A Natural Interface for Sign Language Mathematics

Nicoletta Adamo-Villani, Bedřich Beneš, Matt Brisbin, and Bryce Hyland

Purdue University, West Lafayette 47907, USA

Abstract. The general goal of our research is the creation of a natural and intuitive interface for input and recognition of American Sign Language (ASL) math signs. The specific objective of this work is the development of two new interfaces for the Mathsigner^{tm} application. Mathsigner^{tm} is an interactive, 3D animation-based game designed to increase the mathematical skills of deaf children. The program makes use of standard input devices such as mouse and keyboard. In this paper we show a significant extension of the application by proposing two new user interfaces: (1) a glove-based interface, and (2) an interface based on the use of a specialized keyboard. So far, the interfaces allow for real-time input and recognition of the ASL numbers zero to twenty.

1 Introduction

Deaf education, and specifically math/science education, is a pressing national problem [1,2]. To address the need to increase the abilities of young deaf children in math, we have recently created an interactive computer animation program (Mathsigner^{tm}) for classroom and home learning of K-3 (Kindergarten to third grade) arithmetic concepts and signs [3]. The program, currently in use at the Indiana School for the Deaf (ISD), is a web/CD-ROM deliverable desktop application aimed at increasing the opportunity of deaf children to learn arithmetic via interactive media, and the effectiveness of hearing parents in teaching arithmetic to their deaf children. The application includes 3D animated signers that teach ASL mathematics through a series of interactive activities based on standard elementary school math curriculum. The user interacts with the application and responds to questions using mouse and keyboard.

Based on feedback collected from ISD teachers, parents and students, and from signers who have tested the application extensively, the current interface presents various limitations.

1. Young deaf children of deaf parents are likely to know the signs for the numbers but might not be familiar yet with the corresponding math symbols. In this case, the children should be able to enter the answer to a problem by forming the correct ASL hand shapes, rather than by pressing a number key.
2. Deaf children of hearing parents use the application not only to increase their math skills, but also to learn the correct signs for math terminology.

G. Bebis et al. (Eds.): ISCV 2006, LNCS 4291, pp. 70–79, 2006.

Presently, the program does not allow the students to test and get feedback on their signing skills since all interactive activities require responses in the form of mouse clicks and/or keystrokes.

3. Hearing parents, undertaking the study of the ASL signs for math terminology, can only test their ability to recognize the signs; they do not have the opportunity to self test their ability to produce the signs correctly (it is common for beginner signers to perform the signs with slight inaccuracies).

In an effort to improve on the current implementation of the program, we propose two new user interfaces which allow for real-time hand gesture input and recognition. Interface (1) uses an 18-sensors Immersion cyberglove [4] as the input device. The user wears the glove and inputs an ASL number in response to a particular math question (for instance, '8' in response to question'3+5=?'). A pre-trained neural network detects and recognizes the number sign. The result is sent to the Mathsigner^{tm} application which evaluates the answer to the question and gives feedback to the user.

Interface (2) (currently under development) is based on the use of a recently developed human-computer communication method for keyboard encoding of hand gestures (KUI) [5], and a specialized keyboard for gesture control [6]. The KUI method allows for input of any hand gesture by mapping each letter key of the keyboard to one degree of freedom of a 3 dimensional hand. Each hand configuration is visualized in real-time by the use of a 3D hand model, and encoded as an alphanumeric string. Hand posture recognition and communication with the Mathsigner^{tm} are implemented as in interface (1).

In Section 2 of the paper we present a brief overview of current approaches in sign language input and recognition. In Section 3 we describe the two new user interfaces in detail, and in Section 4 we discuss their merits and limitations, along with future work. Conclusive remarks are presented in the last section.

2 Background

'Computer technology offers the opportunity to create tools that enable literacy and learning in ways accessible to signing users' [7]. In order to be effective, these tools need to support sign language interfaces, i.e., ways of input, recognition, and display of signing gestures.

Sign language input and recognition has been an active area of research during the past decade. Currently, there are two main approaches to gesture input: direct-device and vision-based input [8,9,10]. The direct-device approach uses a number of commercially available instrumented gloves, flexion sensors, body trackers, etc. as input to gesture recognition [11,12]. Some advantages of direct devices, such as data gloves, include: direct measurement of hand and finger parameters (i.e., joint angles, wrist rotation and 3D spatial information), data input at a high sample frequency, and no line-of-sign occlusion problems. Disadvantages include: reduced user's range of motion and comfort and high cost of accurate systems (i.e., gloves with a large number of sensors –18 or 22–).

Vision based approaches use one or more video cameras to capture images of the hands and interpret them to produce visual features that can be used to recognize gestures. The main advantage of vision-based systems is that they allow the users to remain unencumbered. Main disadvantages include: complex computation requirements in order to extract usable information, line-of sign occlusion problems, and sensitivity to lighting conditions.

Recently, researchers have started to develop gesture input systems that combine image- and device- based techniques in order to gather more information about gestures, and thereby enable more accurate recognition. Such hybrid systems are often used to capture hand gestures and facial expressions simultaneously [13].

Recognition methods vary depending on whether the signs are represented by static hand poses or by moving gestures. Recognition of static signing gestures can be accomplished using techniques such as template matching, geometric feature classification, neural networks, or other standard pattern recognition methods to classify the pose [14]. Recognition of dynamic gestures is more complex because it requires consideration of temporal events. It is usually accomplished through the use of techniques such as time-compressing templates, dynamic time warping, Hidden Markov Models (HMMs) [15,16] and Bayesan Networks [17].

In this paper we are concerned with static or semi-static ASL gestures. The goal is input and recognition of ASL numbers 0-20 which are represented by static hand-shapes (numbers 0-9) and by hand gestures requiring a very limited range of motion (numbers 10-20) [2,18]. To capture the hand gestures, we have chosen a direct-device approach because research findings show that this approach yields more accurate results [19]. The specialized keyboard of interface (2) is not a whole-hand input device since the input is not derived from direct measurements of hand motions, but from measurements of the motions (keystrokes) of a device manipulated by the hand. However, the keyboard allows for intuitive and natural input of hand gestures if we consider that the layout of the key sites corresponds to the layout of the movable joints of the hand (see Figure 4). Thus, we can think of the specialized keyboard as a 'semi direct' input device.

3 Implementation

3.1 Interface (1): Glove-Based

This interface makes use of a light-weight Immersion cyberglove which provides 18 angles as inputs. The glove has two bend sensors on each finger, four abduction sensors, and sensors for measuring thumb cross-over, palm arch, wrist flexion, and wrist abduction. To recognize the sign gesture input via the glove, we have used two approaches: (1) a basic metric measure in the space of the possible glove configurations, and (2) neural networks.

Distance Metrics. For this approach, five signers used the glove to input the ASL numbers 0-20 once. A stand alone program developed in C++ was used to capture and store the hand-shapes for later comparison. During interaction within the Mathsigner^{tm}, the C++ application compares the distance measures of the input gesture to the pre-stored ones. The distance measure is the classical Euclidian metrics, where each of the two angles α and α' is compared as:

$$dist = \sqrt{(\alpha - \alpha')^2}.$$

This test is performed for each angle. If the distance measures fall within the sensitivity level, the hand shape is recognized. Based on the first-fail test, if any distance measure is larger than the sensitivity level, the hand-shape is not matched to any of the gestures in the training data set. The experimentally set level was 30^o. With this method, while speed of response was fairly high (20kHz), recognition accuracy with unregistered users (i.e., users not represented in the training data set) was low. This is due primarily to variations in users' hand size. The neural networks approach, described in the next section, provided a better solution.

Neural Networks. This approach is based on the Fast Artificial Neural Network Library, (FANN) [20] a freely available package from Sourceforge. This library supports various configurations of neural networks. We have experimented with the following two configurations. The first one is based on a single neural network for all signs, whereas the second one uses different neural networks for different signs. The first configuration involves 18 neurons on the input and 21 on the output. The input neurons correspond to the input angles from the data glove. The 21 output values define 1-of-21 possible hand gestures. While this configuration yielded fairly accurate recognition results, it did not provide high speed of recognition. The configuration described in the next paragraph provides higher accuracy rate and real-time recognition.

This configuration is the standard complete backward propagation neural network with symmetrical sigmoid activation function [20]. Instead of using one neural network, it uses a set of networks (one per sign) with 18 input neurons that corresponds to the 18 angles from the data glove. One output neuron for each network determines whether the input configuration is correct (value close to 1) or incorrect (value close to -1 because of symmetrical sigmoid function). Each neural network uses two hidden layers of completely connected neurons, each layer containing 25 neurons (see Fig. 1). The training error was set to 10^{-6} and training of all 21 neural networks for all the input sets was realized in about 10 minutes on a standard laptop with 1.6 GHz Intel Pentium. The neural networks were correctly trained after not more than 10^4 epochs.

The detection of one sign was, on the same computer, performed at the rate of about 20Hz . The accuracy rate with registered users was 90%.The accuracy rate with three unregistered users was 70%. The relatively poor performance for unregistered users is probably due to the small training set of the neural network.

Sign detection is described by the following pseudocode. It is important to note that the signs 0-10 are represented as a single sign, while numbers greater than 10 are represented as a sequence of two signs.

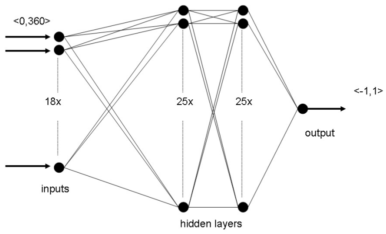


Fig. 1. The neural network has 18 inputs in the input layer, two hidden layers with 25 neurons, and 1 output neuron. This network recognizes one sign.

1. Load all trained neural networks $a[i]$.
2. Until the end of the simulation
   (a) Read the data from the data glove
   (b) for (i=0;i<10;i++) process the data with the $a[i]$.
   Remember the index of the maximum.
   (c) If the maximum is greater than 10, read the following sign and process it in the same way. The two signs define the number.
   (d) Send the recognized number to the Mathsigner^{tm}.
3. Destroy networks, free memory

Training. The training data set was provided by five signers. Each signer input the hand shapes corresponding to ASL numbers 0-20 three times. The training data set for each number is composed of 3×5 correct signs and 15 incorrect signs. The training set for each number includes the 15 ASL handshapes corresponding to that number, and 15 randomly selected ASL configurations corresponding to different numbers (provided by the same signers).

Communication with Mathsigner^{tm}. Continuous communication between the cyberglove (or the specialized keyboard, for interface (2)) and the Mathsigner^{tm} application (developed in Macromedia Director MX) was implemented using a built-in Lingo function that allows access to the system 'clipboard'. After sign recognition occurs, the C++ application formats and copies the number corresponding to the ASL hand gesture to the 'clipboard'.

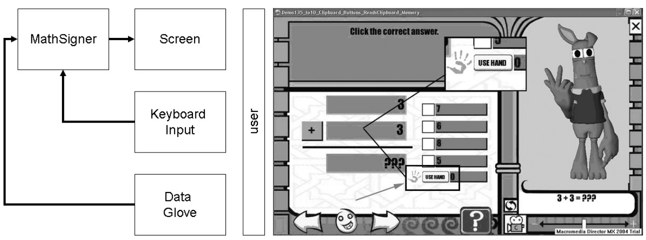
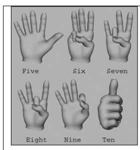


Fig. 2. Schema of the system, left; screenshot of the Mathsigner^{tm} with arrow pointing to the button (and hand icon) used to input the sign answer, right

Within Mathsigner^{tm}, the value retrieved from the 'clipboard' is displayed to the student. To submit an answer to a mathematical question, the student has two options. The student can mouse-click on one of four possible answers, only one being correct; or the student can press the 'Use Hand' button and submit the signing gesture corresponding to the answer (see Fig. 2). Upon submission of the answer, the 3D avatar signs whether the student's response is right or wrong. A video illustrating the use of interface (1) is available at http://www2.tech.purdue.edu/cgt/I3/mathinterface/.

3.2 Interface (2): Keyboard-Based

This interface makes use of a recently developed keyboard-based method for input, modelling, and animation of hand gestures (KUI) [5]. KUI is based on the realization that a hand gesture path requires the same number (26) of parameters as the letters of the English alphabet, thus, via keyboard input, it is possible to enter any hand pose in real-time. By touching a letter key, the user rotates the corresponding joint of a 3D hand a pre-specified number of degrees around a particular axis. The rotation 'quantum' induced by each keystroke can be easily changed to increase or decrease hand configuration precision. The keystrokes corresponding to particular hand poses are recorded and reduced to alphanumeric compact codes; the codes can be used for hand gesture recognition, or as keyframes to produce animation sequences. Figure 3 shows the ASL handshapes for numbers 8-10 produced with the KUI method, and their corresponding alphanumeric codes (the rotation 'quantum' was set to 10 degrees for finger flexion and 5 degrees for finger abduction).



Configuration	Code
Five	Ah2PT3
Six	a3b5c3hq4r9s4t3
Seven	a3b5c3h2m3n11o3p
Eight	a5b4c2dh2i4j10k4T2
Nine	a5b2ce7f8g2PT3
Ten * *(since the handshape is dynamic, it requires 3 codes, one for each position)	A2D4e3f9g6Hi5j9k6m4n9o6pq5r9s6Tu4w3 A2D4e3f9g6Hi5j9k6m4n9o6pq5r9s6Tu6w3 A2D4e3f9g6Hi5j9k6m4n9o6pq5r9s6Tu2w3

Fig. 3. Alphanumeric codes for ASL numbers 8-10

Recently, the KUI method was developed into a more powerful technique by the realization of a specialized, reconfigurable keyboard whose layout approximates the projection of the hand joints locations on a plain [6]. The keyboard is shown in Fig. 4.

With this keyboard, the signer inputs a hand gesture by mimicking the fingers' motion of a hand guiding another hand placed under it. The hand configuration, represented by the alphanumeric code, is visualized in real-time in a floating window. When the user is satisfied with the hand-shape, she/he clicks on the 'hand button' (see Fig. 2) in the Mathsigner^{tm} application. The alphanumeric code is converted to joint angles and recognition and communication with the Mathsigner^{tm} are achieved as in interface (1). For this interface, the training

Fig. 4. Alphabetical code for the hand joints, left; specialized keyboard, center; position of hand over keyboard, right

data set was provided by five signers who used the specialized keyboard to input ASL number configurations 0-20 three times.

4 Discussion

Both interfaces have their own merits and limitations. The main advantage of interface (1) lies in allowing the user to input signs in a natural way, without intermediary devices. Another merit is high speed of sign recognition and accuracy rate. One drawback is the high cost of the cyberglove due to the large number of sensors required to input the hand gesture with precision. Currently, the cost of the glove is a major obstacle to immediate dissemination of the program to parents and children for testing at home, and for future commercialization of the application. We are investigating more affordable types of gloves available on the market (http://www.vrealities.com/glove.html) or created by researchers specifically for input of signing gestures [21,22].

The main advantage of interface (2) is the low cost of the keypad. In addition, even if interface (2) is still under development, we anticipate higher accuracy level since variation in hand size is not an issue. The main drawback is that the specialized keyboard is not a true direct input device like the glove. While input of finger flexion (pitch rotations) is fairly natural and intuitive, input of finger abduction (yaw rotations) and wrist rotation and translation (position and orientation of the hand in 3D space) requires a certain degree of learning, abstraction, and practice. The research team is currently working on development of a new hand shaped keyboard which has an 'anatomical cradle' to support the hand, and which allows for more intuitive input of fingers' yaw rotations.

Presently, a limitation of both interfaces is that recognition is restricted to ASL numbers 0-20. In order to enable the user to answer any math question included in the application, input and recognition need to be extended to include numbers 1-1000, decimals, fractions, and the finger-spelling alphabet. In addition, one characteristic of ASL numbers is that they are signed in different ways depending on their meaning (i.e., numbers used to describe quantities–cardinals–, numbers for monetary values, numbers associated with tell-time activities, etc.) [23]. For instance, for dollar numbers 1-9, the number hand-shape is associated with a twisting motion (wrist roll) to indicate dollars. In future implementations of the interfaces, the recognition system will consider these variations.

Many aspects of the interfaces still need to be tested and improved. A comparative evaluation of the interfaces will be carried out in Fall 2006 at ISD with deaf children, parents, and ASL teachers. Besides assessing the usability of the interfaces, the full-scale evaluation will address the problem of signer-independent recognition. An ideal sign recognition system should give good recognition accuracy for signers not represented in the training data set [24]. Inter-person variations that could impact sign recognition include different signing styles, different sign usage due to geographical and social background, and fit of gloves. Many works report that recognition accuracy for unregistered signers decreases severely (by 30- 40%) when the number of signers in the training set is small, and when the signs involve significant, continuous movement [24]. For interface (1), we are concerned with the problem of degradation of recognition accuracy due to fit of the gloves, but we anticipate good recognition results considered that many of the signs are static or involve minimal motion. Studies show that recognition accuracy for unregistered signers is relatively good when only hand shapes and/or limited motion are considered [25]. So far, three unregistered signers have used interface (1); recognition accuracy was 70%.

5 Conclusions

The interfaces presented in this paper are still to be considered prototypes since many of their features are only at a first stage of development. But in spite of their limitations, they are, to our knowledge, the first sign language interfaces specifically designed for input and recognition of ASL signs for mathematics. One interface includes an 18-sensor cyberglove as the input device, and makes use of neural networks for sign recognition. The other interface uses a specialized keyboard for input of signing gestures, and neural networks for recognition.

Many applications to math and science education of the Deaf are conceivable using these interfaces, even at this stage of development. Applications to Virtual Environments are easy to envision. For example, future work involves adapting the glove-based interface for navigation and gesture input/ recognition within an Immersive Virtual Learning Environment that we have recently developed for deaf children [26].

In conclusion, research findings show that automatic analysis of Sign Language gestures has come a long way, and current work can successfully deal with dynamic signs which involve movement and which appear in continuous sequences [24]. However, much remains to be done before sign language interfaces may become commonplace in face to face computer human interaction. Aspects of gesture recognition that need further investigation and attention are building signer-independent recognition systems, and addressing the most difficult aspects of signing, such as grammatical inflections and mimetic signs, and non-manual signals (NMS). While interpretation of NMS in conjunction with gesture recognition is fundamental for understanding sign language communication in general [27], it is not so important for ASL mathematics. Therefore, considered that most ASL mathematics signs are represented by static or semi-static signs, and do not rely greatly on NMS, we believe that the realization of a natural American Sign Language interface for mathematics is a goal achievable in the near future.

Acknowledgments

This research is partially supported by the College of Technology at Purdue University (I3 grant - Proposal #00006585 - http://www.tech.purdue.edu/cgt/I3/), by the Envision Center for Data Perceptualization, and by PHS-NIH Nat. Inst. Deafness, Comm. Disorders award # 5 R01 DC005241-02. We thank Professor Ronnie Wilbur from the Department of Audiology and Speech Sciences at Purdue University for her valuable contributions to the project. In addition, we are grateful to all the signers who have participated in the development and initial evaluation of the interfaces, and to the Indiana School for the Deaf (ISD) for providing the testing ground for a thorough assessment of the interfaces in Fall 2006.

References

1. Holt, J.A., Traxler, C., Allen, T.: Interpreting the scores: A user's guide to the 9th edition stanford achievement test for educators of deaf and hard-of-hearing students. Technical report, Gallaudet Research Institute, Washington, D.C (1997)
2. Caccamise, F., Lang, H.: Signs for Science and Mathematics: A Resource Book for Teachers and Students. Rochester, NY, National Technical Institute for the Deaf, RIT (1996)
3. Adamo-Villani, N., Doublestein, J., Martin, Z.: The mathsigner: An interactive learning tool for american sign language k-3 mathematics. In: IEEE Proceedings of IV04 - 8th International Conference on Information Visualization, Los Alamitos, CA, USA, IEEE Computer Society (2004) 713–716
4. Corporation, I.: 3d interaction. http://www.immersion.com/3d/products/cyber_glove.php (2001)
5. Adamo-Villani, N., Beni, G.: Keyboard encoding of hand gestures. In: Proceedings of HCI International - 10th International Conference on Human-Computer Interaction. Volume 2. (2003) 571–575
6. Adamo-Villani, N., Beni, G.: Reconfigurable keyboard for gesture control. In: HCI International-11th International Conference on Human-Computer Interaction, on a CD-ROM (2005)
7. Frishberg, N., Corazzo, S., Day, L., Wilcox, S., Schulmeister, R.: Sign language interfaces. In: Proc. of INTERCHI-93, Amsterdam, The Netherlands (1993) 194–197
8. Huang, T., Pavlovic, V.: Hand gesture modeling, analysis, and synthesis. In: Proceedings of the International Workshop on Automatic Face and Gesture Recognition, Zurich (1995)
9. Geer, D.: Will gesture-recognition technology point the way? Computer **37** (2004) 20–23
10. Yi, B., Jr., F.C.H., Wang, L., Yan, Y.: Real-time natural hand gestures. Computing in Science and Engineering **7** (2005) 92–96, c3

11. Sturman, D.J.: Whole-Hand Input. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA (1992)
12. Sturman, D.J., Zeltzer, D.: A survey of glove-based input. IEEE Comput. Graph. Appl. **14** (1994) 30–39
13. Culver, V.R.: A hybrid sign language recognition system. In: IEEE Proceedings of the 8th International Symposium on Wearable Computers (ISWC04). Volume 00., Los Alamitos, CA, USA, IEEE Computer Society (2004) 30–33
14. Vamplew, P.: Recognition of sign language gestures using neural networks. In: Proc. of 1st Euro. Conf. Disability, Virtual Reality Assoc. Tech., Maidenhead, UK (1996) 27–33
15. Starner, T., Pentland, A.: Real-time american sign language recognition from video using hidden markov models. Technical Report MIT TR-375, Media Lab, MIT (1996)
16. Rabiner, L.: A tutorial on hidden markov models and selected applications in speech recognition. In: Proceedings of the IEEE. Volume 77(1)., Los Alamitos, CA, USA, IEEE Computer Society (1989) 257–286
17. Avils-Arriaga, H., Sucar, L.E.: Dynamic bayesian networks for visual recognition of dynamic gestures. Journal of Intelligent and Fuzzy Systems **12** (2002) 243–250
18. Flodin, M.: Signing illustrated: the complete learning guide. The Berkley Publishing Group, New York (1994)
19. Hernandez-Rebollar, J.L., Lindeman, R.W., Kyriakopoulos, N.: A multi-class pattern recognition system for practical finger spelling translation. Proc. of IEEE 4th Int'l Conference on Multimodal Interfaces (ICMI'02) (2002) 185–190
20. Nissen, S.: Fast artificial neural network library. http://leenissen.dk/fann/ (2000)
21. Hernandez-Rebollar, J.L., Lindeman, R.W., Kyriakopoulos, N.: The acceleglove: a whole hand input device for virtual reality. In: Proc. of ACM Siggraph 2002 - 29th Int'l Conference on Computer Graphics and Interactive Techniques - Sketches and Applications. (2002) 259
22. Kuroda, T., Tabata, Y., Goto, A., Ikuta, H., Murakami, M.: Consumer price dataglove for sign language recognition. In: Proc. of 5th Intl Conf. Disability, Virtual Reality Assoc. Tech., Oxford, UK (2004) 253–258
23. Dawnsign: Numbering in American Sign Language: Number Signs for Everyone. DawnSign Press (1998)
24. Ong, S., Ranganath, S.: Automatic sign language analysis: A survey and the future beyond lexical meaning. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 873–891
25. Su, M.: A fuzzy rule-based approach to spatio-temporal hand gesture recognition. IEEE Trans. Systems, Man, and Cybernetics, Part C: Application Rev. **30** (2000) 276–281
26. Adamo-Villani, N., Carpenter, E., Arns, L.: An immersive virtual environment for learning sign language mathematics. In: ACM Proceedings of Siggraph 2006 - 33rd International Conference on Computer Graphics and Interactive Techniques - Educators, ACM Siggraph (2006)
27. Valli, C., Lucas, C.: Linguistics of American Sign Language: a Resource Text for ASL Users. Washington, D.C.: Gallaudet University Press (2002)