring.

Extension to Other Motion Tasks. Problems addressed in this paper – such as the sensing of human body configurations, delivering physical instruction, evaluation of positional correctness, and computation of relevant feedback — are applicable to many forms of motion tasks. A mixed reality motion teaching system could be used for swimming, musical instruction (perhaps augmenting the tactile system described by Xia et al. [66]), physical therapy, and more. In general terms, all motion teaching tasks are united by the need for an abstract definition of correctness that is robust against differences in body size and shape. Future work could attempt to define a generalized human motion encoding system akin to a whole-body version of HNS.

Hardware Improvements. We currently use the HTC VIVE Pro and ZED mini camera as the display devices for our MR learning environment alongside a custom sensing glove. This setup is functional, but requires a tethered connection to the computer. We believe this issue will be mitigated as MR/AR technologies advance. Multiple companies have recently announced MR/AR headsets equipped with inside-out tracking [21, 62, 64]. Some of these devices [62, 64] are standalone, meaning they don't require connection to a computer. These advancements, taken together, offer an opportunity to deliver a complete mobile immersive ASL teaching experience on hardware that costs less than $400 (at the time of writing).

9 CONCLUSION

We designed, implemented, and evaluated a system that leverages mixed reality to provide an immersive and interactive learning environment for teaching ASL. With scalability as the overarching goal, we designed a scalable teaching approach based on HNS, a generalized notation system that factors individual signs into a small set of primitive HNS features. We developed a portable sensing system that continuously monitors a learner's motion. The raw sensing data was then translated into HNS features to present descriptive, real-time feedback to the learner to correct any motion errors. We demonstrated our approach with hardware implementation and Unity game development to create the mixed-reality environment and learning lessons. Experiments with 60 novice users revealed a statistically significant improvement of our system in teaching ASL signs, in comparison to the traditional desktop-based learning. We expect our approach to ultimately allow the teaching of thousands of signs and be extended to other types of physical motion tasks.

ACKNOWLEDGMENTS

We sincerely thank the reviewers for their insightful comments that helped improve the paper. Special thanks to Nicholas A. Feffer for helping the learning lesson design and Brianna Aubrey for helping build ASL-HNS dictionary. This work is in part supported by the National Science Foundation under IIS-1822819, and IIS-1839379. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the funding agencies or others.

REFERENCES

[1] 2019. Lifeprint Cirriculum | ASL 101. https://www.lifeprint.com/asl101/curriculum/curriculum.htm. (2019). Online; accessed 3 November 2019.

[2] N. S. Altman. 1992. An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. *The American Statistician* 46, 3 (1992), 175–185. http://www.jstor.org/stable/2685209

[3] Fraser Anderson, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2013. YouMove: Enhancing movement training with an augmented reality mirror. *UIST 2013 - Proc. of the 26th Annual ACM Symposium on User Interface Software and Technology*, 311–320. https://doi.org/10.1145/2501988.2502045

[4] Autodesk, INC. 2018. Motionbuilder. https:/autodesk.com/maya. (2018). Online; accessed 3 November 2019.

[5] Autodesk, INC. 2019. Maya. https:/autodesk.com/maya. (2019). Online; accessed 3 November 2019.

[6] J. C. Becker and N. V. Thakor. 1988. A study of the range of motion of human fingers with application to anthropomorphic designs. *IEEE Transactions on Biomedical Engineering* 35, 2 (Feb 1988), 110–117. https://doi.org/10.1109/10.1348

[7] Heike Brock, Felix Law, Kazuhiro Nakadai, and Yuji Nagashima. 2020. Learning Three-Dimensional Skeleton Data from Sign Language Video. *ACM Trans. Intell. Syst. Technol.* 11, 3, Article 30 (April 2020), 24 pages. https://doi.org/10.1145/3377552

[8] Teak-Wei Chong and Boon-Giin Lee. 2018. American Sign Language Recognition Using Leap Motion Controller with Machine Learning Approach. *Sensors* 18, 10 (2018). https://doi.org/10.3390/s18103554

[9] C. Chuan, E. Regina, and C. Guardino. 2014. American Sign Language Recognition Using Leap Motion Sensor. In *2014 13th International Conference on Machine Learning and Applications*. 541–544. https://doi.org/10.1109/ICMLA.2014.110

[10] DGS-Korpus. 2019. Signing Gesture Markup Language (SiGML). https://www.sign-lang.uni-hamburg.de/hamnosys/input/. (2019). Online; accessed 3 November 2019.

[11] Youchen Du, Shenglan Liu, Lin Feng, Menghui Chen, and Jie Wu. 2017. Hand Gesture Recognition with Leap Motion. *CoRR* abs/1711.04293 (2017). arXiv:1711.04293 http://arxiv.org/abs/1711.04293

[12] M. Fiala. 2005. ARTag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*.

[13] Mark Fiala. 2005. Artag fiducial marker system applied to vision based spacecraft docking. In *Proc. Intl. Conf. Intelligent Robots and Systems (IROS) 2005 Workshop on Robot Vision for Space Applications*.

[14] Deen Freelon. 2013. ReCal OIR: Ordinal, Interval, and Ratio Intercoder Reliability as a Web Service. *Int. J. Internet Sci.* 8 (06 2013), 10–16.

[15] Laura Freina and Michela Ott. 2015. A literature review on immersive virtual reality in education: state of the art and perspectives. In *The International Scientific Conference eLearning and Software for Education*.

[16] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco Madrid-Cuevas, and Manuel Marín-Jiménez. 2014. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition* 47 (06 2014), 2280–2292. https://doi.org/10.1016/j.patcog.2014.01.005

[17] Thomas Hanke. 2019. HamNoSys 4 Handshapes Chart. https://www.sign-lang.uni-hamburg.de/dgs-korpus/files/inhalt_pdf/HamNoSys_Handshapes.pdf. (2019). Online; accessed 3 November 2019.

[18] Thomas Hanke. 2019. HamNoSys-Hamburg Notation System for Sign Languages. https://www.sign-lang.uni-hamburg.de/dgs-korpus/index.php/hamnosys-97.html. (2019). 2019-11-13 09:12:05 -0500.

[19] Hanke, Thomas. 2004. HamNoSys—Representing sign language data in language resources and language processing contexts. In *4th International Conference on Language Resources and Evaluation(LREC)*.

[20] Haskin Lab at Yale University. 2019. ASL Signbank. https://aslsignbank.haskins.yale.edu/about/copyright/. (2019). Online; accessed 3 November 2019.

[21] David Heaney. 2019. HTC Vive Pro Is Getting Finger Tracking. (2019). https://uploadvr.com/htc-vive-finger-tracking/ Online; accessed 3 November 2019.

[22] Jiahui Hou, Xiang-Yang Li, Peide Zhu, Zefan Wang, Yu Wang, Jianwei Qian, and Panglong Yang. 2019. SignSpeaker: A Real-time, High-Precision SmartWatch-based Sign Language Translator. In *Proc. of MobiCom*.

[23] htc Inc. 2019. HTC VIVEPro. https://www.vive.com/us/product/vive-pro/. (2019). Online; accessed 3 November 2019.

[24] Achiou Inc. 2019. Achiou Winter Knit Gloves. (2019). https://www.amazon.com/gp/product/B077MLRYNN/ref=ppx_yo_dt_b_asin_title_o02_s00?ie=UTF8&psc=1 Online; accessed 3 November 2019.

[25] ARDUINO Inc. 2019. MKR1000. (2019). https://store.arduino.cc/usa/arduino-mkr1000 Online; accessed 3 November 2019.

[26] FlexPoint Inc. 2019. Flex Sensors. (2019). https://shop.flexpoint.com/ Online; accessed 3 November 2019.

[27] ValBox Inc. 2019. Cardboard Box. (2019). https://store.arduino.cc/usa/arduino-mkr1000 Online; accessed 3 November 2019.

[28] Interlink Electronics Inc. 2019. Force Sensor Datasheet. (2019). https://cdn2.hubspot.net/hubfs/3899023/Interlinkelectronics%20November2017/Docs/Datasheet_FSR.pdf Online; accessed 3 November 2019.

[29] Derek Kamper, T. George Hornby, and William Rymer. 2003. Extrinsic flexor muscles generate concurrent flexion of all three finger joints. *Journal of biomechanics* 35 (01 2003), 1581–9. https://doi.org/10.1016/S0021-9290(02)00229-4

[30] Ratchadaporn Kanawong and Aniwat Kanwaratron. 2017. Human Motion Matching for Assisting Standard Thai Folk Dance Learning. 49–53. https://doi.org/10.5176/2251-1679_CGAT17.11
[31] Bassem Khelil and Hamid Amiri. 2016. Hand Gesture Recognition Using Leap Motion Controller for Recognition of Arabic Sign Language.
[32] Terry Koo and Mae Li. 2016. A Guideline of Selecting and Reporting Intraclass Correlation Coefficients for Reliability Research. *Journal of Chiropractic Medicine* 15 (03 2016). https://doi.org/10.1016/j.jcm.2016.02.012
[33] klaus krippendorff. 2011. Computing Krippendorff's Alpha-Reliability. (01 2011).
[34] S. J. Lederman, R. D. Howe, R. L. Klatzky, and C. Hamilton. 2004. Force variability during surface contact with bare finger or rigid probe. In *12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004. HAPTICS '04. Proceedings.*
[35] Hong Li, Wei Yang, Jianxin Wang, Yang Xu, and Liusheng Huang. 2016. WiFinger: Talk to Your Smart Devices with Finger-grained Gesture. In *Proc. of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16)*. ACM, New York, NY, USA, 250–261. https://doi.org/10.1145/2971648.2971738
[36] R. Likert. 1932. *A Technique for the Measurement of Attitudes*. Number nos. 136-165 in A Technique for the Measurement of Attitudes. publisher not identified. https://books.google.com/books?id=9rotAAAAYAAJ
[37] Ultraleap Ltd. 2019. Leap Motion. (2019). https://www.leapmotion.com Online; accessed 3 November 2019.
[38] Yongsen Ma, Gang Zhou, Shuangquan Wang, Hongyang Zhao, and Woosub Jung. 2018. SignFi: Sign Language Recognition Using WiFi. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 23 (March 2018), 21 pages. https://doi.org/10.1145/3191755
[39] Rajesh B. Mapari and Govind Kharat. 2016. American Static Signs Recognition Using Leap Motion Sensor. In *Proc. of the Second International Conference on Information and Communication Technology for Competitive Strategies (ICTCS '16)*. ACM, New York, NY, USA, Article 67, 5 pages. https://doi.org/10.1145/2905055.2905125
[40] G. Marin, F. Dominio, and P. Zanuttigh. 2014. Hand gesture recognition with leap motion and kinect devices. In *2014 IEEE International Conference on Image Processing (ICIP)*. 1565–1569. https://doi.org/10.1109/ICIP.2014.7025313
[41] Stefan Marks, David White, and Manpreet Singh. 2017. Getting up your nose: a virtual reality education tool for nasal cavity anatomy. In *SIGGRAPH Asia 2017 symposium on education*.
[42] Kenneth Mcgraw and S.P. Wong. 1996. Forming Inferences About Some Intraclass Correlation Coefficients. *Psychological Methods* 1 (03 1996), 30–46. https://doi.org/10.1037/1082-989X.1.1.30
[43] Pedro Melgarejo, Xinyu Zhang, Parameswaran Ramanathan, and David Chu. 2014. Leveraging Directional Antenna Capabilities for Fine-grained Gesture Recognition. In *Proc. of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14)*. ACM, New York, NY, USA, 541–551. https://doi.org/10.1145/2632048.2632095
[44] Microsoft. 2019. Kinect. (2019). https://support.xbox.com/en-US/xbox-360/accessories/kinect-sensor-components Online; accessed 3 November 2019.
[45] Motion Capture Manual. 2019. Vicon Blade. http://www.cs.uu.nl/docs/vakken/mcanim/mocap-manual/site/vicon-blade/. (2019). Online; accessed 3 November 2019.
[46] Hawkar Oagaz, Anurag Sable, Min-Hyung Choi, Wenyao Xu, and Feng Lin. 2018. VRInsole: An unobtrusive and immersive mobility training system for stroke rehabilitation. In *2018 IEEE 15th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*.
[47] Pedregosa, Fabian and Varoquaux, Gaël and Gramfort, Alexandre and Michel, Vincent and Thirion, Bertrand and Grisel, Olivier and Blondel, Mathieu and Prettenhofer, Peter and Weiss, Ron and Dubourg, Vincent and Vanderplas, Jake and Passos, Alexandre and Cournapeau, David and Brucher, Matthieu and Perrot, Matthieu and Duchesnay, Édouard. 2011. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* 12 (Nov. 2011), 2825–2830. http://dl.acm.org/citation.cfm?id=1953048.2078195
[48] Wei Pei, Guanghua Xu, Min Li, Hui Ding, Sicong Zhang, and Ailing Luo. 2016. A motion rehabilitation self-training and evaluation system using Kinect. In *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*.
[49] Panayiotis E Pelargos, Daniel T Nagasawa, Carlito Lagman, Stephen Tenn, Joanna V Demos, Seung J Lee, Timothy T Bui, Natalie E Barnette, Nikhilesh S Bhatt, Nolan Ung, et al. 2017. Utilizing virtual and augmented reality for educational and clinical enhancements in neurosurgery. *Journal of Clinical Neuroscience* 35 (2017), 1–4.
[50] PhySix Gear Sport Inc. 2019. Physix Gear Sport Waterproof Kinesiology Tape. https://www.amazon.com/gp/product/B017TH9X22/ref=ppx_yo_dt_b_asin_title_o05_s00?ie=UTF8&psc=1. (2019). Online; accessed 3 November 2019.
[51] Panneer Selvam Santhalingam, Al Amin Hosain, Ding Zhang, Parth Pathak, Huzefa Rangwala, and Raja Kushalnagar. 2020. MmASL: Environment-Independent ASL Gesture Recognition Using 60 GHz Millimeter-Wave Signals. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 1, Article 26 (March 2020), 30 pages. https://doi.org/10.1145/3381010
[52] Jiacheng Shang and Jie Wu. 2017. A Robust Sign Language Recognition System with Multiple Wi-Fi Devices. In *Proc. of the Workshop on Mobility in the Evolving Internet Architecture (MobiArch '17)*. ACM, New York, NY, USA, 19–24. https://doi.org/10.1145/3097620.3097624
[53] Haryong Song, Wonsub Choi, and Haedong Kim. 2016. Robust vision-based relative-localization approach using an RGB-depth camera and LiDAR sensor fusion. *IEEE Transactions on Industrial Electronics* 63, 6 (2016), 3725–3736.
[54] Stereolabs Inc. 2019. ZED Mini. https://www.stereolabs.com/zed-mini/. (2019). Online; accessed 3 November 2019.

[55] Toshihiro Tagami, Toshihiro Kawase, Daisuke Morisaki, Ryoken Miyazaki, Tetsuro Miyazaki, Takahiro Kanno, and Kenji Kawashima. 2018. Development of Master-slave Type Lower Limb Motion Teaching System. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

[56] I. H. Trase, Z. Xu, Z. Chen, H. Z. Tan, and J. X. J. Zhang. 2019. Flexible Electrostatic Transducers for Wearable Haptic Communication*. In *Proc. of IEEE World Haptics Conference (WHC)*.

[57] Vicon Motion Systems Ltd UK. 2019. VICON. (2019). https://www.vicon.com Online; accessed 3 November 2019.

[58] Unity Technologies. 2019. Unity Engine. https://unity.com/. (2019). Online; accessed 11 November 2019.

[59] Vicon Motion Systems. 2019. Vero Series camera. https://docs.vicon.com/display/Tracker33/Compatibility+with+Vicon+Vero+cameras. (2019). Online; accessed 3 November 2019.

[60] Vicon Motion Systems Limited. 2006. Vicon MX Hardware System Reference. http://bdml.stanford.edu/twiki/pub/Haptics/MotionDisplayKAUST/ViconHardwareReference.pdf. (2006). Online; accessed 3 November 2019.

[61] Virtual Humans Group from University of East Anglia. 2019. JASigning. http://vhg.cmp.uea.ac.uk/tech/jas/vhg2019/CWASA-plus-gui-panel.html. (2019). Online; accessed 3 November 2019.

[62] Oculus VR. 2019. Introducing Hand Tracking on Oculus Quest - Bringing Your Real Hands into VR. (2019). https://www.oculus.com/blog/introducing-hand-tracking-on-oculus-quest-bringing-your-real-hands-into-vr/ Online; accessed 11 November 2019.

[63] Steven Wall and Stephen Brewster. 2006. Feeling What You Hear: Tactile Feedback for Navigation of Audio Graphs. In *Proc. of CHI*.

[64] Julia White. 2019. Microsoft at MWC Barcelona: Introducing Microsoft HoloLens 2. (2019). https://blogs.microsoft.com/blog/2019/02/24/microsoft-at-mwc-barcelona-introducing-microsoft-hololens-2/ Online; accessed 3 November 2019.

[65] Ed.D. William G. Vicars. 2020. LifePrint. https://www.lifeprint.com/asl101/topics/highschoolcurriculum.htm. (2020). Online; accessed 1 August 2020.

[66] Gus Xia, Carter Jacobsen, Qianwen Chen, Xingdong Yang, and Roger B. Dannenberg. 2018. ShIFT: A Semi-haptic Interface for Flute Tutoring. *CoRR* abs/1803.06625 (2018). arXiv:1803.06625 http://arxiv.org/abs/1803.06625

[67] Zhongkai Zhang, Jeremie Dequidt, and Christian Duriez. 2018. Vision-based sensing of external forces acting on soft robots using finite element method. *IEEE Robotics and Automation Letters* 3, 3 (2018), 1529–1536.

[68] Zijie Zhu, Xuewei Wang, Aakaash Kapoor, Zhichao Zhang, Tingrui Pan, and Zhou Yu. 2018. EIS: A Wearable Device for Epidermal American Sign Language Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 4, Article 202 (Dec. 2018), 22 pages.