A Conceptual Image-Based Data Glove for Computer-Human Interaction

Leandro A. F. Fernandes [1]
Vitor F. Pamplona[1]
João L. Prauchner[1]
Luciana P. Nedel[1]
Manuel M. Oliveira[1]

Abstract: Data gloves are devices equipped with sensors that capture the movements of the hand of the user in order to select or manipulate objects in a virtual world. Data gloves were introduced three decades ago and since then have been used in many 3D interaction techniques. However, good data gloves are too expensive and only a few of them can perceive the full set of hand movements. In this paper we describe the design of an image-based data glove (IBDG) prototype suitable for finger sensible applications, like virtual objects manipulation and interaction approaches. The proposed device uses a camera to track visual markers at finger tips, and a software module to compute the position of each finger tip and its joints in real-time. To evaluate our concept, we have built a prototype and tested it with 15 volunteers. We also discuss how to improve the engineering of the prototype, how to turn it into a low cost interaction device, as well as other relevant issues about this original concept.

1   Introduction

Data gloves were introduced by Sandin et al. [1]. Since then, gloves are highly used devices in virtual reality environments, having been mapped to many selection and manipulation techniques [2, 3, 4]. Among all commercial solutions, there are simple models capable of giving pitch and roll information of the hand of the user and of measuring finger flexure [5, 6]. However, there are special cases [7, 8] where a glove needs to map the full movement of hands and fingers. This includes pitch, yaw, roll and XYZ-translations of the hand and fingers per joint flexion and adduction. Unfortunately, this set of features are restricted only to the most expensive data gloves [9, 10, 11].

Concerned about the high cost of the most complete commercial solutions, we propose a new input device: the *Image-Based Data Glove* (IBDG). Figure 1 illustrates the overall idea of the IBDG. By attaching a camera to the hand of the user and a visual marker to each finger tip, we use computer vision techniques to estimate the relative position of the finger tips. Once we have information about the tips, we apply inverse kinematics techniques [12] in

[1]Instituto de Informática - PPGC, UFRGS, Caixa Postal 15064 - CEP 91501-970 - Porto Alegre - RS - Brazil
{laffernandes,vfpamplona,jlprauchner,nedel,oliveira}@inf.ufrgs.br

Figure 1. The Image-Based Data Glove prototype: a camera attached to the user's hand tracks his/her fingers using real-time computer vision techniques. Information about finger movements is estimated using inverse kinematics techniques.

order to estimate the position of each finger joint and recreate the movements of the fingers of the user in a virtual world. Adding a motion tracker device, we can also map pitch, yaw, roll and XYZ-translations of the hand of the user, (almost) recreating all the gesture and posture performed by the hand of the user.

The main contributions of this project are the design of the image-based data glove, whose features include:

- To perform continuous real-time tracking of finger tips positions; and

- To reproduce flexion and adduction of the fingers through inverse kinematics techniques.

We also describe the engineering of the prototype of the IBDG and present a discussion on how to improve it and make it a low cost and easy to build input device.

The remaining of the paper is organized as follows: Section 2 discusses the history and some existing commercial and non-commercial data gloves. Section 3 present the concepts behind the proposed device and describes the construction of the prototype. Section 4 describes the testbed application developed to evaluate the data glove prototype, and presents the evaluation criteria and parameters for the experiments. Section 5 presents an analysis of precision and usability of our current implementation and Section 6 makes a discussion

on how to improve the IBDG prototype. Finally, Section 7 concludes the paper with some observations and directions for future work.

2   Related Work

Several glove based devices have been developed in last few decades and an in-depth discussion about many of these devices can be found in [13].

De Fanti and Sandin [1] developed the Syre's Glove at University of Chicago. This glove detects movements of fingers by using, along each finger, flexible tubes with a light source at one extremity and a photosensitive cell at the other. As users bent their fingers, the variation of light is sensed and correlated to the finger bending.

Zimmerman developed the VPL Data Glove [11], which was an improvement over existing devices and techniques. It provides real-time tracking of the hand position and orientation, as well as monitoring fingers adduction and flexion. The VPL Data Glove consists of a Lycra glove with optical fibers attached along the backs of the fingers. Finger flexion bends the fiber, which changes the attenuation of transmitted light. This attenuation is sent to a processor which determines joint angles. A magnetic tracking device is attached in the back of the hand in order to capture position and orientation of the palm. Despite its commercialization and widespread use, the VPL Data Glove is not accurate enough for complex gesture recognition.

Another commercial glove developed in the 1980s is the Dexterous HandMaster [13]. It consists of an exoskeleton-like device worn on the fingers and hand. It uses hall-effects sensors as potentiometers at the joints to measure flexion and adduction of the fingers and the thumb. Its speed (200 samples per second) and accuracy make it suitable for fine work or clinical analysis of hand function and impairment. One drawback of this device is the fact that it is hard to put on and take off, besides the need of adjustments to fit the hand properly.

The Mattel toy company manufactured a well-known glove peripheral for the Nintendo video game, the Power Glove [13]. It was inspired in VPL Data Glove, but with many modifications that allowed it to be used with a slow hardware and sold for an affordable price. While VPL Data Glove can detect yaw, pitch and roll, uses fiber optics sensors to detect finger flexure and has a resolution of 256 positions per 5 fingers, the Power Glove can only detect roll, and it uses sensors coated with conductive ink yielding a resolution of 4 positions per 4 fingers. Acoustic trackers that accurately locate the hand in space with respect to a companion unit located in the top of television monitor are mounted in the back of the hand. The trackers also provide roll orientation for the hand. It works with several Nintendo games, some of them especially designed for the Power Glove.

Currently, several data gloves with a broad range of features are available for prices

ranging from \$60 to \$16,000. The cheapest device is the P5 Glove [14], developed by extinct company Essential Reality. It consists of a support to which flexible sticks are connected. These sticks are attached to the fingers. The user moves its hand in front of a receptor "tower" containing two infrared sensors. Sensors detect visible LEDs on the glove (there are eight altogether), and convert each of them to a position (x,y,z) for the hand, and a spatial orientation in terms of pitch, yaw, and roll. The device provides six degrees of tracking at a 60Hz refresh rate. Although the P5 Glove is a low-cost device, its optical tracker suffers from line of sight limitations.

The Pinch Glove, developed by Fakespace [15], consists of flexible cloth gloves augmented with conductive cloth sewn into the tips of each of the fingers. When two or more pieces of conductive cloth come into contact with another one, a signal is sent back to the host computer indicating which fingers are being "pinched". It is distinct from whole-hand glove input devices, which report continuous joint angle information used for gesture or posture recognition. Because of this difference, the Pinch Glove is suited only for applications involving gestures with any combination of two to ten fingers, all touching another one.

The 5DT Data Glove, developed by Fifth Dimension Technologies [5], is a Lycra Glove with fiber optics flex sensors attached along the fingers to generate finger-bend data. It is capable of measuring finger flexion. Newer versions incorporate a tracking sensor attached in the back of the glove, measuring pitch and roll of the hand. Its sampling rate (200Hz) makes it suitable for real-time applications.

The DG5-VHand, manufactured by DGTech [6] is very similar to the 5DT Data Glove. Differences consist in the resolution of the finger sensors (10 bit) and the sampling rate of 100 MHz. Besides, the control board is connected to the PC and can be detached from the glove, making it possible to attach it to other parts of the body of the user.

The Fingertracking is a video tracking system developed by A.R.T. GmbH [16]. Its performance is about 20 Hz when only three fingers are tracked and 12 Hz when five fingers are tracked. Fingertracking is less invasive than our approach, but costly and hardly depending of a specific environment set up. It uses four cameras fixed in the environment and this is not always possible or desirable. On the other hand, our proposal intended to be low-cost and independent of the environment setting (a single camera is attached to the user's hand).

Finally, the ShapeHand, developed by Measurand [10], and the CyberGlove II, by Immersion [9], are the most sophisticated and expensive glove-based solutions. They are wireless, which provides more comfort and freedom of motion, are capable of measuring fingers flexion and adduction, and able to track hand movements with six degrees of freedom. They can be easily integrated with full-body motion capture systems.

Despite the large set of possibilities, only the most expensive gloves satisfy special cases where joint flexion and adduction information for each finger are needed. In this paper,
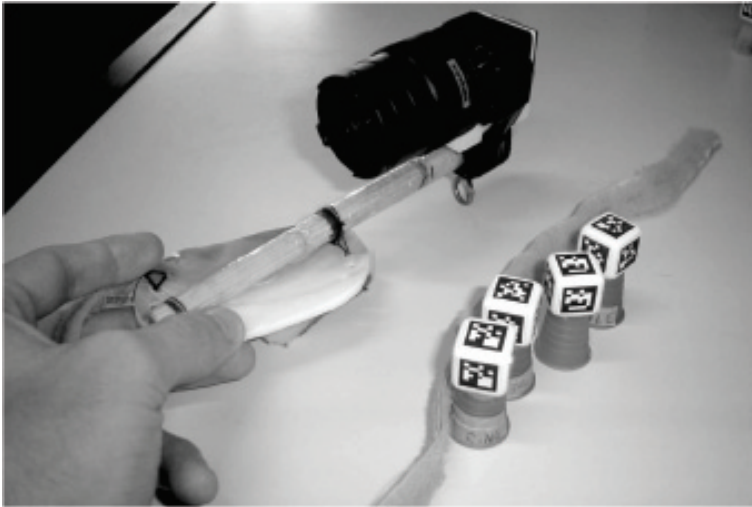
Figure 2. The Image-based Data Glove Prototype. The camera support (a wooden stick) is attached in the hand palm of the user with a Velcro strip. Thimbles are placed in fingers tips, while the camera is located in the other extremity of the stick, 23 cm far from the fingers tips.

we propose a computer-vision-based solution for fingers gestures input, which is explained in the next sessions.

3   Image-Based Data Glove

The image-based data glove is a visual tracking system that estimates the 3D position of finger tips and, in turn, uses this information to estimate the flexion and adduction of the fingers using inverse kinematics. Figure 1 shows the IBDG prototype, where a camera is used to track visual markers attached to each finger of the user's hand. In order to build such a system, three main problems need to be solved:

1. The data glove (hardware) assembling (Section 3.1);

2. The tracking system (Section 3.2); and

3. The virtual hand representation and animation with inverse kinematics (Section 3.3).

These problems were solved by integrating a simple hardware design with three well known libraries: ARToolkitPlus [17] to make the real-time tracking of the markers, V-ART [18]
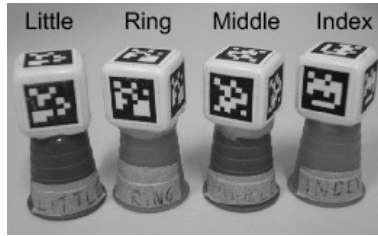
Figure 3. Dices with ARToolkitPlus patterns. Since the thumb is not taken into account in the data glove prototype, there are only four dices.
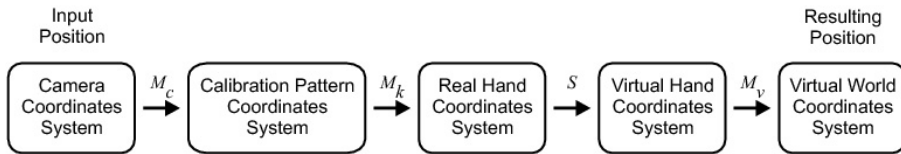

Figure 4. The sequence of system coordinates changes performed in order to transform an input marker position from the camera reference system to the virtual world reference system. M_c, M_k, S and M_v are transformation matrices described in Equation 2.

to create the virtual environment and load the hand model, and Blender inverse kinematics module [19] to compute flexion and adduction of the fingers based on their tip positions. In the next sub-sections we describe each part of this integration.

3.1   Prototype engineering

The IBDG hardware is composed by a marker attached to each finger tip and a firewire camera attached to the user's palm hand. Figure 1 shows a user wearing our IBDG prototype (without the camera connecting cable), while Figure 2 shows the prototype disassembled. In our prototype, the markers are glued to 15 mm dice attached thimbles (see Figure 3). At the faces of each dice, we have square visual patterns with 10 mm edge. The edges of the visual patterns are 5 mm smaller than the edges of the dices because we must guarantee a white margin around the pattern (a requirement of many tracking systems).

In order to avoid finger misclassification, we assign a different visual pattern to each one of them and replicate the pattern of the finger to all the faces of its dice. By doing so, we guarantee that the camera will always see at least one marker per finger, unless it is occluded by other finger. Fortunately, the occlusion problem is easily handled by analyzing the path of the fingers along time.

One should notice that we do not track the thumb, but the remaining four fingers. It is because the field of view of the camera that we use has issues in getting some thumb positions. This is, in fact, a limitation of our prototype and not of the IBDG concept. Section 6 discusses some possible solutions for this drawback.

All fingers need to be in the camera's field of view and all markers must be in focus all the time. In order to satisfy these requirements, we placed the camera some centimeters behind the user's palm (Figure 1). By doing so, the camera can see the markers from *closed hand* to *opened hand* gestures, fingers do not touch the lenses, and markers are always in focus.

Our prototype uses a Flea camera from Pointgrey Research [20]. The Flea model is a compact IEEE-1394 based digital camera with 30×31 mm size that is able to capture high quality images (1024×768 pixels) at 30 fps, being an ideal choice for demanding imaging applications. We are also using 3.5-10.5 mm Computar Varifocal Lenses [21]. The camera and the lenses set together have 115 g weight. In order to avoid both radial and tangential distortions introduced by the lenses, we estimate the intrinsic parameters of the camera and its distortion coefficients using the Camera Calibration Toolbox for MATLAB [22] and compensate such distortions using a lookup table. Notice that the estimation of the intrinsic parameters must be performed once.

3.2 Tracking system

The tracking system is responsible for estimating the 3D position and orientation of the visual patterns and from them computing the position of the finger tips. The tracking system we chosen for the prototype is the ARToolkitPlus library [17] and the pattern images used (Figure 3) were retrieved from the BCH Id-encoded patterns collection of this library. We are using ARToolKitPlus since we have verified it is more accurate and precise to estimate the position and orientation of tracked patterns than other well known tracking solutions, as the ARToolkit library [23]. Additionally, ARToolkitPlus is more robust than ARToolKit against false identification of patterns due to a CRC algorithm to restore damaged pattern images instead of performing just a comparison, as the simpler version does [24].

The tracking library retrieves the position and orientation of the pattern that is glued over the dices. Also, the retrieved information is related to the camera reference system. However, we want to estimate the location of the markers (*i.e.,* the center of the dice) in the coordinate system of the virtual world. Therefore, first we need to estimate the markers position in the camera reference system and then perform the related coordinate transformations in order to place them correctly into the virtual environment.
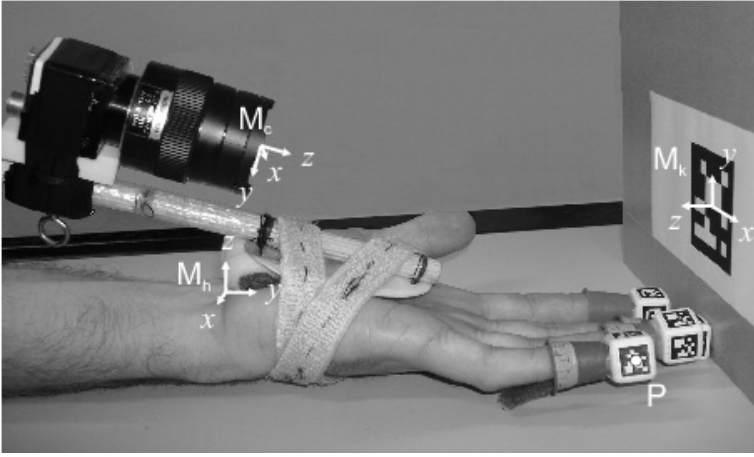
Figure 5. Relationship between P, M_c, M_h and M_k during the calibration session. P is the marker position, M_c is the camera reference system, M_h is the hand reference system and M_k is the reference system of the calibration pattern.

The position P of a marker expressed in the camera coordinate system is computed as

$$P = Q - d\,N, \tag{1}$$

where Q is the position of the center of the pattern; N is the normal vector of the pattern identified by the ARToolkitPlus; and d is half the size of the edges of the dice.

With the markers position, we need to estimate its location in the virtual world. Therefore, some system coordinates transformations must be performed, which are illustrated by Figures 4 and 5 and summarized in Equation 2

$$P' = M_v\,S\,M_k\,M_c\,P \tag{2}$$

where P' is the position at the virtual world. M_c is the matrix that models the camera position and orientation relative to a calibration pattern and M_k is the reference system of the calibration pattern relative to the real hand reference system (M_h in Figure 5). By assuming that the real and the virtual hand reference systems are the same, up to a scale factor, we can define a scale matrix S that gives the relation between the real and the virtual world. Although there exist hands of different sizes, we did not experience any kind of problem by empirically setting S just as an approximation for each user. Once in the virtual hand reference system, we transform the marker to the virtual world using the M_v transformation matrix (Equation 2).

The coordinate system transformation just described depends on the use of a calibration pattern (see the second stage of the pipeline shown in Figure 4) and the execution of a
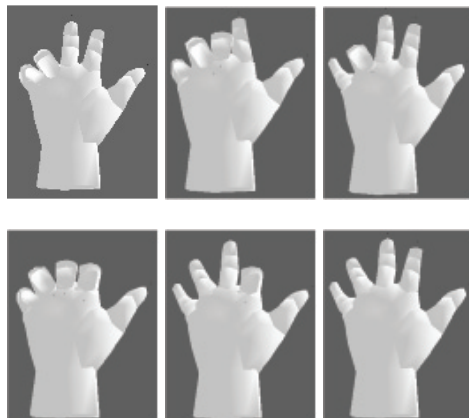
Figure 6. Hand gestures used in the testbed application. In order to not favor more skillful users, we picked up poses that are relatively easy to reproduce.

calibration section, which defines the relation between the camera and the real hand reference system. In such calibration, the user must perform a hand gesture (see Figure 5) where the relation between the reference system of the calibration pattern (M_k) and the reference system of the real hand (M_h) is known (*e.g.,* M_k can be approximated using a ruler). By doing so, the computation of the M_c is performed using:

$$M_c = M_p^{-1} \tag{3}$$

where M_p is the reference system of the camera relative to the pattern. Fortunately, M_p is retrieved by ARToolkitPlus when the pattern is identified. It is important to notice that the calibration procedure needs to be executed only once for each user, before he/she starts to use the IBDG in a real task.

3.3 Hand model

Information about the finger joints is not retrieved by the tracking system. Therefore, being able to reproduce flexion and adduction movements are of paramount importance to make the representation of the gesture of the user more realistic. We solved this problem by using an inverse kinematics tree [12] independently for each finger. The root node is on the base of the finger and the action node is on the center of the dice. Inverse kinematics is solved every frame and updates joints transformations from the prior computed pose using the current markers position.

For the virtual hand representation and animation, we used the hand model from V-

ART toolkit [18]. Since V-ART does not implement inverse kinematics techniques, we created a wrapper that integrates the Blender Kinematics Module [19] to its structure.

4   Evaluating the Device

In order to evaluate the proposed device, we have built a testbed application (Section 4.1) and performed some tests (Section 4.2) involving 15 subjects. Our population was heterogeneous, consisting of 2 women and 13 men, of which 13 right handed and 2 left handed, aging from 21 to 38 years old. Each user tested the prototype imitating 6 hand gestures (Figure 6). We picked up poses that are relatively easy to reproduce. By doing so we not favor more skillful users. The position of each tracked marker was recorded automatically by the application, while data about the users and their satisfaction was collected through a simple questionnaire.

4.1   Testbed application

The testbed application was written in the C++ using OpenGL. We used V-ART toolkit to load and display the hand model, Blender Kinematics Module 1.8 to compute joints transformations, and ARToolKitPlus 2.1 to track the visual markers. Figure 7 shows the application task window, where two virtual hand models are displayed side-by-side. In the left side is the test goal, a specified gesture of the hand that the user must imitate, and in the right side is the virtual hand that represents the actual tracked gesture of the user.

Our system takes as input a configuration file that defines:

- The V-ART polygonal model of the hand that will be used;

- The file that is used by the ARToolKitPlus to keep the camera intrinsic parameters and its distortion coefficients;

- The path to the file that defines the list of patterns that will be tracked by the ARToolKitPlus;

- The list of files that defines a hand pose for each task; and

- The list of inverse kinematics roots and tip joints that identify each finger in the model.

Also, the application takes as on-the-fly input the position of tracked markers. As output, the application shows to the user a representation of its hand pose and creates a file where all the movements of the user are logged.

Figure 7. Testbed application showing a task executed by the user. The right hand corresponds to the hand of the user when using the data glove. The left hand means the computer controlled hand holding a pose which must be imitated by the user.

The goals of this testbed application are: (i) to verify the quality of the finger tips tracking; (ii) to verify the gesture reproduction; and (iii) to collect some impressions of the users while using our IBDG prototype. Although users can freely move their arms during the tests, at this moment we are not concerned about the hand position and orientation tracking.

4.2 Tasks and procedures

A pre-test form (Appendix 1) was answered by the users, which provided us with the following information about the subjects: 66.67% of the users have already had experiences using data gloves and all of them are used to perform activities that demand greater motor coordination of the fingers.

When the application starts, the calibration section begins and the user must align his hand with the calibration pattern in order to get the correct position and orientation of the camera with respect to the reference system of the hand (see Figure 5 and the description in Section 3). After that, he/she presses the space bar key and the training section begins. During the training section the user is free to perform any movement with his/her hand and see what happens in the system. Figure 8 illustrates a training section. The camera's view is showed to the user in order to help him/her to get familiarized with the IBDG concept. When he/she becomes familiarized with the device the real tasks begin. Figure 7 shows a task screen, when two virtual hand models are displayed side-by-side. The hand on the left is the pose that the user must imitate, and the one on the right is the virtual hand controlled
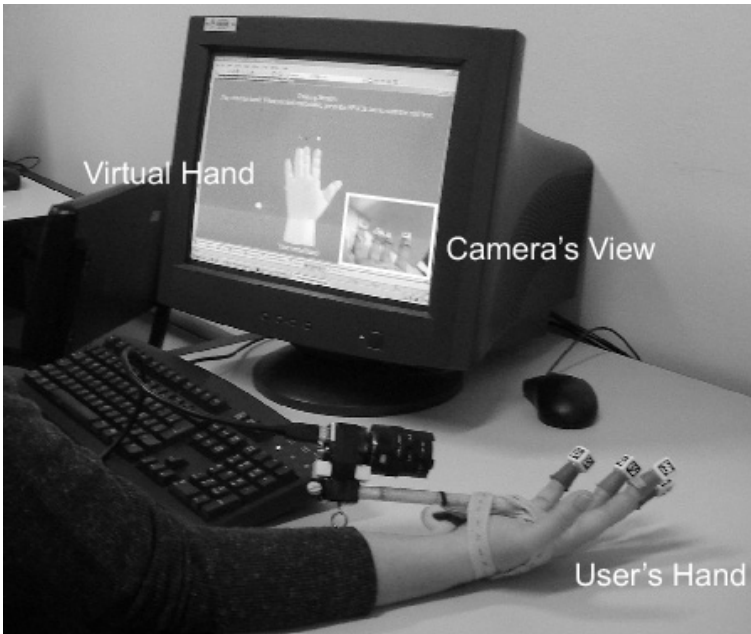
Figure 8. During the training section the user is free to perform any movement with his/her hand and see what happens in the system. During the training section the camera's view is showed to the user.

by him/her. When the user believes that the pose of his/her hand is equivalent to the left one, he/she presses a key and holds his/her hand in that pose for five seconds and then goes to the next task (see the hand poses in Figure 6). Notice that the user decides when the pose of its virtual hand is equivalent to the suggested pose. The order of the tasks is previously shuffled by the application.

A pos-test questionnaire (Appendix 2) was answered by the users. It contains questions about the comfort, the easiness and the precision of the device. The collected information is discussed in Section 5.

5   Results

Using the data collected from the tests described above, we measure the precision (Section 5.1) and the usability (Section 5.2) of the proposed device.
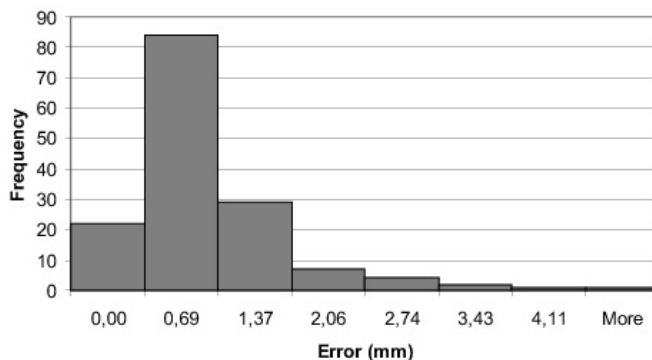
Figure 9. Observed error histogram of tracked markers positions. The mean observed error was 0.6 mm. These results show that the tracking scheme is precise.

5.1 Precision analysis

In order to measure the precision of the IBDG prototype in a usual situation, we logged the tracked markers position while users hold their hands in a suggested pose (Section 4.2). For each hand pose of each volunteer, the collected data of each finger marker position was used to compute the mean marker position. Then, we estimated the mean position variation along time as the mean distance between the mean position and the observed markers positions plus a confidence interval. The observed error for a marker position is computed as:

$$e_i = \bar{d}_i + 2.33\sigma_{\bar{d}_i} \tag{4}$$

where $\bar{d}_i$ is the mean distance between the mean marker position and the observed data; 2.33 is a Student's-t variable with $n-1$ degrees of freedom, such that the probability of a measure d belongs to the confidence interval is 99%; n is the observations number; and $\sigma_{\bar{d}_i}$ is the standard deviation.

By computing the histogram of the observed error (Figure 9) we notice that the most frequent error is smaller than 2.06 mm and the mean observed error is 0.6 mm. This data prove the tracking precision of the proposed device, since the error is a fraction of a millimeter.

5.2 Usability analysis

The mean latency time of the system is about 237 ms. It is bounded by the imaging process step (*i.e.,* ARToolKitPlus procedures). The system response can be improved by

using a faster computer. All measurements were made on a 2.4 GHz PC with 2 Gb of memory.

We verify that special care must be taken in order that hands of different sizes can be handled by the prototype of the IBDG. For instance, the focus interval of our current lenses configuration can not handle hands longer than 21.5 cm (*i.e.,* measured from middle finger tip to the beginning of the wrist). Actually, we had to reject two volunteers because their hands were too big for the current prototype.

Regarding comfort, the volunteers pointed out that our prototype can be classified as acceptable, with the value 2.73, where 5 means comfortable and 1 means uncomfortable. Also, they classify the precision of the system as 2.87, where 5 mean precise and 1 means imprecise. However, we believe that those concepts can be further improved. We can achieve better precision by re-calibrating the degrees of freedom of the inverse kinematics joints structure. Also, the comfort can be enhanced by replacing the current camera for a lighter one (the prototype weights 195 g, without the firewire cable, where about 58.97% of the weight came from the camera and the lenses set) and by changing the dices by smaller ones.

6 Discussion

Illumination changes ARToolkitPlus uses binary visual patterns. Therefore, the input image must be converted to black and white before the detection procedure. Fortunately, the library performs an automatic threshold selection before the color conversion. As a result, we did not experience any kind of problem regarding to changes of the lighting conditions nor the projection of shadows over the visual patterns. For dark places, one could use an infrared enabled camera.

Occlusion problem Occlusions happen in two situations: (i) when the hand is completely open and the camera can not see the patterns; or (ii) when a finger is in front of other. In the current prototype, when an occlusion happens, the fingers position and orientation is set to the last valid captured values to that finger. We have achieved good results with this naive solution.

A better solution for handling occlusions is the use two or more cameras like in Fingertracking system [16]. However, it is costly then our proposed single camera approach (our intention is to build a low-cost device) and the cameras must be fixed in the environment. Notice that the dependence of the environment setting is not always possible or desirable.

Another solution is to predict the fingers position and orientation by analyzing the path of the fingers along time and DOF limits. Since the fingers perform only flexion and adduction it is easy to predict where the finger tips are going from one frame to another. We believe that this solution is suitable to be incorporated in our system.

Reproducibility When asked about the reproducibility of the device, 80% of the 15 volunteers (all students of computer science) said that, given the technical specifications, they can build their own IBDG.

Performance The performance of the IBDG is bounded by the frame rate of the camera. In our prototype, the camera captures 30 frames per second, however we experienced a lower frame rate (about 13 fps on a 2.4 GHz PC with 2 Gb of memory). The bottleneck of our implementation is the image processing procedures of the ARToolkitPlus. We notice that, by setting a fixed threshold for the color conversion and reducing the resolution of the captured image from 1024×768 to 640×480 pixels, the performance is greatly improved (about 23 fps) but the identification of the patterns is compromised at gazing angles.

Improving comfort By replacing the current camera by a lighter one (*e.g.,* a micro camera or a webcam) the comfort can be improved. A wireless or bluetooh enabled camera and a lighter support would help. Some cameras with large field of view can be positioned closer to the users' hand, diminishing the weight as a lever. The comfort can also be improved by using the patterns glued directly on thimbles, for instance using five planar plastic faces attached on it. In our observations, the dices remove the natural interaction of the glove, making some poses (*e.g.,* closed hand) difficult to reproduce. Another idea is the use of a latex cleaning glove and glue the patterns on each glove finger.

Special care must be taken when placing the patterns in order to keep their planarity, due to some ARToolkitPlus assumptions, and orthogonality among the five sides of this new marker. The orthogonality is required for the Equation 1 to detect the center of the cube and the real position of the finger tip.

Allowing thumb tracking The thumb moves differently from others fingers but it is possible to track it using a camera with a large field of view.

Reducing the prototype cost The Flea camera is the most expensive component of our system and can possibly be replaced by a modern webcam.

7 Conclusions and Future Work

In this paper we proposed the idea and describe the engineering of our prototype of the *Image-Based Data Glove*. The information about finger joints is estimated by inverse kinematics techniques, so (almost) all gestures of a human hand can be mapped to virtual environments. The IBDG uses a single camera per hand, is proper for continuous tracking of finger tips position and can reproduce both flexion and adduction of finger joints. Tests were performed with 15 users imitating 6 hand gestures. From the results, we conclude that the current prototype must be improved in order to be more comfortable. However, it is precise and it can be easily replicated. Regarding future work we intend to investigate a different

assembly, with lighter materials, stronger structures and smaller markers.

We believe that the IBDG idea is promising. By investigating a different assembly to the prototype we intend to build a low-cost finger tracking solution comparable (or even better) than existing commercial ones.

Acknowledgment

References

[1] Tom A. DeFanti and Daniel J. Sandin. Final report to the national endowment of the arts. Technical Report US NEA R60-34-163, University of Illinois at Chicago Circle, 1977.

[2] Joseph J. LaViola Jr. A survey of hand posture and gesture recognition techniques and technology. Technical Report CS99-11, Department oc Computer Science, Brown University, 1999.

[3] Luciana Porcher Nedel, Carla Maria Dal Sasso Freitas, Liliane Jacon Jacob, and Marcelo Pimenta. Testing the use of egocentric interactive techniques in immersive virtual environments. In *Proceedings of the INTERACT 2003, Ninth IFIP TC13 International Conference on Human-Computer Interaction*, pages 471–478. IOS Press, September 2003.

[4] Cleber S. Ughini, Fausto R. Blanco, Francisco M. Pinto, Carla M.D.S. Freitas, and Luciana P. Nedel. EyeScope: a 3D interaction technique for accurate object selection in immersive environments. In *Proceedings of the SBC Symposium on Virtual Reality 2006*, pages 77–88, May 2006.

[5] 5DT. 5DT data glove. `http://www.5dt.com/products/pdataglovemri.html`, 2005.

[6] DGTech Engineering Solutions. DG5-VHand data glove. `http://www.dg-tech.it/vhand/eng`, 2007.

[7] Sidney S. Fels and Goeffrey E. Hinton. Glove-TalkII: an adaptive gesture-to-formant interface. In *Proceedings of the Computer Human Interaction*, pages 456–463, May 1995.

[8] Falko Kuester, Mark A. Duchaineau, Bernd Hamann, Kenneth I. Joy, and Antonio E. Uva. 3DIVS: 3-dimensional immersive virtual sculpting. In *Proceedings of the 1999 Workshop on New Paradigms in Information Visualization and Manipulation*, pages 92–96. ACM Press, 1999.

[9] Immersion Corporation. 3D interaction products. `http://www.immersion.com/3d/products/cyber_glove.php`, 2007.

[10] Measurand. ShapeHand. `http://www.measurand.com/products/ShapeHand.html`, 2007.

[11] Thomas G. Zimmerman, Jaron Lanier, Chuck Blanchard, Steve Bryson, and Young Harvill. A hand gesture interface device. In *Proceedings of the SIGCHI/GI conference on human factors in computing systems and graphics interface*, pages 189–192. ACM Press, 1987.

[12] Michael Girard and A. A. Maciejewski. Computational modeling for the computer animation of legged figures. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, pages 263–270. ACM Press, July 1985.

[13] David J. Sturman and David Zeltzer. A survey of glove-based input. *Computer Graphics and Applications*, 14(1):30–39, January 1994.

[14] Virtual Realities. P5 glove: virtual reality glove. `http://www.vrealities.com/P5.html`, 2007.

[15] Fakespace Systems Inc. Pinch glove. `http://www.fakespace.com/pinch.htm`, 2007.

[16] A.R.T. GmbH. Fingertracking. `http://ar-tracking.eu/Fingertracking.54.0.html`, 2007.

[17] D. Wagner. ARToolKitPlus. `http://studierstube.icg.tu-graz.ac.at/handheld_ar`, 2007.

[18] Carla Maria Dal Sasso Freitas and Luciana Porcher Nedel. V-ART: virtual articulations for virtual reality. `http://www.codeplex.com/vart`, 2007.

[19] Blender Foundation. Blender. `http://www.blender.org`, 2007.

[20] Pointgrey Research. Flea. `http://www.ptgrey.com/products/flea/index.asp`, 2006.

[21] CBC (AMERICA) Corp. `http://www.cbcamerica.com`, 2006.

[22] Jean-Yves Bouguet. Camera calibration toolbox for Matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc`, 2007.

[23] Hirokazu Kato. ARToolKit. `http://www.hitl.washington.edu/artoolkit`, 2007.

[24] Wagner Daniel and Schmalstieg Dieter. ARToolKitPlus for pose tracking on mobile devices. In *Proceedings of the 12th Computer Vision Winter Workshop*, Feb 2007.

Appendix 1   Pre-Test Form

1. Name: _____

2. Gender
   ◯ Male   ◯ Female

3. Age: _____

4. Education: _____

5. Hand size: _____

6. What are your dominant hand?
   ◯ Right's   ◯ Left's

7. Do you already used data gloves?
   ◯ Yes   ◯ No

8. Do you frequently do some activities that demand greater motor coordination of the fingers?
   ◯ Yes   ◯ No

   If yes, which of them?
   ◯ Chinese Balls of the health   ◯ Dactylography
   ◯ Musical instrument   ◯ Games (console/computer)
   ◯ Jugglery   ◯ Cellphone messages
   Others: _____

Appendix 2   Post-Test Form

1. How was the **confort** of the device?
   ○ 5 - Confortable   ○ 4   ○ 3   ○ 2   ○ 1 - Unconfortable

2. How was the **precision** of the device?
   ○ 5 - Precise   ○ 4   ○ 3   ○ 2   ○ 1 - Imprecise

3. How was the **hardness** of the tests?
   ○ 5 - Easy   ○ 4   ○ 3   ○ 2   ○ 1 - Hard

4. Which of the hand's pose was the **easiest** to reproduce?
   ○ Pointing   ○ Opened hand
   ○ Index flexed   ○ Litter and Ring flexed
   ○ All fingers flexed   ○ Ring flexed

5. Which of the hand's pose was the **hardest** to reproduce?
   ○ Pointing   ○ Opened hand
   ○ Index flexed   ○ Litter and Ring flexed
   ○ All fingers flexed   ○ Ring flexed

6. Given the technical specifications, could you remake the device by yourself?
   ○ Yes   ○ No

7. Suggestions and comments.
   _____
   _____
   _____
   _____