Semiautomatic Data Glove Calibration for Sign Language Corpora Building

Zdeněk Krňoul, Jakub Kanis, Miloš Železný, Luděk Müller
University of West Bohemia, Faculty of Applied Sciences, NTIS - New Technologies for the Information Society
Univerzitní 8, 306 14 Pilsen, Czech Republic
zdkrnoul@ntis.zcu.cz, jkanis@ntis.zcu.cz, zelezny@ntis.zcu.cz, muller@ntis.zcu.cz

Abstract

The article deals with a recording procedure for sign language dataset building mainly for avatar synthesis systems. Combined data glove and optical capture technique is considered. We present initial experiences with the motion capture data produced by the CyberGlove3 gloves and a set of new tools to ease the recording process, glove calibration and proper interpretation by the 3D model. It results in a more flexible solution for the sign language capture integrating manual glove calibration with an automatic initialization, time synchronization and high-resolution sensor readings.

Keywords: data glove calibration, sign language, motion capture

1. Introduction

Whilst data processing as tracking of markers, 3D reconstruction or fitting a generic skeleton model provided by the VICON motion capture system are very beneficial for capturing of sign languages, there is no such option for a processing of the CyberGlove3 motion data. In this scenario, the capturing of sign languages consists of two steps: (1) the glove calibration to angular motion data of an internal hand model/skeleton at the beginning of the capturing session and (2) re-targeting of the angular motion data to a target avatar as the data post-processing step. Full automatic calibration approaches including the CyberGlove3 (Carmel et al., 2014) are often rejected as an incomplete solution (Wang and Neff, 2013), (Kahlesz et al., 2004), unaddressed visual fidelity of hand shapes (Griffin et al., 2000) and/or complexity of the calibration process (Elliott et al., 2008). Therefore, higher flexibility in the recording procedure using CyberGlove3 gloves is still a research issue.

The calibration process consists of finding the conversion relationship between raw data of the sensor and the actual bending (rotation in one axis) of the finger segment (in degrees). The CyberGlove3 software package includes Virtual hand tool (Carmel et al., 2014) that provides communication of the data glove with the computer. The main part of this tool is the Device Configuration Utility (DCU) that provides a user interface for glove calibration. The sensor maximum standard deviation nonlinearity is 0.6 % over the full joint range (Carmel et al., 2014) and the conversion relationship is approximated by a linear transformation, i.e. a scale (a) and an offset (b) of a line equation:

$$Y = a \cdot X + b \tag{1}$$

The DCU offers an automatic calibration of the linear relationship identified just from two predefined hand shapes. Theoretically, the technique is sufficient since two different finger bends clearly determine two points in a line (i.e. linear relationship of the sensor). But the result provided by the DCU is very inefficient. As was noted earlier in (Huenerfauth and Lu, 2010), the middle, the ring and the pinky finger abductions as well as flexion are incorrectly identified from the hand shapes (a flat hand and the touch of the thumb and the index finger) because there are two ambiguous sensor readings.

The second and preferred option of the DCU is a manual calibration. The initial experiment with sign language capturing shows that it is possible to calibrate basic (simple) hand shapes like a fist or a flat palm. However, the very laborious work is calibration of the touch of the thumb/index finger and the touch of the thumb/pinky finger was not achieved anyway. In this context, problems of sensor cross-coupling are often discussed that single finger segment bend can influence multiple sensors, or that some sensors measure a different motion of the hand. The cross-coupling is solved for the finger abduction sensors (Steffen et al., 2011) or the thumb roll/abduction (Wang and Neff, 2013) for CyberGlove2. The next issue is a reuse of the calibration parameters after re-dressing of the gloves by the same subject. In this case, an average standard deviation is only 3° for a single sensor (Carmel et al., 2014), nevertheless the previously identified calibration does not match key hand shapes. This inconsistency must be taking into account while creating of the sign language corpora.

In the paper, we present initial experiences and custom tools that address the weaknesses of the current sign language recording process: allow flexible setting of the data glove parameter; time synchronization between left, right glove and body motion capturing; an automatic initialization of the calibration parameters and its manual refinement; and high resolution sensor readings of CyberGlove3 with 12-bit A/D conversion instead of precision loss by the DCU.

2. Combining Optical and Data Glove Recording

The combination of the optical and the data glove motion capturing is one possible sign language recording technique. The measurement principle of the finger bending is based on the resistive sensors that provide robust measurements of finger contacts on one or mutually between hands. In addition, the CyberGlove3 glove measures palm flex and wrist rotation like pitch and yaw. On the other hand, the

reading of one sensor is relative to the reading of the preceding finger segment or the wrist and thus we never get absolute 3D positions. Thus, the CyberGlove3 motion capture data are relative to the 3D position of the forearm.
It is necessary to determine which part of the arm will be a link for two types of the motion capture data. The first option is to use a mapping of the wrist pitch and jaw sensor to the target model and the VICON system determining only the wrist position and the forearm twist. The second and also preferred option is tracking full/global wrist rotation as *finger direction* and *palm rotation* by the VICON system. In this case, at least two markers on the hand back have to be added to two markers placed on the wrist joint. The wrist pitch and yaw of the data glove are then ignored.

3. New Data Glove Recording Procedure

We consider the combined record dataset building in two phases: (1) recording phase for raw glove motion data recording without glove calibration, (2) processing phase allowing interpretation of the raw data and including the glove calibration. For the first phase, we developed a new tool for communication with the CyberGlove3 gloves. The tool provides an interface for recording with one or two (left and right) gloves at a time and also enables necessary time synchronization between the gloves and the VICON system. For the second phase of sign language dataset building, we develop scripts for converting the raw data records in the native format to a motion data standard file format.

3.1. CyberGlove3 Recording Tool

The CyberGlove3 recording tool consists of two Python scripts to control recording with CyberGlove3 gloves. The first one is for the recording with one glove and the second one allows time-synchronized recording of both gloves. A particular command is sent to the both gloves at the same time but it can be executed by each glove with a slightly different delay depending on the processing unit of each glove. To time-synchronize the data recording we set the same internal time for both gloves by one command and then start the recording with another one. A joint time setting allows us to reach the time difference between gloves in a range of a one time frame, i.e. 0.0333 s because there are 30 time frames for each second. But the time difference can be greater and we have to keep the internal time setting until the difference is acceptable. As soon as it is acceptable, we can start the recording.
Another useful and important feature of the script is a possibility of a supporting video recording of the calibration take by a connected web camera. The exact video recording of the calibration take is important for an accurate and clear glove calibration. The control script allows the following functionality: display battery voltage, display and set internal time, display and set active (recorded) file name, list of all memory card files, start and stop recording and enabling of the video recording. Records of the glove data are stored on an internal memory card in the processing unit of each glove. In general, glove motion data can be on-line streamed to host computer and or stored on the memory card. Nevertheless according to our experience, the Wi-Fi on-line stream is not robust enough for reliable recording of large data needed for the corpus. The data are in CyberGlove3 native binary format as raw responses of the glove sensors. In advance, this option provides the recorded data without transformation by the built-in calibration of the CyberGlove3 software.

3.2. Data Preprocessing Tool

We developped a Python script for converting the raw data records to a more suitable motion data file format as TRC (Track Row Column[1]). The TRC format enables easy joining of the raw motion data for the left and right hand and an easy import to the state-of-the-art 3D character animation software. In addition, the TRC recordings of both gloves can be joined to one TRC file by the next auxiliary python script.

3.3. Semiautomatic CyberGlove3 Glove Calibration

We consider the glove calibration during the data post-processing phase and to completely avoid the built-in glove calibration of the DCU. For this purpose, we used a professional 3D character animation software MotionBuilder and created a new calibration tool.

3.3.1. Calibration Tool

The tool has the form of several embedded templates. The main template integrates the graphical user interface similar to the manual calibration dialogue panel of the DCU, see Figure 1. All templates are built from *Relations constraints* elements and connected between themselves to model all needed relationships.

Figure 1: One user interface for manual calibration of two data gloves at a time. In the figure, only the half of the interface for the right hand is shown.

MotionBuilder also supports character skeletal animation. Hereby, we can directly refer from the templates to both the target avatar model and the object represents the motion capture data imported from the given TRC file, see Figure 2. All needed mathematical operators are created by the *Relations constraints* in a simple way, that in general

[1]http://simtk-confluence.stanford.edu:8080/display/OpenSim/Marker+(.trc)+Files

can be edited and used as building blocks for very specific actions.
Nevertheless, we assume linear relationships of the sensor raw data and angular rotations of fingers. We added blocks *Scale and offset (Number)* so that each block implements one scalar linear equation. Thus, it provides a baseline framework for the manual calibration technique similar to the one originally designed (Carmel et al., 2014).
In advance, only three prototype sub-templates: *finger*, *thumb* and *pinky* are needed for the glove calibration. The *finger* prototype sub-template models relationships of the finger bones of the target model and the raw sensor data labeled as *metacarpal*, *proximal* and *abduction*, and optionally *distal flexion* respectively. Thus, there are three (four) blocks for three (four) scalar linear equations depending on the number of used sensors (18 versus 22 glove sensors). For the 18 sensors, we interpolate the distal flexion from the proximal flexion in all prototype sub-templates. Generally, the *finger* sub-template is used for index, middle and ring fingers. The second distinct prototype sub-template has to be considered for thumb due to the different treatment. Nevertheless, we assume also one more prototype sub-template to enable different treatment for the pinky finger.

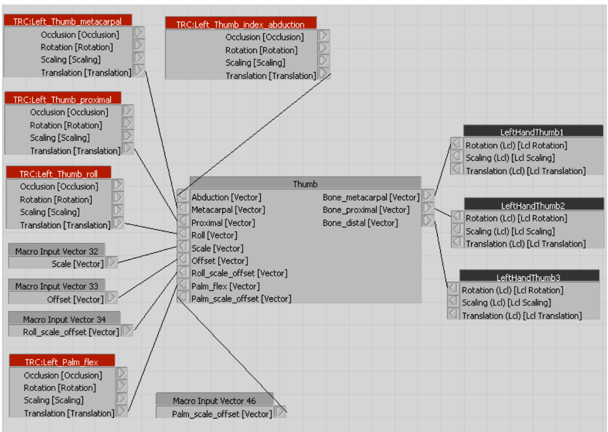

Figure 2: The prototype template for the thumb.

The prototype sub-templates are assigned to all fingers of the left and the right hand to assemble the main calibration template, see Figure 1. Each of the assigned sub-templates connects the relevant imported glove sensor data on its input with the joints of the target model skeleton on its output, see Figure 2. Each sub-template has also a user input form usable for the manual setting of the calibration parameters (the parameters controlling the scalar linear equation).

3.3.2. Target Model

The choice of the target model is done to overcome a limitation of the built-in hand model internally used by the CyberGlove3 that does not provide enough degrees-of-freedom for interpretation of sign language hand shapes. In addition, the new target model has to be appropriate also for the VICON motion capture data and potentially for facial motion capture data.
We used the 3D model for a whole human body automatically generated by a character generator[2]. The target model has all finger bones, facial bones, support for characterization of whole body motion capture data and also allows next data post-processing such as retargeting to different body proportions (sign language speaker/model).
In general, the property of such character models is the standard decomposition of its skeleton bone rotation to three basic operations: bone flexion, abduction and twist. It is done that one rotation axis of the bones in its local coordination system is in the direction of the bone head. We found that this feature is not suitable for mapping of thumb glove motion capture data. Therefore, we modified the local coordinate system of the thumb roll joints (left/right) of the generated model. We added a post-rotation of the local coordinate system so that there exists a direct mapping from the roll sensor and index/thumb abduction sensor each to one rotation axis, see Figure 3. The modification causes the rolling the thumb under the palm just by rotation of the x-axis and simultaneously *clear* side-to-side thumb/index abduction for second y-axis. For the such generated model, the post-rotation of the local coordinate system for the right hand was determined to be (-39°,-13°,27°) and mirrored values for the left hand.

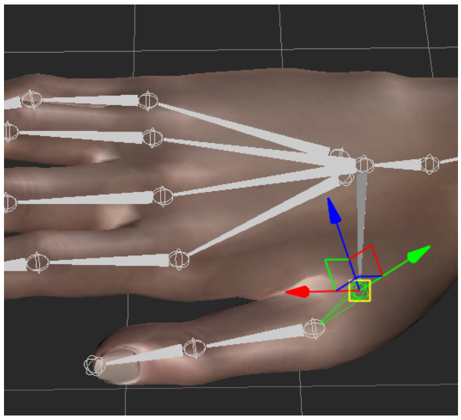

Figure 3: The modification of the local coordinate system for thumb metacarpal bone enabling mapping from thumb roll and abduction sensor.

3.3.3. Palm Flex Integration

In contrast to CyberGlove3 software, we integrate to the target model also a new relationship of the palm flex sensor. The palm flex sensor measures the flexion of the back of the hand. In particular, we consider the sensor to control the palm flexion of the target model by the pinky finger. We found that the motion capture data of the sensor are very important to reach hand shapes incorporating a touch of thumb and pinky of the target model.
Therefore, we extended the above-mentioned pinky prototype sub-template about linear mapping of the palm flex sensor data TRC_{px} to rotation of the pinky hand bone (*os metacarpi V*) as R_{ppx} (2). We added one more *Scale and offset (Number)* block and created one addition linear equation per hand. Now, the role of the pinky hand bone is

[2]https://charactergenerator.autodesk.com/

similar to the thumb metacarpal bone and it causes rolling of the pinky base joint in a direction to the palm.

$$R_{ppx} = a_{ppx} TRC_{px} + b_{ppx}, \tag{2}$$

where a_{ppx} and b_{ppx} are new calibration parameters for the pinky palm flex gain and the pinky palm flex offset.

However, this essential extension does not result in the thumb/pinky touch by the target model. We observed, for such hand shape, there is some kind of dependence of the palm flexion and the pinky metacarpal flexion. We experimentally extend the linear equation for the pinky metacarpal rotation (R_{pmx}) about 75 % of calibrated value of the palm flex sensor as:

$$R_{pmx} = a_{pmx} TRC_{pmx} + b_{pmx} + 0.75 R_{ppx}, \tag{3}$$

and the equation for the metacarpal thumb abduction (R_{tma}):

$$R_{tma} = a_{tma} TRC_{tma} + b_{tma} \tag{4}$$

about an additional palm flex mapping through new *Scale and offset (Number)* block. We got replacement of (4) as:

$$R_{tma} = a_{tma} TRC_{tma} + b_{tma} + a_{tpx} TRC_{px} + b_{tpx}, \tag{5}$$

where thumb palm flex gain a_{tpx} and thumb palm flex offset b_{tpx} are next two new parameters.

3.3.4. Automatic Initialization Tool

The goal of the initialization tool is automatic estimation of all parameters of all sensor transformations, i.e. for the linear transformation to find all gains a and offsets b. To be able to find such parameters we need to know at least two different values of each sensor and their true reference values because there are two unknown parameters a and b for each sensor. Two sensor values guarantee the exact solution of each linear transformation, however, it is not possible to cover the full range of movements of all fingers only by two hand shapes. In addition, there are even four unknown parameters for the thumb metacarpal abduction conversion (5), i.e. we need at least four pairs of sensor reading and the reference value to find these parameters.

In the experiment, the calibration take consists of five hand shapes: a flat hand, a stretching of all fingers, a fist and two "o" hand shapes, the one with thumb – index touch and the second with thumb – pinky touch respectively. However, we need to solve an overdetermined system of linear equations. We used a well-known method of least squares to find an approximate solution. This solution found can then be used in the main MotionBuilder template as a starting point for the manual refinement of the calibration parameters. We need to acquire the necessary reference joint values of the target model for the automatic calibration too. This can be done by an animator or as we did by our calibration tool manually set for a one particular calibration take.

4. Conclusion

The time-consuming and laborious calibration of two gloves is moved from a recording session to the phase of an off-line data post-processing when the presence of the signer is not required. During the recording session, all data glove takes are always stored as raw data on glove SD cards. The proposed calibration procedure involves only a calibration sequence during the recording session. We chose five predefined hand shapes that have good visual interpretation and cover the measuring ranges of the sensors. Change or extension of additional calibration hand shapes depends on the task and requirements of the researchers on the accuracy of the target hand shapes.

The time required for the calibration sequence is very short, about 2 minutes, it can be captured simultaneously for both hands. Nevertheless, about 20 minutes is needed to reach the operating temperature of the CyberGlove3 sensors before the recording session. We advise this sequence to capture one more at the end of the recording session. We recommend also to use a webcam to store video of the calibration hand shapes. The video is particularly useful for the phase of the off-line data post-processing which will help resolve potential ambiguities between the calibration hand shapes and the captured hand shapes. All tools are freely available at *http://www.kky.zcu.cz/en/download*.

5. Acknowledgements

This research was supported by the Ministry of Education, Youth and Sports of the Czech Republic, the project No. LO1506.

6. Bibliographical References

Carmel, R., Ullrich, C., Silver, J., Discoe, B., and Sharma, S., (2014). *VirtualHand v3.1.3 - User's Guide*. CYBERGLOVE SYSTEMS LLC.

Elliott, R., Glauert, J. R. W., Kennaway, J. R., Marshall, I., and Safar, E. (2008). Linguistic modelling and language-processing technologies for avatar-based sign language presentation. *Univers. Access Inf. Soc.*, 6(4):375–391, January.

Griffin, W., Findley, R., Turner, M., and Cutkosky, M. (2000). Calibration and mapping of a human hand for dexterous telemanipulation. In *ASME IMECE 2000 Symposium on Hoptic Interfaces for Virtual Environments and Teleoperator Systems*, pages 1–8.

Huenerfauth, M. and Lu, P. (2010). Accurate and accessible motion-capture glove calibration for sign language data collection. *ACM Trans. Access. Comput.*, 3(1):2:1–2:32, September.

Kahlesz, F., Zachmann, G., and Klein, R. (2004). Visual-fidelity dataglove calibration. In *Computer Graphics International 2004 (CGI 2004)*, pages 403–410. IEEE Computer Society, June.

Steffen, J. F., Maycock, J., and Ritter, H. (2011). Robust dataglove mapping for recording human hand postures. In *International Conference on Intelligent Robotics and Applications (ICIRA)*, volume 7102. Springer.

Wang, Y. and Neff, M. (2013). Data-driven glove calibration for hand motion capture. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '13, pages 15–24, New York, NY, USA. ACM.