Automatic Schaeffer's Gestures Recognition System

Francisco Gomez-Donoso Miguel Cazorla
Alberto Garcia-Garcia
Jose Garcia-Rodriguez
Computer Science Research Institute. University of Alicante
P.O. Box 99. E-03080. Alicante. Spain

April 10, 2016

Abstract

Schaeffer's sign language consists of a reduced set of gestures designed to help children with autism or cognitive learning disabilities to develop adequate communication skills. Our automatic recognition system for Schaeffer's gesture language uses the information provided by an RGB-D camera to capture body motion and recognize gestures using Dynamic Time Warping combined with k-Nearest Neighbors methods. The learning process is reinforced by the interaction with the proposed system that accelerates learning itself thus helping both children and educators. To demonstrate the validity of the system, a set of qualitative experiments with children were carried out. As a result, a system which is able to recognize a subset of 11 gestures of Schaeffer's sign language online was achieved.

keywords: Schaeffer's gestures, 3D gesture recognition, Human-Machine Interaction, RGB-D sensors

1 Introduction

Disabled people are a group that requires special attention from governments and people with cognitive disabilities (learning difficulties, cerebral palsy, etc.) are a special group within that one. Caregivers and educators need a way to communicate with them. In the case of children with autism, aided communication systems from low (pictures) to high tech (speech devices), have been used. However, gestural communication or sign languages are still recommended as well to provide autistic children with augmentative or alternative means of communication [29].

Several studies have been presented in fields like psychology, neuroscience or linguistics, demonstrating that the combination of speech with gestures forms an integrated and useful system during language production and comprehension [27]. In particular, several works have studied how that combination can improve children comprehension in many teaching contexts [28]. For that purpose, a

special gesture set was developed by Schaeffer et al. [18]. To the best of our knowledge, there exists no system that is able to recognize these gestures.

Gesture languages based on hand poses (static gestures) or movement patterns (dynamic gestures) have been used for implementing command and control interfaces [1–4]. Gestures, which involve spontaneous hand and arm movements that complement speech, have been proven to be a very effective tool for multimodal user interfaces [5–9]. For instance, object manipulation interfaces [10–12] use the hand for navigation, selection and manipulation tasks in virtual environments.

Several applications, such as heavy machinery control or manipulators, handling computer-generated avatars or musical interaction [13], use the hand as an efficient control device and with a high number of Degree of Freedom (DoF). In addition, some applications such as surgical immersive virtual reality simulations [14] and training systems (VGX, nd), include the manipulation of complex objects in their own definition.

Almost all Human-Computer Interaction (HCI) systems based on gestures use hand movements as their main input. Currently, the most effective hand motion capture tools are electromechanical or magnetic detection devices (data gloves) [15, 16]. These devices are placed on the hand to measure the location and angles of the finger joints. They offer the most comprehensive set of real-time measurements, they are application-independent and allow full functionality of the hand in HCI systems. However, they have several disadvantages in terms of use and cost. In addition, they also hinder the movement of the hand and require complex calibration and installation procedures to obtain accurate measurements.

Computer vision represents a promising alternative to data gloves because of its potential to provide a more natural interaction without intrusive devices. However, there are still several challenges to overcome for it to become more widely used, namely precision, processing speed, and generality. Among the different parts of the body, the hand is the most effective tool for general purpose interaction due to its communicative and manipulative functionalities. Some trends in interaction tend to adopt both modalities, thus allowing an intuitive and natural interaction.

This work focuses on part of HCI, which is the branch of computer science that studies and develops new paths in the communication process between humans and machines. HCI is a multidisciplinary field that includes topics such as artificial intelligence, design, social sciences, and natural language understanding.

In particular, we present a gesture recognition system that is specially designed for Schaeffer's gestures. The system, apart from gesture recognition, is able to show the user examples of the correct way to make the gestures, combined with speech reproduction of the described concept. Once the user imitates the gesture, the system provides the most similar one in the dictionary.

This document is structured as follows. In Section 2 we briefly introduce Schaeffer's gestures, what they are used for, and how they can help cognitively disabled people. Section 3 describes the general system architecture and explains the modules of the system: the input data, preprocessing and classification. Next, in Section 4 we present some experiments that were carried out using the gesture recognition system with different parameters over a set of recorded gestures in order to test its performance and its impact on children learning.

Finally, we draw some conclusions and outline future works in Section 5.

2 Schaeffer's gestures

Research on hand gesture falls within the behavioural analysis of interactions. Behavioural studies often involve observational methods, i.e., the adoption and/or adaptation of reliable coding systems shared by the scientific community [30]. There are many research studies that propose different taxonomies of gesture languages [30–34].

However, we are interested in a special group of patients: children with autism or cognitive learning disability. This kind of patients need a special way to learn to talk. In the year 1980, Schaeffer, Musil and Kollinzas published a book entitled "Total Communication: A signed speech program for non-verbal children" [18] in which they lay the foundations for interaction among people who are not able to speak. That work describes a complete sign language so that these people can interact with others more effectively.

The speech signed program is an example of a system of signs, as classified by Kiernan [19], in which the therapist introduces the user to the speech signed language. It follows the structure of oral language with some spoken words which are accompanied with signs. The real strength of this system is that its use is based on the child's overall development framework. The study of common development enables us to understand the communicative disorders that certain diseases cause. We can use this speech signed program without special authorization or training and it can be modified to meet the personal needs of people who might use it.

Its learning and use does not obstruct or hinder, or therefore slow the onset of language, quite the opposite, it promotes and motivates language development. Both this Alternative Communication System (ACS) and other alternative systems can be more than augmentative speech enhancers since they unlock this unique way of communication and allow others to be developed. The theoretical basis for ACS appeared in the USA in the year 1980, and a revised edition was published in 1994. The proposed system can recognize a subset of Schaeffer's gestures (see Table 1). We aim to recognize the complete Schaeffer's language in the future.

The system has been designed as a tool which educators can use with autistic children that do not speak at all. The curiosity and interest of children in computers can accelerate the learning process that begins with the children just repeating the gesture, but after some time it includes the pronunciation of the words. The system can help to promote self-correction. There are three main components in a sign: the final movement, the position of the hand (relative to the body), and the shape of the hand.

Firstly, it is necessary to use tactile help to show the children how to develop the gesture. Next, imitation aspects are very important and one of the steps where the system could be helpful. Finally, verbal support include full words, parts of words or even sentences. That part could also be improved with the system.

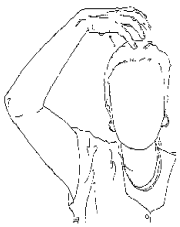
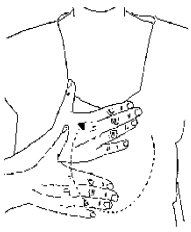
			
Water	Help	Sandwich	Sleep
			
Shower	Clean	Mom	Dad
			
Want	Sick	Dirty	

Table 1: Schaeffer's gestures.

3 Gestures recognition

In this section we explain the full system by giving an overview and describing the main device: the Microsoft Kinect v2[1]. Next, we outline the gesture acquisition and characterization process and describe the classification methodology. Finally, we provide description of the model database.

3.1 System Overview

Figure 1 shows a diagram of the system. When a person makes a gesture in front of the camera, the motion is captured by the Kinect v2. This information, summarized and packaged, is what the system understands as a gesture. The

[1] `https://dev.windows.com/en-us/kinect`

gesture object is sent to the Gesture Class Pre Selection (GCPS) module, which quickly executes with a naively selected subset of possible classes for the gesture, discarding others to improve performance. Then, both the subset of possible candidates classes and the gesture itself are sent to the classifier. The classifier compares the unknown gesture with every gesture present in the model using Dynamic Time Warping (DTW) [21], and it uses the Nearest Neighbor (NN) algorithm [22] to select the one with the shortest distance. The class is returned as the tag for the unknown gesture. This result is then sent to the user. The whole process is executed online.

There is also an offline stage that handles the training of the model. Editing and condensing processes are applied in order to obtain a fitted and error free model.

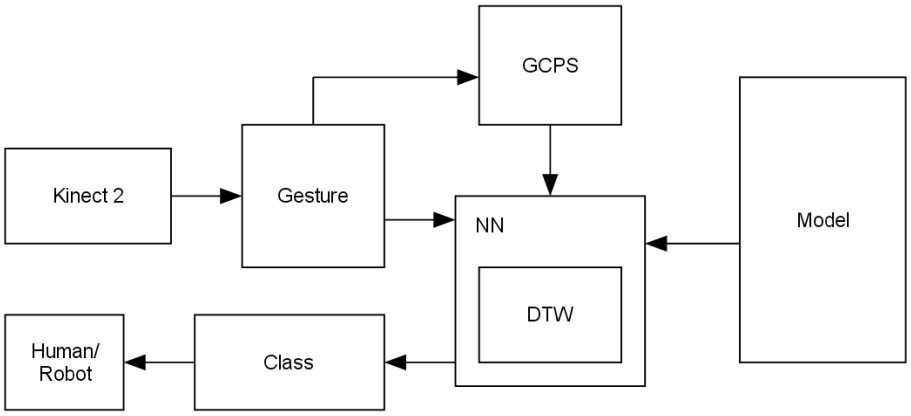


Figure 1: System architecture

3.2 Acquiring gestures: Microsoft Kinect v2

Microsoft Kinect v2 is an RGB-D camera capable of capturing the color and depth of a scene separately. The color stream is obtained with a common high definition RGB digital camera and for the depth stream it sends light beams that reflect on the surfaces and return to the sensor. By measuring the time difference between the emission and reception it calculates the depth of the element reflecting the beam. This process is known as Time of Flight (ToF).

This device is capable of capturing color images with a resolution of 1920×1080 pixels, while the depth images are captured with a resolution of 512×424. The device is also able to capture audio and its direction of origin, segment elements such as bodies and other objects, and most importantly for the task at hand: by using its API, it is also able to detect the joints of the skeleton of a person, which makes us able to detect gestures. Kinect v2 is able to capture up to 25 joints, although our recognition system only uses 11 joints for the upper part of the body of the person, the others are ignored.

3.3 Gesture Characterization

Once Kinect has captured these 11 joints of interest, two tasks are carried out before proceeding to the next module of the system: firstly, the points are grouped by joint type in order to facilitate the DTW comparison process in the classifier module, and then it runs a downsampling process in order to speed up the system. Kinect captures information at 30 fps, which means 330 points per second as long as a gesture lasts. Working with this amount of data implies a high computational cost, so the system reduces the information. The downsampling method used is k-means clustering with 20 centroids.

In addition, it is necessary to obtain independence of the angle at which the subject is located when performing the gesture and its position in the scene. For this purpose, a change on the reference system of the captured points is performed. Firstly, the system obtains two vectors, one from the neck joint to the right shoulder one, and another from the neck joint to the head one. These represent the X and Y axes of the new reference system. The Z axis is obtained by performing the cross product of these two vectors. Then, the transformation matrix is calculated from the rotation and translation between the new reference system and the camera one. At last, the transformation matrix is multiplied by all the points that compose the gesture. Some sample gestures and their associated point clouds are shown in Figure 2.

Figure 2: *Clean* and *Sleep* gestures with their associated joint point clouds.

3.4 Gesture Class Pre Selection

The classification process compares the unknown gesture with all the gestures that compose the model, making it possible to accelerate the process by comparing it only with a subset of gestures.

This GCPS module implements a series of naive but fast to evaluate rules which examine the gesture features, such as hand position or point cloud centroids, and it is able to discard some classes. For example, if the gesture for *Want* is performed with the hand below the shoulders continuously, all gestures performed above them are automatically discarded. In this way, some classes are taken as impossible and pruned, so when the comparison with the model occurs, it contains a reduced set of examples, which leads to an improved runtime. The rules are evaluated in order and if the gesture does not meet any of the rules, the classifier runs with the full model. See Table 2 for an example of how the rules are checked.

The rules implemented by the GCPS are:

- Rule 1: If the performer keeps the hand all the time over his head and there is a z-index variation below a threshold of 0.1 meters, the gesture will be classified as “Shower”.
- Rule 2: If the performer moves the hand over and between his shoulders all the time, and there is a z-index variation below a threshold of 0.05 meters, the gesture will be classified as “Dirty”.
- Rule 3: If the performer has his hands all the time closer than 0.22 meters one from the other, the selected classes will be “Help” or “Sleep”.
- Rule 4: If the performer keeps his hand under and between his shoulders, and far from his head 0.2 meters, the selected classes will be “Help”, “Want” or “Sandwich”.
- Rule 5: If the performer moves his hand over his neck all the time, the selected classes will be “Sick”, “Shower” or “Water”.
- Rule 6: If the centroid of the points that describe the hand motion when performing a gesture is located to the right of the performer, the selected classes will be “Water”, “Dad” or “Clean”.
- Rule 7: If the average distance between the hand and the head is below a threshold of 0.28 meters, the selected classes will be “Mom”, “Dad”, “Sick” or “Water”.

 “Help” gesture	Rule 1: Is the performer keeping the hand all the time over his head and there is a z-index variation below a threshold of 0.1 meters? No. Evaluate the next rule. Rule 2: Is the performer moving the hand over and between his shoulders all the time, and is there a z-index variation below a threshold of 0.05 meters? No. Evaluate the next rule. Rule 3: Is the performer having his hands all the time closer than 0.22 meters one from the other? Yes. Run the classifier with examples of the “Help” and “Sleep” classes.

Table 2: The GCPS module evaluates the rules from first to last (the order is important). If at some point a rule is evaluated to true, the classifier will be executed with the examples bonded to that rule.

3.5 Classifiers

In the classification process, the distance provided by DTW is calculated between the unknown gesture and every gesture that composes the model. The

distance between two gestures is calculated by adding the partial distances arising from comparing each point collection of each joint. That is why they are packed in such a way in the Gesture module. The NN algorithm is then executed and its class is returned as the label for the unknown gesture.

An early abandon technique is also applied as follows: after each comparison between the unknown gesture and a gesture of the model, a check is performed to determine if the distance returned is the minimum distance so far, in which case it is stored. This distance is sent to the following comparison as a threshold value and if at some point of the comparison between two gestures the partial distance obtained is greater than that threshold, the algorithm finishes. In this way, we improve the runtime of this module [24].

3.6 Model

The model is composed of all the gestures that the system has learned. These gestures have been obtained from multiple persons who were recorded as they performed every gesture several times. Then, the gestures were labeled and stored. Within the model there are gestures that were not properly performed, mislabeled, or provide redundant or useless information. To eliminate these problems, two processes are performed.

Firstly, the editing algorithm [25] is applied so those mislabeled gestures are discarded, and then the condensing or CNN algorithm [26] is executed. The latter extracts only those examples that actually provide new information to form the final model. This whole training process is performed offline. The model used by the final recognition system is composed of 253 different gestures spread over 11 classes .

4 Experiments

Two different sets of experiments were carried out. On the one hand, we tested the system with a number of samples to prove its accuracy recognizing the proposed gestures. On the other hand, a small group of children with autism used the system under the supervision of educators to study the impact of its use on the sign language and speech learning process. A screenshot of the developed application is shown in Figure 3.

4.1 Gesture Recognition Accuracy

The experimentation consists of using the system to classify a collection of 264 gestures captured with Kinect v2. In this collection, there exist 24 samples of each gesture type performed by five different persons. The classifier was set up with various parameters in order to determine the configuration that provides the best performance:

- Downsampling with 20 centroids, with GCPS activated and using k-Nearest Neighbors (k-NN) ($k = 3$) for the classifier.
- Downsampling with 20 centroids, with GCPS deactivated and using NN for the classifier.

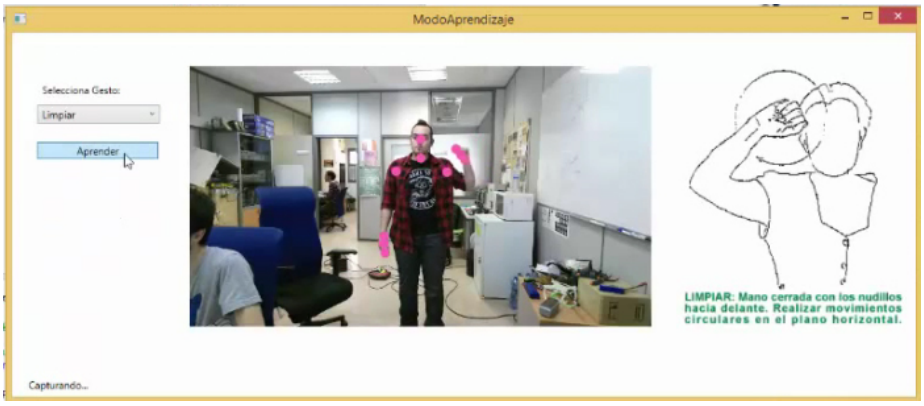


Figure 3: Screenshot of the application which shows a person choosing a gesture and performing it. The recognized gesture is shown on the right side.

- Downsampling with 20 centroids, with GCPS activated and using NN for the classifier.
- Downsampling with 10 centroids, with GCPS activated and using NN for the classifier.

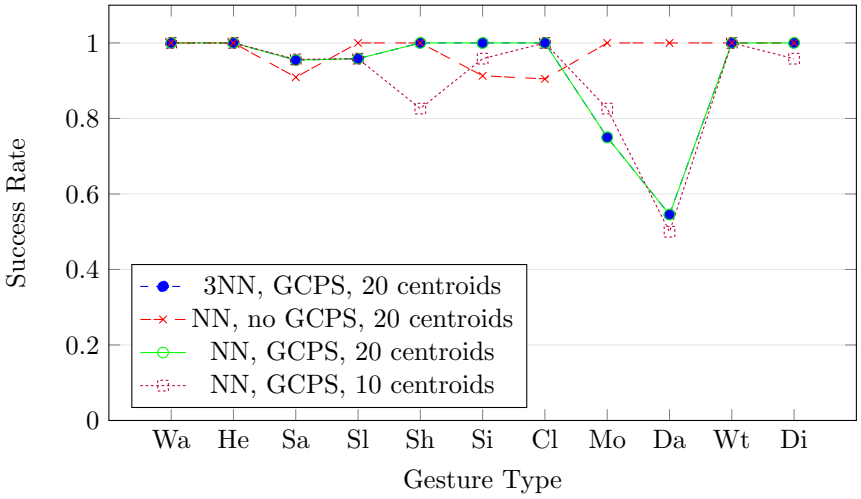


Figure 4: Success rate with the different parameter configurations.

Figure 4 shows the results for the different parameter configurations. The system setup that provides the best success rate is the one downsampled with 20 centroids, with the GCPS module deactivated and using NN for the classifier method. However, this configuration requires too much runtime, making it impractical for real-time uses, which is what this project aims to address. Figure 5 shows that the fastest method for gesture classification tasks is the 10 centroids, with the GCPS module activated and using NN, but its success rate is below the threshold of acceptance. The second fastest system configuration,

the one downsampled with 20 centroids with the GCPS module activated and using NN, provides a very high success rate, making it the best option with a reasonable ratio of success rate to elapsed time. Figure 5 shows the average elapsed run time of a five cross validation round for these four configurations (a round is composed of 55 unknown gestures classifications).

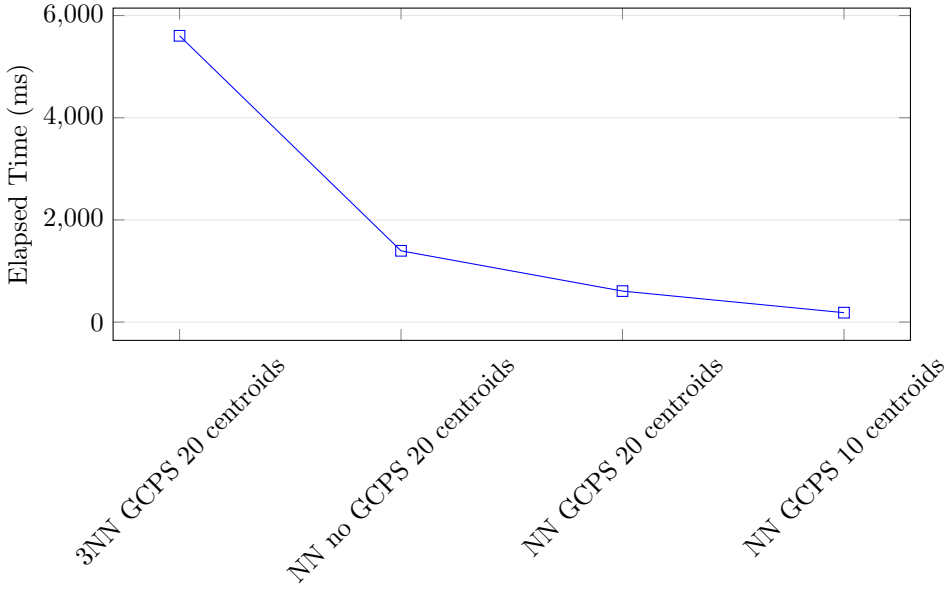


Figure 5: Execution time for a five cross validation round.

Table 3: Timing for different configurations. All values are in ms.

5CV round	3NN GCPS 20 centroids			NN no GCPS 20 centr.	NN GCPS 20 centroids			NN GCPS 10 centroids		
	Total time	GCPS time	Class. time	Total time	Total time	GCPS time	Class. time	Total time	GCPS time	Class. time
1	5642	97	5545	1519	720	99	621	212	62	150
2	5396	58	5338	1463	563	59	504	179	29	150
3	6113	66	6047	1465	644	68	576	201	34	167
4	5708	60	5648	1399	632	63	569	190	31	159
5	5158	54	5104	1129	473	51	422	138	24	114

As seen in Table 3, running the system with the GCPS module activated gives almost no impact in the overall runtime yet it improves the classification stage runtime. As exposed in Section 3.4 the GCPS module preselects some classes, decreasing the model size thus producing less comparisons in the classification stage.

Although the GCPS module introduces an error, it improves the execution time in every case while providing a high success rate, so its use is justified. We can see how the configurations with 20 centroids improve the execution time as well, while the 10 centroids one provides the best runtime but fails in terms

success rate. Regarding the classification algorithm, the k-NN ($k = 3$) provides a high success rate but with a prohibitive processing time. Instead of that, the best option is to use the nearest neighbor algorithm, which not only provides a high success rate but it is also faster.

So, in the light of the experiments, the best setup for the gesture classification task is provided by the NN, GCPS activated and 20 centroids system.

The evaluation of the evolution of the gesture learning process should be based on two main groups of features: sign component (shape, position respect body and final movement), and conditions (in presence of the object or in absence of the object). Finally, it is also important to consider if it was necessary full help, partial help or absence of help by the educator.

4.2 Learning Improvement

To evaluate the impact of the system in the children learning process there are three main stages: gesture imitation, gesture and speech imitation, and autonomous speech.

The study with a large group of children in parallel with and without the use of the system was unable because of the absence of a large number of children in the same learning stage. The system is used mainly with children between 2 and 5 years and only a few children (1 or 2) begin at the same time.

However, the qualitative evaluation of the educators indicates that, in all cases, the learning process is accelerated. Once they pass the phase in which a tactile help is necessary, the use of the system stimulates the visual imitation and fosters self-correction thus reducing the average time to achieve autonomous spoken communication.

5 Conclusions

This approach provides an innovative, customizable and reliable system for Schaeffer's gesture language detection and learning using Kinect v2. The main use case of the system is providing a tool to help children with autism to communicate with gestures and speech. The system has been tested with excellent results with a group of children between two and five years old. It is also oriented to human-machine interaction for everyone, including people with cognitive disabilities, who can use this system to communicate with another person or a robot companion.

The system can only recognize a subset of 11 different gesture classes, but we aim, as a future work, to recognize the whole of Schaeffer's sign language and implement a system to detect when a gesture starts and ends in order to create a continuous real-time classification system. The main drawback of the proposed system is the GCPS module, as it is necessary to achieve real-time response, but its use decreases the accuracy of the system. In order to improve the performance of the system and make it more suitable for a real-time application with more gestures in the model, we also aim as a future work to parallelize the NN module. In addition, the whole system could be deployed in a heterogeneous computing platform for mobile robots such as the NVIDIA Jetson TK1, which features a multi-core CPU together with a many-core GPU for parallelization with low power consumption.

The proposed system has been tested with children, obtaining a positive qualitative evaluation from the educators. We could not get a quantitative evaluation due to the small population we can reach in a short period of time. We plan as a future work evaluate the system with more children in order to get a representative population.

Acknowledgments.

This work has been supported by the Spanish Government DPI2013-40534-R Grant, supported with Feder funds. We would like to thank APSA, an association that helps people with mental handicaps in Alicante (Spain) for their collaboration in the experiments with autistic children.

References

[1] F. Quek, "Unencumbered gestural interaction". IEEE Multi-Media 3, 1996.

[2] M. Turk, "Handbook of Virtual Environments: Design, Implementation, and Applications". Hillsdale: K.M. Stanney, 2002.

[3] S. Lenman, "Using marking menus to develop command sets for computer vision based hand gesture interfaces". NY: ACM Press, 2002.

[4] M. Nielsen, "A procedurefor developing intuitive and ergonomic gesture interfaces for HCI". 5th International Gesture Workshop, 2003.

[5] A. Wexelblat, "An approach to natural gesture in virtual environments". 1995.

[6] F. Quek, "Multimodal human discourse: gesture and speech". 2002.

[7] R. Bolt, "Put-that-there: voice and gesture at the graphics interface". NY: ACM Press, 1980.

[8] D.B. Koons, "Iconic: speech and depictive gestures at the human-machine interface".NY: ACM Press, 1994.

[9] M. Billinghurst," Put that where? Voice and gesture at the graphics interface". 1998.

[10] D. Bowman, "Principles for the design of performance-oriented interaction techniques". Hillsdale: Lawrence Erlbaum Associates, 2002.

[11] J. Gabbard, "A taxonomy of usability characteristics in virtual environments". Department of Computer Science, University of Western Australia, 1997.

[12] V. Buchman, "ingARtips: gesture based direct manipulation in augmented reality". NY: ACM Press, 2004.

[13] D. Sturman, "Whole hand input". MIT, 1992.

[14] A. Liu, "A survey of surgical simulation: applications, technology, and education". 2003.

[15] D. Sturman, "A survey of glove-based input". 1994.

[16] E. Foxlin, "Motion tracking requirements and technologies". Hillsdale: Lawrence Erlbaum Associates, 2002.

[17] I. Oikonomidis, N. Kyriazis and A.A. Argyros, "Efficient model-based 3D tracking of hand articulations using Kinect", in Proceedings of the 22nd British Machine Vision Conference, BMVC 2011, University of Dundee, UK, Aug. 29-Sep. 1, 2011.

[18] B. Schaeffer, A. Musil and G.Kollinzas, "Total Communication: A Signed Speech Program for Nonverbal Children". Research Press, 1980.

[19] C. Kiernan, "Alternatives to speech: A review of research and manual and other forms of communication with the mentally handicapped and other noncommunication populations". British Journal of Mental Subnormality, 1977.

[20] A. Rebollo et al., "Diccionario de signos para alumnos con necesidades educativas especiales en el área de comunicación/lenguaje : programa de comunicación total habla signada de B. Schaeffer". Conserjería de Educación y Universidades de la región de Murcia, 2011.

[21] H. Sakoe and S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, Acoustics, Speech and Signal Processing, IEEE Transactions on, vol 26, 1978.

[22] S. Arya, et al, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions". University of Maryland, 1994.

[23] S. P. Lloyd, "Least square quantization un PCM". Bell Telephone Laboratories Paper, 1982.

[24] J. Li and Y. Wang, "EA DTW: Early abandon to accelerate exactly warping matching of time series". College of Computer Science and Technology, Huazhong University of Science and Technology, 2007.

[25] D. S. Wilson, "Asymptotic properties of nearest neighbor rules using edited data". IEEE Transactions on Systems, Man, and Cybernetics, vol. smc-2, no 3, 1972.

[26] P.E. Hart, The condensed nearest neighbor rule. IEEE Transactions on Information Theory, IT-14(3):515516, 1968.

[27] S.D. Kelly, S.M. Manning and S. Rodak, Gesture Gives a Hand to Language and Learning: Perspectives from Cognitive Neuroscience, Developmental Psychology and Education,Language and Linguistics Compass 2 (2008).

[28] S. Wagner-Cook and S. Goldin-Meadow, The Role of Gesture in Learning: Do Children Use Their Hands to Change Their Minds?, Journal of Cognition and Development, 7(2), 211232, 2006.

[29] J.B. Schwartz, C. Nye, Improving Communication for Children with Autism: Does Sign Language Work?, EBP Briefs, 1(2), 1-17, 2006.

[30] R. Bakeman, J.M. Gottman, Observing interaction. An introduction to sequential analysis, II edn. Cambridge University Press, New York 1997.

[31] J.B. Bavelas, N. Chovil,D.A. Lawrie, A. Wade, Interactive gestures. Discourse Processes 15, 469489, 1992.

[32] P. Ekman, W.V Friesen, The repertoire of nonverbal behavior. Semiotica 1, 4998, 1969.

[33] A. Kendon, Gesture and speech: How they interact. In: Wiemann, J.M., Harrison, R.P. (eds.) Nonverbal Interaction, pp. 1345. Sage Publications, Beverly Hills, 1983.

[34] D. McNeill, Hand and Mind. The University of Chicago Press, Chicago, 1992.