A Developer-Oriented Visual Model for Upper-Body Gesture Characterization

Simon Ruffieux[1], Denis Lalanne[2], Omar Abou Khaled[1], and Elena Mugellini[1]

[1] University of Applied Sciences of Western Switzerland, Fribourg
{Simon.Ruffieux,Omar.AbouKhaled,Elena.Mugellini}@Hefr.ch
[2] University of Fribourg
Denis.Lalanne@unifr.ch

Abstract. This paper focuses on a facilitated and intuitive representation of upper-body gestures for developers. The representation is based on the user motion parameters, particularly the rotational and translational components of body segments during a gesture. The developed static representation aims to provide a rapid visualization of the complexity for each body segment involved in the gesture for static representations. The model and algorithms used to produce the representation have been applied to a dataset of 10 representative gestures to illustrate the model.

Keywords: natural interaction, human-computer interaction, multimodality, visualization tools, developer-oriented.

1 Introduction

Recent advances in computer-vision and in low-cost hardware and embedded systems are widening the real-time methods used to recognize body and hands gestures. A few years ago, most systems were restricted to research and game industry; nowadays the miniaturization of hardware such as inertial motion units (IMU) allows embedding them in innovative devices such as watch, clothes, etc. [1-2]; while advances in computer-vision allows efficient recognition of body movements through camera and depth sensing cameras [3]. These advances and the growing number of different types of sensors available imply more work on the design of gestures for Human-Computer Interaction in order to develop gestures that can be recognized by most sensors individually.

When designing applications for Human-Computer Interaction relying on air gestures, the developers have a large choice of potential gestures from literature and many sensors available on the market. However, they are not always aware of the implications of their preliminary choices. Developers usually design gestures for a specific system with limited considerations for the portability of the gestures to other technologies. However, the recognition rate might largely vary depending on the type of sensors and their location for a same gesture. The representation developed in this work should help providing a facilitated solution to visualize and characterize gestures from a developer point of view by helping them to design gestures and their

M. Kurosu (Ed.): Human-Computer Interaction, Part V, HCII 2013, LNCS 8008, pp. 186–195, 2013.
© Springer-Verlag Berlin Heidelberg 2013

motion with a particular focus on recognition using specific sensor in the best location or to design specific gestures that can comply all types of sensors by automatically identifying the most significant motion components of gestures. This work focuses on identifying the main motion components involved in a gesture, as retrieved by inertial motion units to infer a characterization for each body segment of the upper-body

This work takes place in the context of the FEOGARM project [4] which goal is to provide a comprehensive framework for facilitating gesture evaluation and recognition methods. This work intends to provide guidelines on how to choose and design gestures depending on the types of sensors considered.

2 Related Work

In the literature, different approaches to classify, characterize and represent gestures have been developed. Classification and taxonomy of the different gestures is a theoretical solution to describe a gesture; discriminating gestures according to their semantic. Textual or visual representations of the motion of gestures is a more practical solution, which uses different methods to describe, illustrate or store the information related to a gesture; such as mathematical definitions, specific file formats or visualization and characterization tools.

Gesture taxonomy and classification has been widely studied by psychologists and computer scientists in the context of human-human interaction [5-6] and in the context of human-computer [7-9] and human-robot interaction [10]. In these researches, several classifications and taxonomies have been proposed. The taxonomy from Pavlovic [9], which divides meaningful gestures from unintentional movements, has been often reused and extended by researchers in the literature. The meaningful gestures are subdivided in different classes according to the presence or not of motion and also according to their semantic meaning. In the work of Aigner & al. [11], they propose an interesting extension of the taxonomy through a schema resuming the information with a visual example for each class of gesture. The meaningful gestures are divided in several sub-categories: pointing gestures, semaphoric gestures (static, dynamic and stroke) are completely unrelated to the meaning and strictly learned; pantomimic gestures represent a specific task being performed, iconic (static and dynamic) gestures are used to convey information about objects or entities such as shape or size or motion path; finally manipulative gestures involve real object manipulation. These classifications are often slightly modified according to the exact subject of application. Such taxonomies are interesting for researchers to share a common language and also important when designing gestures to produce intuitive gestures according to their intended function and taxonomy.

The storage and visual representation of motion has been mostly studied in works related to music and dance, often taking inspiration from the Laban notation providing a similar idea as partitions for music [12–14]. A specific storage format linking music and gesture has been produced in the form of the GDIF format [3]. This format has been developed as a tool for standardizing the way music-related movement data are described, stored and streamed. Storage of the motion information is sensor dependent; the information cannot be stored similarly for video streams or for accelerometers.

In the HCI domain, less research has been developed toward a standardized representation of gestures. Formal definitions have been developed, for example Pavlovic & al. [9] developed the following mathematical definition in the context of a hand gesture: "*Let* $\boldsymbol{h}(t) \in \boldsymbol{S}$ *be a vector that describes the pose of hands and/or arms and their spatial position within an environment at time* $\boldsymbol{t}$ *in the parameter space* $\boldsymbol{S}$. *A hand gesture is represented by a trajectory in the parameter space* $\boldsymbol{S}$ *over a suitably defined interval* $\boldsymbol{I}$". This definition illustrates perfectly what a dynamic gesture is in mathematical terms and is useful when developing algorithms to clarify what to process and recognize; although it can efficiently visually illustrate the translation of gestures involving a single element, it is less suitable to illustrate gestures involving more elements. The visualization of the motion of gestures usually adopts a representation based on the kinematic properties of the human skeleton; a kinematic tree consisting of segments that are linked by joints [15]. This solution is widely used amongst researchers and quite efficient to understand the motion of a particular gesture however it generally requires either video or multiple consecutive pictures to illustrate a dynamic gesture. In works presenting databases of gestures, the gestures are generally illustrated using several classical approaches: with one or more pictures containing arrows to indicate the movement of a subject such as in the work of Song & al. [5], with dashed lines to indicate the final posture of the body such as in the NASA standards[1] or with videos available on a website [2].

In various fields, the characterization of specific features of gestures is used to improve or optimize processes involving motion. In the medical fields, different studies try to characterize medical gestures to improve their efficiency. In [16], they characterized the motion during chest physiotherapy; which can be seen as a repetitive tangible dynamic gesture; they monitored the force and trajectories of the hands of the physiotherapist to infer quality of the medical act to potentially to improve it. In [17], the information retrieved from a Kinect sensor is used to monitor the motion of patients effectuating in-home rehabilitation in order to characterize and improves the gestures. In the musical field, a similar approach has been developed; they use the characterization of the motion of a musician while playing to infer the relation between the sounds produced and the gestures [18].

Recently, in the work by Glomb et al. [1], focusing on the creation of a dataset of hand gestures for HCI, a table illustrating the gestures was partially characterizing the complexity of the gestures by using the most significant motion components of the hand and fingers. The work presented in this paper brings that model further by characterizing automatically each segment of the arm through its main motion components and providing a visual tool to intuitively represent the information.

3 Model

The model developed has several key points: the definition of the terms used to describe the motion components, the segments that have been taken into account, the terms to define the quantity of motion and finally the visualization tool to provide the

[1] `http://msis.jsc.nasa.gov/`

information to the users. Note that the plans and axes described in the following sections reference the definitions from the chapter "Anthropometry and Biomechanics" in Man-Systems Integration standards document from the NASA[2].

The terms that have been chosen to represent the significant component(s) of the motion of a particular segment are "None", "Static", "Translational", "Rotational" and "Complex". The **"None**" component represents the fact that the motion of the segment is not significant for the gesture. This can be inferred by detecting significant variations of the motion of a segment between different occurrences of a same gesture. The "**Static"** component represents the absence of motion of a particular segment during a gesture. If the segment is not static, the gesture would not be recognized. The "**Translational**" component represents linear motion along one of the transverse, vertical or sagittal plane. The "**Rotational**" component represents the rotation of a segment along its axis. The rotations along the two other axes are not considered in the present work as they do not have as much implications for the recognition using visual recognition. This component is mostly represented in the forearm and hand segments for the present work. The "**Complex"** component, also referred as "**Trans&Rot**" in the visual representation, indicates that both translation and rotational motions of the segment occurred during the gesture.

The gestures considered in this paper are limited to one-hand gestures and therefore, only segments corresponding to the right part of the upper-body are mentioned. We considered the assumption that the unreferenced segments are labeled "None" and thus are not significant for the gestures. The "**Torso**" segment is mostly used as reference as it tends to be static or not significant during gestures; it corresponds to the upper torso, the IMU sensor is placed on the back of the neck of the subject. The "**Arm**" segment corresponds to the right upper-arm; the sensor is placed just above the elbow. The "**Forearm**" segment corresponds to the right forearm of the subject. The sensor is placed right before the wrist where the translation and rotation of the segment is maximal. Finally the "**Hand**" segment corresponds to the right hand of the subject; the sensor is placed in the palm of the hand to avoid providing visual clues to video sensors. Note that the arm, forearm and hand IMUs are placed such that they all bear the same orientation with z-axis upward when the subject performs a T-pose.

To classify the quantity of motion in several meaningful classes, a specific color code for the visual representation has been defined along with terms to describe each class. A "**Grey**" color represents the absence of signification of the segment for the considered gesture. A "**Black**" color represents a static component, the "**Red**" color represents a small quantity of motion, the "**Orange**" color represents a medium quantity of motion and the "**Green**" color represents a large quantity of motion.

Finally the visualization tool provides a mean to rapidly and intuitively visualize the final characterization of a particular gesture. A synthetic representation of the human upper right body part has been chosen, on top of which are displayed the processed information from the algorithm.

[2] `http://msis.jsc.nasa.gov/`

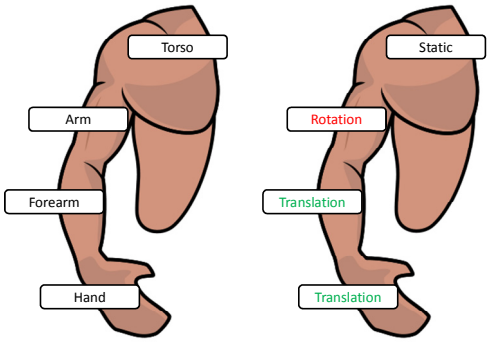

Fig. 1. An illustration of the visual representation tool. On the left, the template illustrating the considered body segments. On the right, a fictive example with a « dynamic pointing» gesture using the arm and hand only.

4 Method

4.1 Data Acquisition

The data has been acquired using the FEOGARM framework [4]. The FEOGARM software allows recording multiple sensors synchronously using distributed computers. To record the gesture database, subjects were carefully equipped with the accelerometers and asked to sit in front of a computer screen. The subject then had to read indication on what was going to happen. Once ready, the recording session started. The information was displayed to the user as in a slideshow. For each gesture, the name of the gesture was displayed on the screen along with a pre-recorded video showing the user what gesture he will have to perform; then after a short delay, the same video was replayed and the user mimics the movement simultaneously to the video. Such a method allows for automatic segmentation and annotation of the data across all sensors.

The dataset contains 10 commonly used gestures in the HCI literature recorded by 10 different subjects. Each gesture has been recorded twice per subject with 3 different resting postures and with two different lightning conditions. It contains a total of 1200 annotated gesture occurrences. The dataset contains all the raw data as acquired from the 4 Xsens MTw IMUs[3] and from one Microsoft Kinect for Windows[4].

4.2 Model Generation

In this work, in order to characterize a gesture, we processed the data acquired from the 4 IMUs. The algorithm developed automatically extracts the information corresponding to each gesture from the whole dataset using the provided annotations. Once

[3] http://www.xsens.com/en/mtw
[4] http://www.microsoft.com/en-us/kinectforwindows

extracted, we obtain, for each of the 10 gestures the 120 occurrences stored in a list. Each occurrence contains the data frames of the gesture and each data frame contains the measured linear acceleration (LinAcc), angular velocity (AngVel), Euler orientation and orientation quaternion. The linear acceleration is processed to remove the gravity component computed using the orientation quaternion.

The average quantities of motion for both the translation (1) and the rotation (2) are retrieved by summing the absolute values for each frame for a particular gesture and then averaging over all occurrences of a gesture.

$$AvgTr = \frac{\sum_0^f |LinAcc - gravity|}{f} \tag{1}$$

$$AvgRot = \frac{\sum_0^f |AngVel|}{f} \tag{2}$$

Then the principle motion components are defined according to the computed average quantities of motion. A simple comparison between the two motion values using specific threshold allows inferring the most significant component as described by the pseudo-code below:

```
if((AvgTr > TrTh) && (AvgRot > RotTh)){Complex-Trans&Rot}
if((AvgTr > TrTh) && (AvgRot < RotTh)){Translation}
if((AvgTr < TrTh) && (AvgRot > RotTh)){Rotation}
if((AvgTr < TrTh) && (AvgRot < RotTh)){Static}
```

Note that the "None" component could currently not be implemented due to the strong homogeneity of the dataset. The translation thresholds "TrTh" and the rotation thresholds "RotTh" have been inferred empirically by performing various gestures and recording their average motion quantities; in the pseudo-code, they correspond to the smallest values of the range "small motion quantity". Depending on the value of each component, the quantity of motion is characterized using the terms defined in section 3. The distinction between the classes is assessed using the following ranges; for the translation: static [0.0, 0.2], small]0.2, 0.5], medium]0.5,3] and large]3,infinite]; for the rotation: static [0.0, 0.3], small]0.3, 0.5], medium]0.5,1], large]1,infinite].

5 Results

The algorithms developed generated the values illustrated on Table 1, an intermediary phase before the automatic creation of the final visualization. Note that this table already presents the results; for example, the "WaveHello" gesture has the following translation values; large for the hand (4.94), medium for the fore-arm (1.34) and a small translation for the arm (0.30). This clearly indicates larger translation of the hand and that a sensor sensitive to translation should focus on that particular segment. However this representation in a table is not intuitive to read and complex to understand, therefore the algorithm converts it into a more human-friendly representation as shown on Fig. 2 and Fig. 3.

Table 1. The motion quantities obtained for the gesture and their motion components with respect to each segment (Hand, Fore-arm, Arm and Torso)

GestureName	Translation (H,F,A,T)	Rotation(H,F,A,T)
TakeFromScreen	(0.65,0.63,0.17, 0.07)	(0.31,0.26,0.27, 0.02)
PushToScreen	(0.61,0.62,0.26,0.09)	(0.26,0.23,0.20, 0.09)
CirclePalmRotation	(0.63,0.59,0.38, 0.13)	(0.65,0.58,0.29, 0.08)
CirclePalmDown	(0.54,0.46,0.25, 0.09)	(0.31,0.26,0.25, 0.08)
WaveHello	(4.96,1.34,0.30, 0.05)	(1.43,1.25,0.42, 0.013)
ShakeHand	(5,63,0.9,0.27, 0.07)	(2.24,1.45,0.86, 0.16)
SwipeRight	(0.55,0.51,0.10, 0.11)	(0.55,0.37,0.29,0.04)
SwipeLeft	(0.56,0.52,0.08, 0.12)	(0.57, 0.42,0.29, 0.04)
PalmUpRotation	(0.13,0.19,0.02, 0.08)	(0.76,0.65,0.13, 0.03)
PalmDownRotation	(0.05,0.03,0.08, 0.04)	(0.87,0.72,0.18, 0.03)

As previously stated, the final results are the pictures automatically generated for each gesture where the information is rapidly readable, even on a static display such as a sheet of paper.

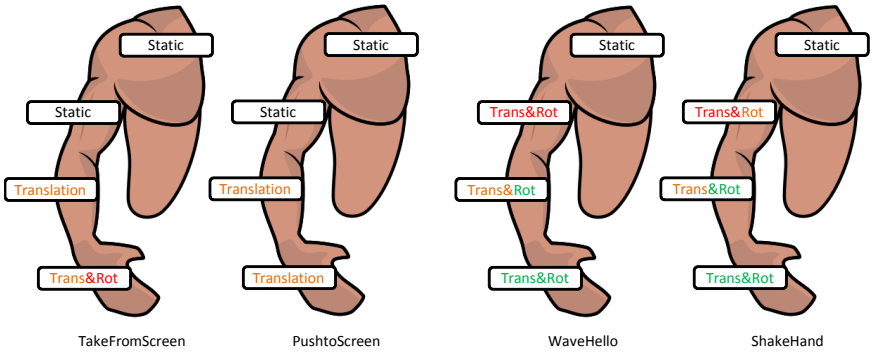

Fig. 2. The resulting characterization figures for the gestures "TakeFromScreen", "PushTo-Screen", "WaveHello" and "ShakeHand"

Using these representations, the developer can rapidly identify segments where the motion occurs and which motion component in particular is present and may pose problems or should be focused to optimize recognition. In Fig. 2 and Fig. 3, the gestures have been grouped by pairs of similar gestures. For example, "TakeFrom-Screen" and "PushToScreen" are very similar gestures where the user moved his arm towards and from the screen, the main difference being a small rotation of the hand for the first gesture while the other remains on the same posture. On the contrary, looking at representation of the gestures "PushToScreen" and "WaveHello", there is an obvious difference between the two gestures; for the latter the quantity of motion is

larger and rotation occurs; a developer might infer that a gesture containing only small or medium rotation might be more difficult to recognize using video sensors that a gesture containing large translation.

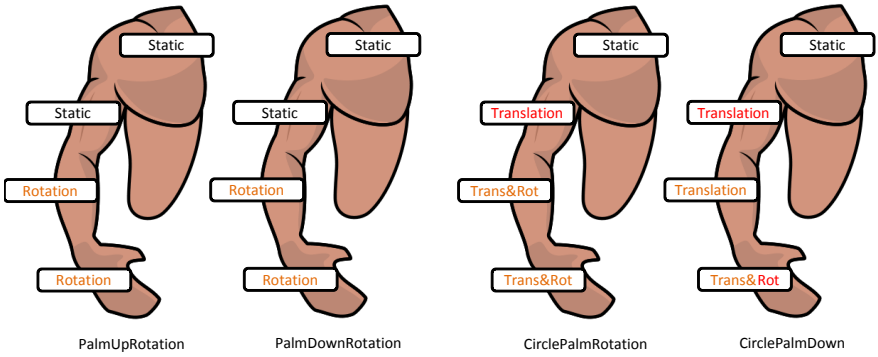

Fig. 3. The resulting characterization figures for the gestures "PalmUpRotation", "PalmDownRotation", "CirclePalmRotation" and "CirclePalmDown"

Such a representation also provides information on which segment should be monitored for optimal results. When designing gestures that should be portable across multiple types of sensors, such a representation should help to rapidly identify a common set of gestures, notably for sensors with specific body placement such as a smartwatch.

6 Conclusion and Future Work

In this paper we presented a simple and intuitive representation to characterize air gestures in the context of close human-computer interaction. The strength of the representation consists in providing developers a tool to identify the main motion components in a gesture; using this information, specific features might be added or removed in order to optimize gesture recognition with a particular type of sensor. Therefore it provides an interesting tool when designing gestures to be ported across multiple types of sensors by identifying, depending on the sensors capabilities, which segments and features to focus on. However some critics can be made about the current model; the tool does not process the data deep enough to clearly identify motion components on each plan and axes; this should be enhanced to provide more precise data and thus improve the visualization tool.

Simple and intuitive visualization and characterization tools for air gesture should become more spread as the algorithms are becoming standards. As the number of sensors on the market grows, the research should tend to focus on global gestures designed for all the heterogeneous sensors technologies used for human-computer interaction.

The algorithm should be applied to a larger dataset and more heterogeneous dataset to assess its reproducibility on other gestures and develop the "None" component class. The representation should also be enhanced to be more precise and provide more information to the developer. In order to improve the precision, the exact position and orientation of each body-segment should be computed for the whole gesture using a direct kinematic algorithm using the data from IMUs or using the skeleton data from the Kinect to define the exact space covered by each segment during a gesture. To provide more information, the obtained result should also be compared with state-of-the-art algorithms; once enough algorithms and sensors compared, an estimation of the global recognition complexity of a gesture with a particular sensor could be inferred from the motion components and quantity of a gesture. Finally, the practical utility of the visualization tool should be assessed by gesture designer/developers.

References

1. Varga, R., Prekopcsák, Z.: Creating a Database for Objective Comparison of Gesture Recognition Systems. In: Proceedings of the 15th International Student Conference on Electrical Engineering, pp. 1–6 (2011)
2. Morganti, E., et al.: A Smart Watch with Embedded Sensors to Recognize Objects, Grasps and Forearm Gestures. Procedia Engineering 4, 1169–1175 (2012)
3. Wachs, J.P., Kölsch, M., Stern, H., Edan, Y.: Vision-based hand-gesture applications. Communications of the ACM 54(2), 60 (2011)
4. Ruffieux, S., Mugellini, E., Lalanne, D., Khaled, O.A.: FEOGARMA Framework to Evaluate and Optimize Gesture Acquisition and Recognition Methods. In: Workshop on Robust Machine Learning Techniques for Human Activity Recognition; Systems, Man And Cybernetics, Anchorage (2011)
5. McNeill, D.: Language and Gesture. Cambridge University Press (2000)
6. Siegman, A.W., Pope, B.: Studies in dyadic communication. Pergamon general psychology series. Pergamon (1972)
7. Eisenstein, J., Davis, R.: Visual and linguistic information in gesture classification. In: Proceedings of the 6th International Conference on Multimodal Interfaces, ICMI 2004, p. 113. ACM Press, New York (2004)
8. Karam, M.: A framework for research and design of gesture-based human computer interactions. University of Southampton (2006)
9. Pavlovic, V.I., Sharma, R., Huang, T.S.: Visual interpretation of hand gestures for human-computer interaction: A review. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(7), 677–695 (2002)
10. Sato, E., Yamaguchi, T., Harashima, F.: Natural Interface Using Pointing Behavior for Human–Robot Gestural Interaction. IEEE Transactions on Industrial Electronics 54(2), 1105–1112 (2007)
11. Aigner, R., Wigdor, D., Benko, H., Haller, M.: Understanding Mid-Air Hand Gestures: A Study of Human Preferences in Usage of Gesture Types for HCI. Microsoft Research Technical Report MSR-TR-2012-111 (2012)
12. Camurri, A., Mazzarino, B., Ricchetti, M., Timmers, R., Volpe, G.: Multimodal Analysis of Expressive Gesture in Music and Dance Performances. In: Camurri, A., Volpe, G. (eds.) GW 2003. LNCS (LNAI), vol. 2915, pp. 20–39. Springer, Heidelberg (2004)

13. Marshall, M., Peters, N.: On the development of a system for gesture control of spatialization. In: Proceedings of the International Computer Music Conference (2006)
14. Zhao, L., Badler, N.I.: Synthesis and acquisition of laban movement analysis qualitative parameters for communicative gestures. University of Pennsylvania, Philadelphia (2001)
15. Poppe, R.: Vision-based human motion analysis: An overview. Computer Vision and Image Understanding 108(1-2), 4–18 (2007)
16. Marechal, L., et al.: Measurement System for Gesture Characterization During Chest Physiotherapy Act on Newborn Babies Suffering from Bronchiolitis. In: 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2007, pp. 5770–5773 (2007)
17. Huang, J.: Kinerehab: A kinect-based system for physical rehabilitation: a pilot study for young adults with motor disabilities. In: The Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS 2011, pp. 319–320. ACM Press, New York (2011)
18. Dobrian, C.: A Method for Computer Characterization of “Gesture” in Musical Improvisation. In: International Computer Music Conference, pp. 494–497 (2012)