

GestureAnalyzer: Visual Analytics for Pattern Analysis of Mid-Air Hand Gestures

Sujin Jang¹, Niklas Elmqvist², Karthik Ramani^{1,2}

¹School of Mechanical Engineering and ²School of Electrical and Computer Engineering
Purdue University, West Lafayette, IN, USA
{jang64, elm, ramani}@purdue.edu

ABSTRACT

Understanding the intent behind human gestures is a critical problem in the design of gestural interactions. A common method to observe and understand how users express gestures is to use elicitation studies. However, these studies require time-consuming analysis of user data to identify gesture patterns. Also, the analysis by humans cannot describe gestures in as detail as in data-based representations of motion features. In this paper, we present GestureAnalyzer, a system that supports exploratory analysis of gesture patterns by applying interactive clustering and visualization techniques to motion tracking data. GestureAnalyzer enables rapid categorization of similar gestures, and visual investigation of various geometric and kinematic properties of user gestures. We describe the system components, and then demonstrate its utility through a case study on mid-air hand gestures obtained from elicitation studies.

Author Keywords

Gesture design; gesture pattern; visualization; motion tracking data; data mining.

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: User Interfaces—*Evaluation/methodology, Graphical user interfaces (GUI)*

INTRODUCTION

Natural human motion has been actively exploited in gestural interactions made possible by successful introduction of motion sensing technologies such as low-cost depth sensing cameras (e.g., Microsoft Kinect, Leap Motion) and hand-held motion sensors (e.g., Nintendo Wii). The fundamental problems in the design of gestural interactions are (1) accurate and efficient gesture recognition, and (2) natural and intuitive gesture vocabularies to control the system. To date, many research efforts have been targeted at the development of algorithms and tools for reliable gesture recognition [22]. In

contrast, relatively little research has focused on studying natural aspects of gesture behavior. Elicitation studies, such as Wizard-of-Oz [6] and Guessability studies [39], are useful to identify the most acceptable gesture patterns from candidate user groups, and to gain insights into the design of natural gestures. These study methods have been employed to generate user-defined gesture vocabularies in various interaction scenarios [13, 27, 30, 40].

However, the analysis of user gestures requires a certain amount of time and effort to generate a user-defined gesture vocabulary. The amount of effort increases when the number of users and gestures becomes larger. Also, elicitation studies are limited to constructing descriptive taxonomies of gestures, and cannot provide detailed characteristics of gestures which can be described by kinematic and geometric features of human motion. For example, natural aspects of gestures also can be described by position and velocity profile of gesturing hands, variations in the joint angles, or the most active/inactive body part in gesturing. The potential of such information, that can be generated from the human motion tracking data, has not been well understood in the design of gestural interactions. Moreover, there is no established guideline in the analysis of such high-dimensional, massive and complex data for the design of gestural interactions. So, a proper method to explore and communicate potential knowledge of the large data collections should be provided to the interaction designers.

Automated data analysis techniques such as data mining and pattern classification can be considered to identify subsets of data having similar properties from the massive and complex motion tracking data. However, such automated analysis methods cannot fully reflect the designer's perceptual definition of gesture similarity. The designer thus needs to integrate and insert their domain knowledge of human gestures into the results of such automated data analyses.

With these needs in mind, we propose GestureAnalyzer, a visual analytics system supporting categorization and characterization of various user gestures expressed in motion tracking data. We define *gesture data* as a sequence of human poses represented by motion tracking data. If a set of similar gesture data is frequently observed from different users, we identify the data set as a *gesture pattern*. To simplify and expedite the analysis procedure, GestureAnalyzer requires a specific data format where the user gestures are individu-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SUI'14, October 4–5, 2014, Honolulu, HI, USA.
Copyright © 2014 ACM 978-1-4503-2820-3/14/10 ...\$15.00.
<http://dx.doi.org/10.1145/2659766.2659772>

ally recorded and labeled with the corresponding elicitation task. Hierarchical clustering of gesture data is implemented to identify the most frequent and similar gesture patterns. The aggregation level of gesture data is dynamically adjustable, and different numbers of gesture clusters are generated. The hierarchical tree structure provides an overview of user gestures associated with a certain task. To support comparison of gesture data, we provide *animations* and *multiple-pose visualization* of motion trends. From these visualizations, the designer is able to compare similarity among the clustered gesture data. GestureAnalyzer also provides visualization of various features of gesture data such as variations in speed, joint angles, or distance between two joints. Through these visualizations, the designer can compare user gestures in various features, then characterize the gestures identifying the most informative features that uniquely define the gestures.

In this paper, we demonstrate GestureAnalyzer focusing on the analysis of user behavior in the design of mid-air hand gestures. We validate the utility of the system in supporting the categorization of various gestures expressed by users, and the exploration of various geometric and kinematic features of the user-defined gestures. In our analysis results, we describe how this approach can be beneficial and promising in the analysis of gesture patterns and detailed motion features. Also, we will explore how the analysis results can be used in the design of gestural interactions. The contributions of this paper include: (1) rapid categorization of user gestures leveraging an interactive hierarchical clustering method, (2) visual exploration of geometric and kinematic motion features of user gestures through interactive visualization, and (3) a visual analytics tool supporting data-driven analysis and precise communication of insights from human motion data.

RELATED WORK

Our work closely relates to gesture elicitation studies, visual analysis of time-series data, and motion sensing data for interaction design. In this section, we briefly summarize prior research on these areas, and compare them to our approach.

Eliciting Gestures from Users

To better understand user behavior and enhance the intuitiveness of interactions, elicited gestures from users have been commonly incorporated in the initial phase of gesture design. In practice, elicitation studies, such as Wizard-of-Oz and Guessability studies, have been performed to observe how users actually express their gestures in various interaction scenarios: surface computing [40], mobile interactions [30], action games for children [13], and mid-air/VR environments [27, 36]. Recently, Morris et al. [23] suggest modifications of the elicitation studies to enhance the quality of resulting gestures by reducing the effect of prior experience of users in eliciting biased gestures. In the analysis of such study results, researchers manually annotate and categorize user behavior to identify a common gesture pattern for a specific task. Basically, this analysis process is time-consuming when the number of subjects increases. Also, the researchers need to define criteria for the categorization of gestures. Some prior research has proposed taxonomies of human gestures (see Wobbrock et al. [40] for a review), but

the criteria could vary with context of interactions and demographics of users. In addition, it is challenging to formalize the criteria before the researchers understand the whole gesture patterns that occurred during the user study. In most previous work, the output gesture vocabulary is reported in a descriptive manner (e.g., illustrations) and does not convey detailed motion features (e.g., variations in joint angles).

In contrast, our approach allows rapid categorization of similar gestures by applying a data mining technique to motion tracking data. Also, visualization of various features of gestures enables the researchers to compare and identify detailed aspects of gestures across users and tasks.

Visual Analytics for Temporal Data

The purpose of visual analytics is to integrate human insights and machine capabilities for improved knowledge discovery in the analysis process using interactive visualization techniques [14]. Specifically, the analysis of temporal data using such an interactive visualization approach is an important topic in various areas such as analysis of movement data, genome information, and financial data. Sensemaking based solely on the visualization techniques is insufficient for the analysis of multi-dimensional and complex time-series data. Thus, data analysis techniques such as temporal clustering (see Warren Liao [38] for a review) have been integrated with interactive visualization techniques.

In visual analytics, the density-based clustering algorithm, OPTICS [1], has been widely implemented to analyze complex and massive movement trajectories such as GPS-tracked position data [28], eye movements [26], and mouse trajectories [21]. Schreck et al. [33] propose a framework integrating interactive Self-Organizing Map (SOM) method with domain knowledge of analyst in the analysis of huge amounts of time-varying stock market data.

Hierarchical clustering methods also have been integrated with interactive visualization in many areas. Guo et al. [8] propose an interactive hierarchical clustering method enabling human-centered exploratory analysis of multivariate spatio-temporal data. Similarly, Seo and Shneiderman [34] present an interactive exploration tool for the visualization of hierarchical clustering results using dendrograms and color mosaic scatterplots. In both works, the analyst is able to control the aggregation and exploration level of clusters by changing clustering parameters. Wu et al. [42] employ the interactive clustering approach to hierarchical modeling of query interfaces for the analysis of Web data sources. Heinrich et al. [11] implement the interactive hierarchical clustering combining table-based data visualization in the analysis of massive genomic data. This approach allows the analyst to control the aggregation of rows and columns in the table.

Previous work on visual analysis of human motion tracking data are relatively few. MotionExplorer [3] is an interactive visual searching and retrieval tool for synthesis of human motion sequences. It provides an overview of sequential human poses in dendrograms and motion graphs. In these node-link models, each node represents a static human pose. This tool

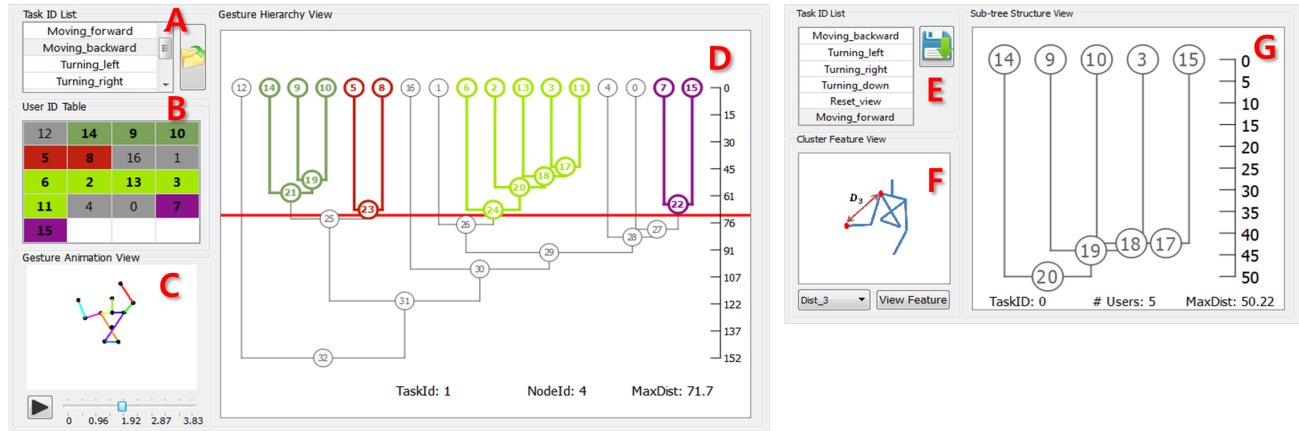


Figure 1. The GestureAnalyzer interface. (A) is a list of tasks loaded from the database. (B) shows a table of user IDs. (C) shows the animation of user gestures. (D) is a panel that shows the interactive hierarchical clustering of gesture data. Information of currently selected task and cluster node are given at the bottom. (E) is a list of output clusters generated from the interactive hierarchical clustering. (F) provides a visual definition of gesture feature. (G) shows a tree diagram of gesture clusters.

uses the Euclidean distance of raw joint coordinates to compute the similarity among human poses.

To provide an overview of motion clustering results, our approach adopts the interactive hierarchical clustering method. Since our data type is sequential human motion representing a trial of gesture, we choose to use individual tree nodes to represent a time-varying sequence of poses rather than a single human pose. Also, to better reflect the context of gestures in the analysis, our approach employs various pose similarity measures introduced by Chen et al. [5].

Motion Data for Interaction Design

Various motion sensing data has been actively used in modern interaction scenarios as an input, *user gesture*, to the system: pen-based [12], fingers/hands [4, 35], and full-body motions [7, 18]. Much research also has been done in the development of gesture authoring tools. Exemplar [10] and MAGIC [2] enable designers to rapidly create sensor-based motion gestures by demonstration. Similarly, Kim et al. [16] introduced a demonstration driven gesture authoring tool, EventHurdle, for designing multi-touch or mid-air gestures. Proton++ [17] and Gesture Coder [20] are prototyping tools for multitouch gestures by demonstration. The main focus of these methods have been on the interpretation of human motion as an input to the system.

Few work has been done in the use of motion sensing data in the understanding of user behavior for the interaction design. Wang and Lai [37] use motion tracking data to investigate the frequency and similarity of gestures used in group brainstorming with different interaction modalities (face-to-face and computer-mediated). Schrammel et al. [32] discuss that interpretation of motion tracking data can be used in the understanding of user behavior such as pattern of pedestrian attention using body and head movements. Hansen [9] discusses the potential of body motion data as a material for interaction design, and argues that visualization of body movement is a necessary step for the exploration and generation of knowledge from the motion data.

Our work also exploits motion tracking data for the analysis and understanding of user behavior. However, to support and expedite the knowledge discovery from the motion tracking data, we propose a visual analytics approach integrating interactive clustering and visualization techniques. Specifically, in this paper, we focus on the use of motion tracking data to support analysis of gesture patterns.

VISUAL ANALYTICS FOR GESTURE ELICITATION

Our goal is to study how to use visual analytics to support categorization and characterization of user-elicited gestures in motion tracking data. The benefit of applying visual analytics to this problem is that allows combining computational methods with the insights and intuition of a human analyst. Here, we discuss the design space for such a system.

Data Model

We define our data model as high-dimensional motion tracking data labeled with user and task information. We require gesture data to have distinct starting and ending poses, since extracting meaningful gestures from continuous motion tracking data is outside the scope of the system.

Analysis Tasks

The main activity in gesture elicitation studies is to categorize a large collection of recorded gesture data sets into a small set of similar gesture patterns. To support this analysis task, visual analytics software should provide an overview of the entire gesture data set as well as information about the relationship between individual gestures. This also requires the capacity to drill down into the data to see similarities, differences, and distances between gestures.

Visualization

Visual representations of the input gesture data set should aid analysts to overview the high-dimensional motion tracking data. Furthermore, being able to play back an animation of an individual gesture data is a key aspect of the visualization.

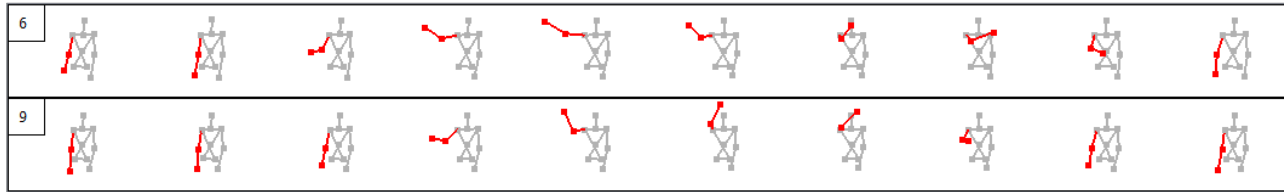


Figure 2. Multiple-pose visualizations. In this example, ten poses are extracted at regular intervals from the same gesture performed by two different users. Note that hands and elbows are detected as active joints, and indicated with red color. Corresponding user id is provided on upper-left corner.

THE GESTUREANALYZER SYSTEM

Based on the design space of visual analytics for elicitation studies, we developed GestureAnalyzer (Figure 1) to support the analysis of gesture patterns. In this section, we describe the system components of GestureAnalyzer.

Gesture Data and Feature Vectors

We define a gesture data as a sequence of human poses which are 3D positions of 11 body joints (hands, elbows, shoulders, head, neck, chest, and pelvis) as shown in Figure 1C. We obtain the human skeletal model using a Kinect camera. Khoshelham et al. [15] suggest the Kinect camera is reasonably accurate to represent indoor human activity, but there are occlusion and noise issues in human body tracking. In our gesture data recording, the users face the camera, and critical occlusion problems (i.e., lost tracking of multiple joints) therefore barely appear. To alleviate noise in the gesture data, we apply a Savitzky-Golay smoothing filter [31] to the motion tracking data. The position and orientation of body pose is normalized by shifting the chest position of each skeleton to the origin, and aligning the normal direction of the initial body pose to the camera direction.

In the early development stage, we noticed that Euclidean distance of raw joint coordinates does not provide sufficient gesture similarity. To address this issue, we adopt a geometry-based pose descriptors, proposed by Chen et al. [5]. Specifically, we use relative joint vector normalized to unit length. We extract the pairwise relative joint vector by extracting the difference between the position of joint i and j : $p_{ij} = \text{unit}(p_i - p_j)$. Because frequent motion in mid-air hand gestures appears on hands, elbows, and shoulders, we define the feature vector, $X \in \mathbb{R}^{21}$ by 7 vectors consisting of relative position of hands and elbows, elbows and shoulders, hands and shoulders, and left and right hand. This selective feature vector is effective in eliminating irrelevant joint information, and reduces the size of feature vector decreasing computing time in similarity measure.

Interface for Gesture Database Access

In the design of an interface for accessing to the gesture database, we consider a simple yet effective way to organize and represent the gesture data sets. Collected gesture data sets are readily available in the database on disk, and loaded to the system by selecting the corresponding task id in the *Task ID* list (Figure 1A). Then, a set of gesture data included in the selected task appears on *User ID* table (Figure 1B). Each gesture data is labeled with the corresponding user id. To provide an overview of similarity among user gestures, the *User ID*

table is organized according to the adjacency of gesture data in *Gesture Hierarchy* view (Figure 1D). By selecting a user id in the table, the corresponding gesture animation is played in the *Gesture Play-back* window (Figure 1C).

Interactive Hierarchical Clustering

GestureAnalyzer uses a hierarchical clustering algorithm to aggregate similar gestures into several groups. A hierarchical tree structure of a given gesture data set is constructed using an agglomerative clustering algorithm. Inherently, human gestures could have infinite variations in motion. Even if one user performs the same gesture for several times, each trial could yield different lengths of gesture. To measure similarity among the variable length of gestures, we decide to use Dynamic time warping (DTW), a distance measure for variable lengths of time-series data, due to its computational efficiency and ease of implementation [24]. The agglomerative clustering algorithm starts from multiple root nodes representing individual gesture data, and merges them into a cluster node progressively until a single node is left. In the definition of distance between clusters, we use complete-linkage method representing maximum distance among cluster members. An example of the hierarchy tree is shown (Figure 1D). The root nodes are positioned at the very top, and labeled with the corresponding user id. The cluster nodes are numbered in a merging order.

The depth of nodes is defined by the maximum distance among associated cluster members. The distance is used as a *cut-off* value to define the depth level of tree structure. To adjust the cut-off value GestureAnalyzer provides a horizontal bar (horizontal red line in Figure 1D). The cut-off value can be read at the bottom of the tree diagram and the vertical scale. The maximum distance stands for the intra-cluster similarity of gesture clusters providing a measure of how cluster members are close to each other. Information of currently selected task and cluster node is given at the bottom of the tree diagram. By changing the cut-off value, different shape of gesture clusters are generated. The generated clusters are represented with different colors as shown in Figure 1D. This coloring scheme is also applied to the *User ID* table (Figure 1B). The colors and order of the cluster nodes are intended to aid rapid and accurate visual understanding of gesture data structure. The hierarchy tree provides a global overview of gesture data set, and indicates candidate gesture clusters having strong intra-cluster similarity and frequent occurrence in the data set. More detailed information on a gesture cluster can be queried through the functions described in the following sections.

Visualization of Gesture Motion

The distance score given in the hierarchical tree view (Figure 1D) provides an immediate similarity measurement in the identification of frequent and concordant user gestures from a given data set. However, the output clusters do not necessarily reflect our perceptual model of gesture similarity. For example, we consider moving the right hand from *left to right* and *right to left* as different gestures, while the clustering algorithm would combine them into the same cluster. So, in the identification of gesture clusters, it is important to investigate what types of gestures are grouped together into a cluster, and how they are actually similar to each other. A key challenge in GestureAnalyzer is to provide an efficient yet effective way to compare the similarity among gesture data in the clusters.

To address this challenge, we provide two visualization methods to compare gesture data: animation and multiple-pose visualization. As shown in Figure 1C, we provide a window for gesture animation. By selecting a user id on the *User ID* table or in the tree diagram, the corresponding gesture data is animated in the window. Using animation of gestures makes it easy to understand motion trends and effective to compare small number of gestures. However, if the number of gesture data being compared increases, animating and replaying the entire gesture data become time-consuming. Due to limited capacity of human memory, some trends of gesture motion observed previously could be confused with new observations. Also, it is hard to judge where to start the investigation in the large data set.

Based on this observation, we provide a small-multiples visualization [29] of multiple gesture data, as a supplement to the gesture animation. As shown in Figure 2, several poses of gestures are extracted at regular intervals from the entire motion frames, and displayed from left-to-right. In Figure 2, ten frames of poses are extracted from *play next music* gesture performed by two different users.

As discussed by Ofli et al. [25], when users perform a gesture, they are not likely to move their entire body; instead there exists a set of joints that moves the most. In the multiple pose visualization, we highlight active joints having higher variance in motion than a certain threshold value. This aims to promote quick detection of active joint location, and reduce the overload of information in the multiple pose visualization.

Generating User-defined Gesture Set

Once the analyst decides an output cluster from a task data set, he/she can save it to a database. The data set can now be compared with other gesture clusters generated from different task data. A complete cluster database consists of gesture clusters generated from each task data. The database stands for a user-defined gesture vocabulary.

In the representation of gesture clusters, we use a sub-tree structure extracted from the original hierarchical tree model rather than a representative gesture data (e.g., averaged gesture motion). This representation is intended to provide more detailed information of user-defined gestures including the internal structure and intra-cluster similarity of each gesture cluster. Figure 1G shows an example of an user-defined ges-

ture expressed in the sub-tree structure. Metadata of the user-defined gesture including motion tracking data, task id, the number of associated users, and maximum similarity score is provided along with the tree structure.

Visualization of Gesture Features

Once a vocabulary of user-defined gestures is generated, GestureAnalyzer provides visualization of various geometric and kinematic features such as joint angles, distance between joints, speed, and relative joint positions to explore various aspects of gestures. This exploratory visualization aims to support rapid detection of the most static and dynamic gesture features across users and tasks.

Saving Gesture Clusters to Database

As a result of the analysis, GestureAnalyzer generates (1) a set of sub-tree structures representing gesture clusters, (2) metadata of the gesture clusters including motion tracking data, task id, the number of associated users, and maximum distance among cluster members, and (3) the visual representation of various gesture features. This collection of data can be saved to the database on disk. Later, specific information can be queried from the database to compose a material for reporting results and findings from the analysis. Also, the database can be a good source in the design of user-informed gesture classifiers.

CASE STUDY AND RESULTS

To demonstrate the utility of GestureAnalyzer, we conducted a case study on gesture data sets obtained from elicitation studies. Here, we briefly explain the procedure of the gesture elicitation studies, and discuss initial results from the analysis of the gesture data sets using GestureAnalyzer.

Gesture Elicitation Studies

We conducted user elicited gesture studies for designing mid-air hand gestures. The studies and tasks were designed based on the procedure described in prior work [27, 36, 40].

Tasks

Our selection of gesture design tasks was intended to explore the design of mid-air hand gestures for the navigation in 3D space and the system control. In the elicitation studies, there were two gesture design scenarios: (1) camera view control in 3D space and (2) music player control. In the first session of study, the participants were asked to design mid-air

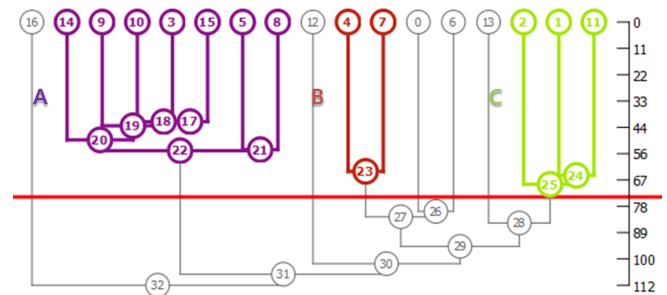


Figure 3. Three gesture clusters, A, B, and C, are generated from the interactive hierarchical clustering.

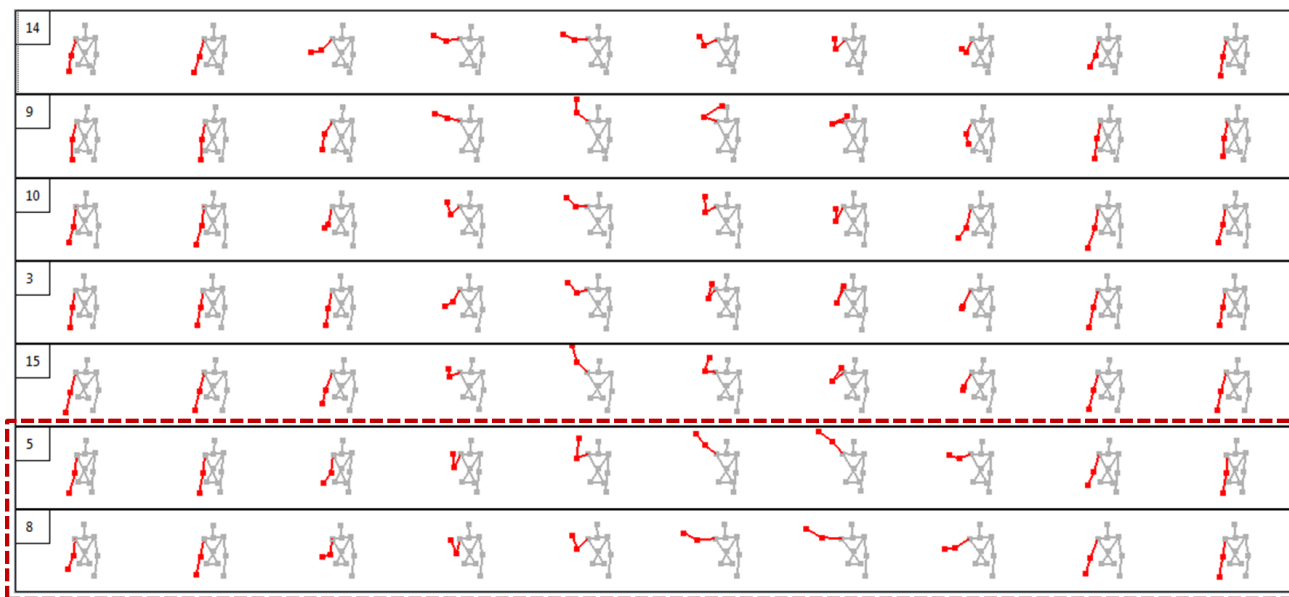


Figure 4. An overview of motion trends in the candidate gesture cluster for *camera moving forward* task. The gesture data within a red box (the gesture data labeled by 5 and 8) have different right hand motion from other gestures.

hand gestures to trigger seven different tasks for camera view control: camera moving forward/backward, turning left/right, turning up/down, and reset the camera view to the initial state (go back to the origin). In the design of gestures for the music player control, six tasks were given to the participants including start(pause)/stop music, previous/next track, and increase/decrease volume.

Participants

Seventeen participants were recruited for the study ranging in age from 21 to 31 years. All of them were male and university students. One of the participants was left-handed, and majority of the participants have used gesture-based interaction devices such as smart phones, tablet PCs, and gaming devices (Nintendo Wii and Xbox Kinect).

Apparatus

The gesture design space was physically defined as 3 x 3 x 5 m to ensure enough space for the participants. A Microsoft Kinect camera and OpenNI SDK was used to extract skeletal model of users. A laptop was connected to a large display equipment to indicate a referent of each task to the participants in the gesture design.

Gesture Data Recording

At the beginning of the study, the participants were briefly introduced to two study scenarios and the type of tasks to be performed. In the preliminary analysis of gesture data, it was noticed that the starting and ending pose of gestures significantly affect the similarity measure of gesture data, even if the intermediate gesture motions are quite similar. Based on this observation, we asked the participants to start and end gestures with natural standing pose. Also, since our current system aims to analyze the body motion tracking data, the participants were asked to use hands and arms rather than fingers in the gesture design.

In the first session of the study to design the camera view control gestures, a 3D model of building was presented in the large screen to show the effect of camera movements. Such an effect presenting the result of gesture is called a *referent*. For each task, an animation showing the referent was presented to the participants. No interface was presented in the design of music player control gestures. The referent was given to the participants while an experimenter controls the music player on computer. At each study session, the referent was presented in random order.

Results

In this section, we demonstrate GestureAnalyzer for the analysis of gesture data sets, and discuss the results. The first phase of analysis is to identify the most frequent and similar gesture patterns from the data sets. This demonstrates how the system supports generation of user-defined gesture vocabularies. Then, the system provides visualization of motion features of gesture groups. From this visual analysis, we are able to compare various features of gestures across the users and tasks. Through the comparison of gesture features, we can identify the most informative features of user-defined gestures.

Categorization of User Expressed Gestures

After importing a task data set into GestureAnalyzer, a hierarchical tree diagram is immediately presented in the *Gesture Hierarchy View* (Figure 1D). This view provides an overview of gesture data structure, and helps to identify candidate gesture clusters from the entire data set. In the rest of this section, we will explain the analysis procedure through an example task data set of the *camera moving forward*.

Identifying candidate gesture cluster: From the data set of *camera moving forward* task, three gesture clusters are generated by dragging the cut-off bar to around 72 value of max-

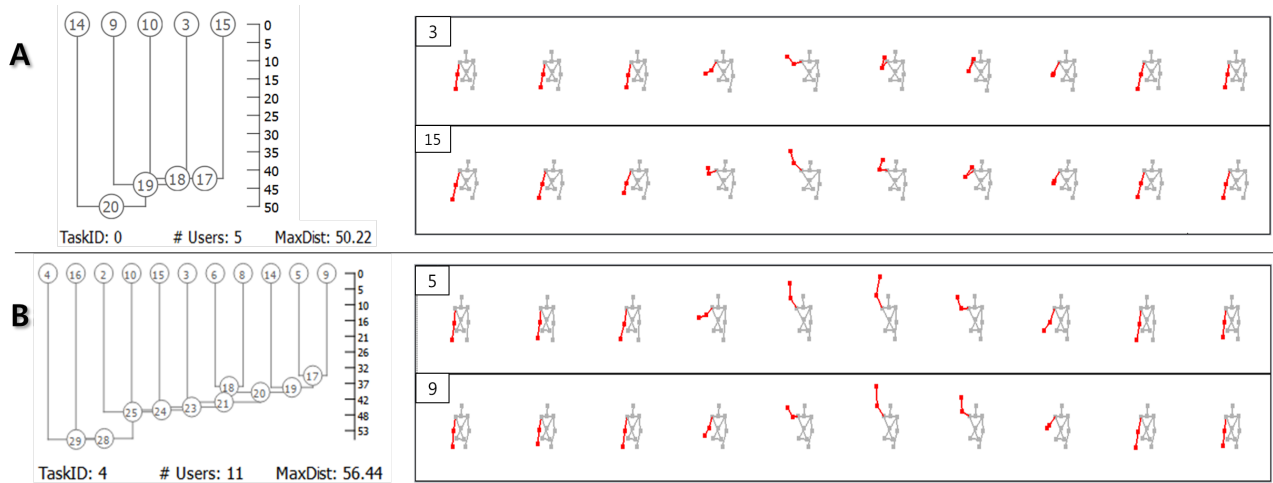


Figure 5. Gesture tree diagrams (left) and multiple-pose visualizations (right). The gesture cluster having the smallest MaxDist is presented in the multiple-pose visualizations. A: user-defined *camera moving forward* gesture. B: user-defined *increase volume* gesture.

imum similarity (Figure 3). From the tree diagram, we notice that the gesture cluster (A) includes larger number of gesture data at the root level than the other candidate clusters. So, the cluster (A) is selected as a candidate gesture group as a user-defined gesture for the imported task data.

Visual investigation of candidate cluster: Detailed investigation of the candidate cluster (A), in Figure 3, is performed through a combination of the sub-tree structure of cluster (A), the multiple-pose visualization, and the gesture animation. In the tree diagram, the order of node label reveals several cluster nodes where detailed investigation of gesture similarity would be required. For example, gesture data labeled by (5), (8) and (14) are merged into the cluster (A) at the last course of clustering with the highest distance value. This indicates that these gesture data could be outliers in the candidate cluster. An overview of motion trends in the candidate cluster is provided by the multiple-pose visualization in Figure 4. The multiple display reveals that the gesture data (5) and (8) show different movement of right arm. The animation of these data confirms that (5) and (8) gestures are moving right arm away from the body, and the other gestures show moving right arm close to the body. So, the outlier gesture data (5) and (8) should be removed from the candidate gesture cluster (A). We decide the output gesture cluster from the *camera moving forward* data set as the cluster node (20), and its corresponding sub-tree structure is shown in Figure 5A.

Generating user-defined gestures: Applying the analysis procedure to the entire task data sets, we were able to generate user-defined gestures represented by multiple poses and tree diagram. Multiple-poses visualizations and quality measures of the complete resulting gesture sets are provided in an appendix. Figure 5 shows examples of output tree diagrams representing the user-defined gestures for *camera moving forward* and *increase volume* tasks. The tree diagrams provide an overview of internal structure and similarity of gesture clusters. Metadata for each tree was stored in a gesture database on disk.

Characterization of Gesture Vocabularies

Once the user-defined gestures are extracted from the task data sets and stored in the database, geometric and kinematic features of each gesture are visualized in the gesture characterization phase. Figure 1F provides a visual definition of gesture features. We normalize the length of gesture data to 100 sec to aid comparison of different length of gestures. In this section, we use user-defined *increase volume* and *decrease volume* gestures to show how gesture features are used in the characterization of gestures.

Comparison across users in the same task: Figure 6 shows four features of user-defined *increase volume* gesture. From the feature visualization, we are able to notice that the joint angle of left elbow (A) and the distance between left hand and left shoulder (B) are static features, while the joint angle of right elbow (C) and the distance between hands (D) show relatively dynamic variations. Internal similarity of gesture features can be investigated from the feature visualization. Feature (D) shows more congruent shape of transition, while feature (C) has wide range of variations in the shape. So, we can consider feature (D) as a representative feature of *increase volume* gesture having similar behavior across users.

Comparison across the tasks: The feature visualization supports comparison of user-defined gestures across different tasks. For example, *increase volume* (Figure 6) and *decrease volume* gestures (Figure 7) show similar shape of gesture feature graphs for the right elbow joint angle and the distance between hands. However, two gestures have different trends in the timing of maximum and minimum point. The *decrease volume* gesture shows early occurrence of the maximum and minimum points. The designers would consider this finding in the design of gesture classifier by using these features.

DISCUSSION AND LIMITATIONS

The utility of GestureAnalyzer is demonstrated in a case study where gesture data of 17 users are obtained from elicitation studies. Results from the case study provide a set of user-defined gestures represented by tree diagrams and mul-

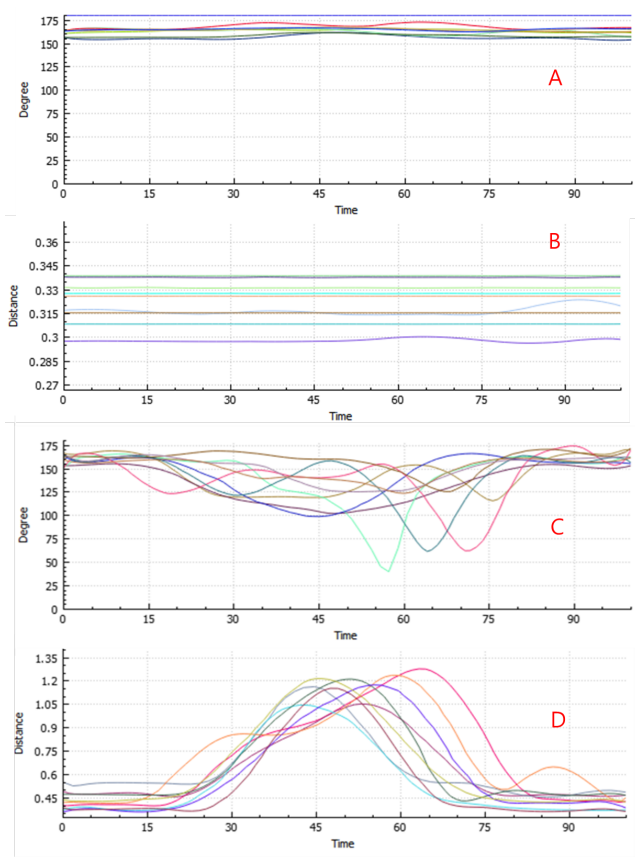


Figure 6. Visualization of geometric features of user-defined *increase volume* gestures of 17 users. Each user gesture is indicated with different color. (A: left elbow joint angle, B: distance between left hand and left shoulder, C: right elbow joint angle, D: distance between hands).

multiple poses. Also, the visualization of gesture features reveals potential knowledge of gesture data such as static/dynamic feature behavior, unique characteristics of gestures, and internal similarity of gestures.

Our system enables rapid identification of frequent gesture patterns from a large data set without having prior knowledge of the data. An interactive hierarchical clustering plays a key role in the decision of output cluster structure. A combination of sub-tree diagram, multiple-pose display, and gesture animation aids to identify a candidate gesture cluster from the whole tree diagram. In our case study, a few mouse clicks was required to identify a frequent gesture pattern.

The gesture patterns are represented using tree structure along with metadata including the maximum distance among cluster members and the frequency of gesture patterns. The tree structure is useful to understand the relationship of cluster members such as how they are similar to each other and in what order they are clustered. The information of clustering order is helpful for detecting outliers in the interactive clustering process. The last clustered data has the highest potential to be an outlier of the cluster. GestureAnalyzer enables to generate such detailed information of the gestures that cannot be provided by traditional analysis.

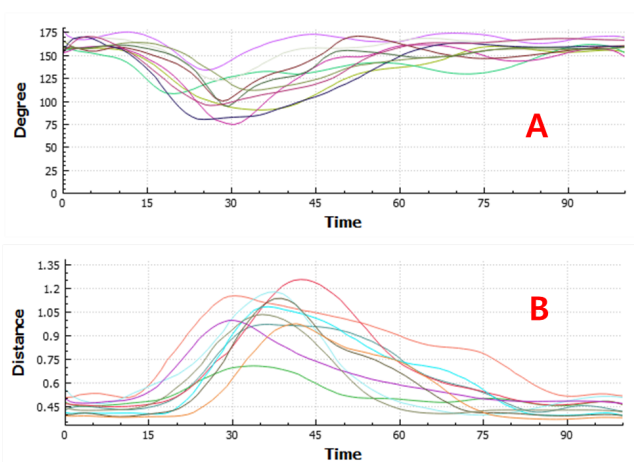


Figure 7. Visualization of geometric features of user-defined *decrease volume* gestures of 17 users. Each user gesture is indicated with different color. (A: right elbow joint angle, B: distance between two hands)

The findings and results generated from GestureAnalyzer can be used as supplementary materials in communicating discovered knowledge of user gestures. For example, detailed information of user behavior such as duration of static posture in a gesture pattern and intermediate motion between successive gestures can be further queried using the result data. Our system also can be used in usability research. Exploratory analysis of gesture patterns could be used in identifying usability problems that frequently occur in the gestural interactions. Then, these problems can be reported and communicated in detail using the data-driven analysis results. The analysis results also can be used in the design of gesture classifier reflecting user behavior.

There are some limitations in the current approach of GestureAnalyzer. In the multiple-pose visualization, reading the entire array of multiple-pose could take a long time when a large number of gesture data is displayed. Applying a color scheme to the multiple display to represent similar poses with the same color could be a strategy to alleviate the issue by enhancing the readability of pose connectivity. Alternatively, we could use the motion graph [19] in the visualization of gesture motions. Our approach is also limited to a specific gesture data format where the gestures start and end with a natural standing pose. To apply our system to broader contexts of interaction analysis, we should consider segmentation and annotation of the whole user study data composed of various natural human motions. We leave it to future work to integrate segmentation of motion data into GestureAnalyzer for detecting the starting and ending pose of the gestures. This extension of analytics capability could be useful in the analysis of natural human poses indicating gesture intent.

From our initial evaluation of GestureAnalyzer, we could imagine several possible extensions to other research areas. Especially, research area involving human behavior analysis could provide a number of interesting opportunities for extension of our research. In collaborative design, our approach could be used to analyze the relationship between natural human motions and collaborative design qualities. Our

approach also could support the analysis of learning process providing new insights from the motion tracing data. Worsley and Blikstein [41] discuss the possibility of hand gesture data in the analysis of expertness in object manipulation. While this work is limited to use only the cumulative displacement of two hands in the analysis, our approach could be used to explore human motion supported by interactive visualization and data processing techniques.

CONCLUSION AND FUTURE WORK

We have described the design and demonstration of GestureAnalyzer, a visual analytics system supporting identification and characterization of gesture patterns from motion tracking data. We implemented an interactive hierarchical clustering method to identify the most frequent gesture pattern without any prior knowledge of the data. Also, the visual exploration of gesture features enabled comparison of the various aspects of gestures, and supported identification of representative gesture features. A case study on motion tracking data obtained from elicitation studies is conducted to demonstrate the utility of GestureAnalyzer. In our future work, we plan to further investigate the usability of the system via expert reviews, and compare it with traditional analysis methods. We will also expand the analytics capability to other motion data types such as finer finger motions or sensor-based motion tracking data. The knowledge discovered from the motion tracking data can be directly used in training gesture classifiers. Integrating gesture classification into GestureAnalyzer will provide a way to understand the analysis results and use it toward interaction design.

ACKNOWLEDGMENTS

This work was partially supported by the NSF Award No. 1235232 from CMMI and 1329979 from CPS, as well as the Donald W. Feddersen Chaired Professorship from Purdue School of Mechanical Engineering. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

REFERENCES

1. Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. Optics: ordering points to identify the clustering structure. *ACM SIGMOD Record* 28, 2 (1999), 49–60.
2. Ashbrook, D., and Starner, T. MAGIC: a motion gesture design tool. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2010), 2159–2168.
3. Bernard, J., Wilhelm, N., Kruger, B., May, T., Schreck, T., and Kohlhammer, J. MotionExplorer: Exploratory search in human motion capture data based on hierarchical aggregation. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2257–2266.
4. Buchmann, V., Violich, S., Billingham, M., and Cockburn, A. FingARtips: gesture based direct manipulation in augmented reality. In *Proceedings of the ACM Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia* (2004), 212–221.
5. Chen, C., Zhuang, Y., Nie, F., Yang, Y., Wu, F., and Xiao, J. Learning a 3D human pose distance metric from geometric pose descriptor. *IEEE Transactions on Visualization and Computer Graphics* 17, 11 (2011), 1676–1689.
6. Dahlbäck, N., Jönsson, A., and Ahrenberg, L. Wizard of Oz studies—why and how. *Knowledge-Based Systems* 6, 4 (1993), 258–266.
7. Gerling, K., Livingston, I., Nacke, L., and Mandryk, R. Full-body motion-based game interaction for older adults. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2012), 1873–1882.
8. Guo, D., Peuquet, D., and Gahegan, M. Opening the black box: interactive hierarchical clustering for multivariate spatial patterns. In *Proceedings of the ACM Symposium on Advances in Geographic Information Systems* (2002), 131–136.
9. Hansen, L. A. Full-body movement as material for interaction design. *Digital Creativity* 22, 4 (2011), 247–262.
10. Hartmann, B., Abdulla, L., Mittal, M., and Klemmer, S. R. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2007), 145–154.
11. Heinrich, J., Vehlow, C., Battke, F., Jäger, G., Weiskopf, D., and Nieselt, K. iHAT: interactive hierarchical aggregation table for genetic association data. *BMC Bioinformatics* 13, Suppl 8 (2012), S2.
12. Hinckley, K., Ramos, G., Guimbretiere, F., Baudisch, P., and Smith, M. Stitching: pen gestures that span multiple displays. In *Proceedings of the ACM Conference on Advanced Visual Interfaces* (2004), 23–31.
13. Höysniemi, J., Hämäläinen, P., and Turkki, L. Wizard of Oz prototyping of computer vision based action games for children. In *Proceedings of the ACM Conference on Interaction Design and Children* (2004), 27–34.
14. Keim, D., Andrienko, G., Fekete, J.-D., Görg, C., Kohlhammer, J., and Melançon, G. Visual analytics: Definition, process, and challenges. In *Information Visualization, A. Kerren, J. Stasko, J.-D. Fekete, and C. North, Eds., vol. 4950 of Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2008, 154–175.
15. Khoshelham, K., and Elberink, S. O. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors* 12, 2 (2012), 1437–1454.
16. Kim, J.-W., and Nam, T.-J. EventHurdle: supporting designers' exploratory interaction prototyping with gesture-based sensors. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2013), 267–276.

17. Kin, K., Hartmann, B., DeRose, T., and Agrawala, M. Proton++: a customizable declarative multitouch framework. In *Proceedings of the ACM Symposium on User Interface Software and Technology* (2012), 477–486.
18. Konrad, T., Demirdjian, D., and Darrell, T. Gesture+ play: full-body interaction for virtual environments. In *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems* (2003), 620–621.
19. Kovar, L., Gleicher, M., and Pighin, F. Motion graphs. *ACM Transactions on Graphics* 21, 3 (2002), 473–482.
20. Lü, H., and Li, Y. Gesture coder: a tool for programming multi-touch gestures by demonstration. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2012), 2875–2884.
21. McArdle, G., Tahir, A., and Bertolotto, M. Spatio-temporal clustering of movement data: An application to trajectories generated by human-computer interaction. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences I-2* (2012), 147–152.
22. Mitra, S., and Acharya, T. Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics* 37, 3 (2007), 311–324.
23. Morris, M. R., Danielescu, A., Drucker, S., Fisher, D., Lee, B., Wobbrock, J. O., et al. Reducing legacy bias in gesture elicitation studies. *Interactions* 21, 3 (2014), 40–45.
24. Needleman, S. B., and Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology* 48, 3 (1970), 443–453.
25. Offli, F., Chaudhry, R., Kurillo, G., Vidal, R., and Bajcsy, R. Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition. *Journal of Visual Communication and Image Representation* 25, 1 (2014), 24–38.
26. Ooms, K., Andrienko, G., Andrienko, N., De Maeyer, P., and Fack, V. Analysing the spatial dimension of eye movement data using a visual analytic approach. *Expert Systems with Applications* 39, 1 (2012), 1324–1332.
27. Piumsomboon, T., Clark, A., Billingham, M., and Cockburn, A. User-defined gestures for augmented reality. In *Human-Computer Interaction—INTERACT*. Springer, 2013, 282–299.
28. Rinzivillo, S., Pedreschi, D., Nanni, M., Giannotti, F., Andrienko, N., and Andrienko, G. Visually driven analysis of movement data by progressive clustering. *Information Visualization* 7, 3-4 (2008), 225–239.
29. Robertson, G., Fernandez, R., Fisher, D., Lee, B., and Stasko, J. Effectiveness of animation in trend visualization. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1325–1332.
30. Ruiz, J., Li, Y., and Lank, E. User-defined motion gestures for mobile interaction. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2011), 197–206.
31. Savitzky, A., and Golay, M. J. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry* 36, 8 (1964), 1627–1639.
32. Schrammel, J., Paletta, L., and Tscheligi, M. Exploring the possibilities of body motion data for human computer interaction research. In *HCI in Work and Learning, Life and Leisure*. Springer, 2010, 305–317.
33. Schreck, T., Tekušová, T., Kohlhammer, J., and Fellner, D. Trajectory-based visual analysis of large financial time series data. *ACM SIGKDD Explorations Newsletter* 9, 2 (2007), 30–37.
34. Seo, J., and Shneiderman, B. Interactively exploring hierarchical clustering results [gene identification]. *IEEE Computer* 35, 7 (2002), 80–86.
35. Takeoka, Y., Miyaki, T., and Rekimoto, J. Z-touch: an infrastructure for 3D gesture interaction in the proximity of tabletop surfaces. In *Proceedings of the ACM Conference on Interactive Tabletops and Surfaces* (2010), 91–94.
36. Vatavu, R.-D. User-defined gestures for free-hand TV control. In *Proceedings of the European Conference on Interactive TV and Video* (2012), 45–48.
37. Wang, H.-C., and Lai, C.-T. Kinect-taped communication: using motion sensing to study gesture use and similarity in face-to-face and computer-mediated brainstorming. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2014), 3205–3214.
38. Warren Liao, T. Clustering of time series data—a survey. *Pattern Recognition* 38, 11 (2005), 1857–1874.
39. Wobbrock, J. O., Aung, H. H., Rothrock, B., and Myers, B. A. Maximizing the guessability of symbolic input. In *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems* (2005), 1869–1872.
40. Wobbrock, J. O., Morris, M. R., and Wilson, A. D. User-defined gestures for surface computing. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (2009), 1083–1092.
41. Worsley, M., and Blikstein, P. Towards the development of multimodal action based assessment. In *Proceedings of the ACM Conference on Learning Analytics and Knowledge* (2013), 94–101.
42. Wu, W., Yu, C., Doan, A., and Meng, W. An interactive clustering-based approach to integrating source query interfaces on the deep web. In *Proceedings of the ACM SIGMOD Conference on Management of Data* (2004), 95–106.