Moving Toward an Ecologically Valid Data Collection Protocol for 2D Gestures In Video Games

Eugene M. Taranta II
ICE Research Cluster
University of Central Florida
Orlando, FL 32816, USA
etaranta@gmail.com

Corey R. Pittman
Jack P. Oakley
Mykola Maslych
ICE Research Cluster
University of Central Florida
Orlando, FL 32816, USA
{cpittman, jack.p.oakley,
maslychm}@knights.ucf.edu

Mehran Maghoumi
Joseph J. LaViola Jr.
ICE Research Cluster
University of Central Florida
Orlando, FL 32816, USA
{mehran, jjl}@cs.ucf.edu

ABSTRACT

Those who design gesture recognizers and user interfaces often use data collection applications that enable users to comfortably produce gesture training samples. In contrast, games present unique contexts that impact cognitive load and have the potential to elicit rapid gesticulations as players react to dynamic conditions, which can result in high gesture form variability. However, the extent to which these gestures differ is presently unknown. To this end, we developed two games with unique mechanics, Follow the Leader (FTL) and Sleepy Town, as well as a standard data collection application. We collected gesture samples from 18 participants across all conditions for gestures of varying complexity, and through an analysis using relative, global, and distribution coverage measures, we confirm significant differences between conditions. We discuss the implications of our findings, and show that our FTL design is closer to being an ecologically valid data collection protocol with low implementation complexity.

Author Keywords

Gestures, Games, Follow the Leader, Ecologically Validity

CCS Concepts

•**Human-centered computing** → **Human computer interaction (HCI); Gestural input;** User studies;

INTRODUCTION

Gestures remain a popular way of interacting with computers at the user interface boundary, thereby motivating researchers to continuously advance the state-of-the-art in pattern matching and gesture set design [15, 29, 34]. To this end, practitioners often collect test samples from participants using data collection techniques that allow one to focus solely on gesticulation and form [6, 26, 35, 36]. Such data is useful, enabling one to make relative comparisons between techniques and demonstrate improvements. However, it is unlikely that gestures produced in this way will capture variabilities present in form when one interacts with an application directly, making it difficult for researchers to understand how their efforts will translate into practice.

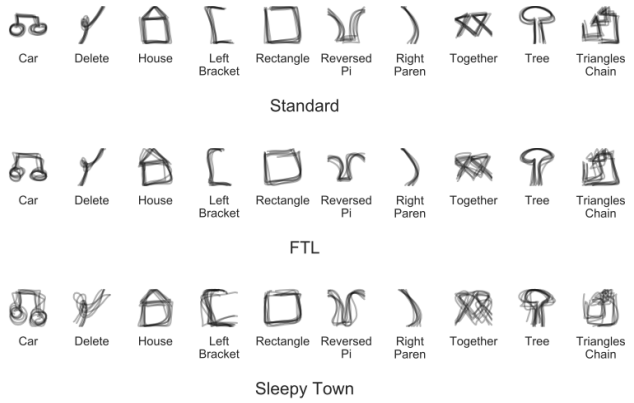

Figure 1: Gesture distributions elicited by three unique data collection applications from a single participant in our user study. Notice how form and varaibility differ signifcantly across the three conditions.

Video games are one domain where such differences likely impact testing and experimentation. Prior research has shown that in-game gesture recognition accuracy is less than that of non-game data collected for training and testing within the same application [3, 27, 28]. Reasons for this drop are presently unknown, but one reasonable assumption is that gesture production variability increases with interaction complexity. For example, players may interact directly with virtual objects, and game-specific interaction requirements may alter gesture speed, size, orientation, and overall form. Understanding these differences will help inform practitioners on how to approach pattern recognition and user interface design for gestures.

CHI '20, April 25–30, 2020, Honolulu, HI, USA.

ACM ISBN 978-1-4503-6708-0/20/04 ...$15.00.
http://dx.doi.org/10.1145/3313831.3376417

Toward this end, we developed three data collection applications for stylus input on an interactive display. These include an application that implements a standard data collection protocol, a simple game called *Follow the Leader (FTL)*, and a complex game called *Sleepy Town*. FTL introduces a trivial game play mechanic and is designed specifically for low implementation effort, enabling others to quickly adopt a new data collection protocol into their own work that increases gesture variability. Finally, Sleepy Town presents a top-down city view and allows for navigation as well as gesture interaction with non-playable characters. Through a within subjects experiment involving 18 participants, we quantify differences in gesture production between all conditions, confirming there does exist significant differences. We discuss our findings and their implications. Specifically, we contribute (1) a demonstration that commonly employed data collection practices inadequately capture gesture production variability relative to that found within practical applications generally and games specifically; and (2) an easy to implement protocol called *Follow the Leader* that increases gesture production variability, yet leverages technology already present in most data collection applications.

RELATED WORK

To ground our discussion, the relationship between two variables such as that between time and position is a *signal* encoding information about an underlying process [23]. In the context of a human computer interaction, this underlying process relates to an intentional communication designed to invoke a specific software function; and when this communication is in the form of a *gesture*, the process generates a well known neuromuscular response linked to the command that one expects their system to recognize. That is, one first decides to interact with software in accordance with a user-specific objective, *e.g.*, to make their avatar backflip in a video game. He or she then forms an *action plan*, a sequence of virtual targets connected by neuromuscular commands that in aggregate form a complex trajectory. To model this behavior Plamondon introduced the *Kinematic Theory of rapid human movements* [18, 19], which proposes that an action plan can be fully described by a Sigma-Lognormal ($\Sigma\Lambda$) model [20]—a vectorial summation of individual lognormal primitives connecting the virtual targets. Each primitive comprising parameters related to activation time, duration, velocity, and angular position is processed by the motor cortex, which in turn activates the appropriate neuromuscular networks needed to generate the desired motion. Since the motor cortex works collaboratively with other mental processes, we should expect that a variety of factors influence human motion.

Variability In Human Motion

Although a prototypical action plan conforms to a standard, its effectuation will fall short of perfection due to fluctuations in physical and emotional state, cognitive load, ability, and other environmental factors. For example, Martín-Albo *et al.* found that each lognormal parameter follows a different random distribution for handwritten words [16], and by perturbing $\Sigma\Lambda$ model parameters according to the identified distributions, they were able to generate realistic synthetic variations. Leiva *et al.* [10] use the same parameter set to generate realistic synthetic gestures. However, it was found that new parameter distributions were needed to synthesize gestures produced by those with low vision [11] because of differences in gesture production variability.

Similarly, across populations, Vatavu *et al.* [31] as well as Hernandez-Ortega *et al.* [8] were able to differentiate between children and adult users based on differences in touch interactions. With respect to input devices, Taranta *et al.* [25] observed that the same gesture shapes collected across different device types produced differences in variability. Similarly, it was found that gesticulation speed impacts form, where unnatural speeds related to increased variability [33].

Emotional state also plays an important role. Luria *et al.* [14] found that dis-automatization of fast and accurate spatial control manifests from physical and mental stress as increased handwriting variability in the form of velocity, movement duration, and writing size. Similarly, Likforman-Sulem *et al.* [12] showed that stroke variability occurs from different emotional states within a single person. Stress, anxiety, and depression are particular culprits that cause variability to a degree that the emotional state of the participant can be classified based on gesture stroke data. With respect to familiarity, Cao *et al.* [2] found that participants over time cut corners when stroking learned forms. Gesture form can also change temporally; Liu *et al.* [13] discovered that retraining a custom gesture recognizer between sessions with recent samples improved its performance.

In the context of gesture recognition, it has been repeatedly observed that in-game accuracy drops relative to offline testing [3, 27, 28]. A common theme among these works is that participants interact with a virtual environment, whether directly or indirectly. For instance, ParЖorɒr [27] allows players to combat enemies or navigate through an urban environment using spatially relevant gestures. Similarly, Lemarchand's Prototype [28] encourages players to gesture with a stylus over incoming enemy zombie arms while manipulating a mechanical device through touch. We believe that interaction with virtual objects is also an important source of variability as players must simultaneously attend to multiple stimuli. Cognitive load may be another factor, where divided attention over multiple tasks may impact care in gesture production. To our knowledge, no effort has been made to understand differences between gestures collected for training and those sampled from a game environment. While we do not work to identify specific factors that cause variability in video games, we show that different application designs result in unique levels of gesture variability.

Data Collection

When it comes to data collection, researchers have traditionally chosen to design a sample-centric environment to collect gesture samples: isolated samples are shown to users and a canonical correct way of performing them is demonstrated. Users are then expected to mimic those canonical forms a few times during a practice round and upon demonstrating acceptable gesticulation, the actual data collection starts. Examples include the UT-Kinect dataset [37] where 10 partici-

Figure 2: Standard data collection application: Left, user draws a gesture as requested above, i.e.,"reversed-pi." Right, two buttons appear that allow user to save or delete and retry.

pants were asked to perform 10 different gesture two times, and NTU RGB+D dataset [22], one of the largest gesture datasets, which contains over 56000 single- and multi-actor samples.

Such collection procedures have some caveats. Most importantly, users tend to focus on the correct gesticulation itself, without paying much attention to the gesture's context, or how that gesture can be interleaved with other interactions. Also, collected gestures tend to be pre-segmented, making them unusable in experiments that require continuous data. Another issue that can arise in such collection settings is user frustration and fatigue due to displeasure with particular gesticulations [26]. While working with public datasets, instances have been observed where users were confused by the directions given to them, resulting in wrong gesticulation and sample labeling. Most notable is an example of the UT-Kinect dataset [37] where a participant is asked to perform a "carry" gesture, but they mistakenly performed a different gesture, leaving a mislabeled example in the dataset [15]. In this work, we are not concerned with data collection errors, but rather how the data collection application and protocol impact gesture variability.

DATA COLLECTION APPLICATIONS

We designed three data collection applications, one that replicates common practice and two that employ gestures in a game environment. We further developed all applications with Unity version 2019.1.11f1, a popular game engine designed for 2D, 3D, VR, and AR experiences. Each of these applications are described in this section.

Standard Data Collection

Our standard data collection application, shown in Figure 2, asks users to draw gestures that our software specifies one at a time. We horizontally center requests at the display's top, and users may gesticulate anywhere on the display. After one produces a gesture sample, two buttons appear that allow one to save or delete their sample. If satisfied, one can save their result and continue onto the next request, or delete their sample and produce a new variant. Gesture requests are purely random, as we provide no guard against consecutive identical requests. With this design, data collection is relatively stress free and comfortable. We impose no time pressure, feedback, or expectations so that users remain in full control.

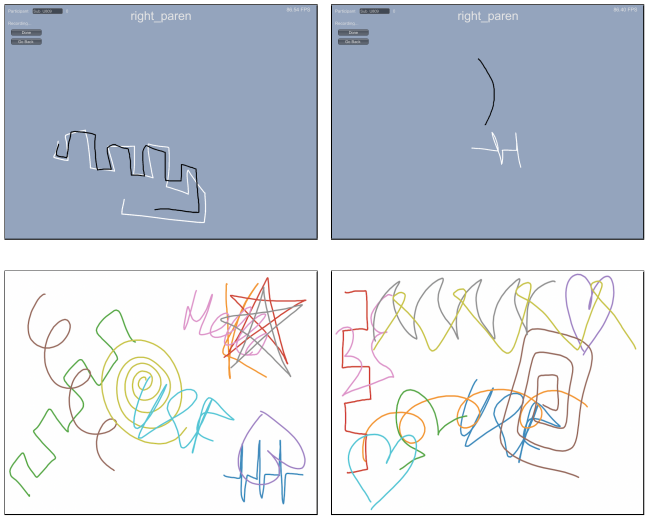

Figure 3: Follow the Leader (FTL): Upper left, a leader (white) renders a random trajectory that the player (black) must replicate in realtime as closely as possible. At random intervals FTL makes a gesture request to which one must immediately attend as the leader continues on without pause (upper right), and to which one must return upon gesture completion. The lower panels each show ten randomly selected leader trajectories to illustrate type and variety.

Game: Follow the Leader (FTL)

Our first game is named after and inspired by a popular children's game called *Follow the Leader* (FTL), in which participants line up behind a leader whom all must follow and whose random actions they must exactly replicate. In a similar way, as presented in Figure 3, we present the leader as a series of seemingly random trajectories on one's display. A player must try to replicate precisely what they see, keeping pace with their leader in time and space. At random intervals, we further present a gesture request, whereby players must stop following, gesticulate anywhere on the screen, and return to following as quickly as possible. The intention of this design is to create a sense of urgency not present in standard data collection protocols.

Leader trajectories are prerecorded samples produced by the developer using our standard data collection application described above. All leader samples are intentionally arbitrary[1], being a random collection of geometric shapes, words, and scribbles (see Figure 3 for some examples). During game play, FTL randomly draws a single leader sample and replays its trajectory at a rate matching its recorded speed. Once the sample is fully rendered, we repeat this process, continuously, until a predetermined number of gesture requests are made and satisfied.

Design Considerations

FTL is designed to be a middle ground between standard data collection and a fully featured video game like those described in the Related Work section. We expect an increase in

[1]We attend to the intentionally arbitrary nature of our design later in the discussion.

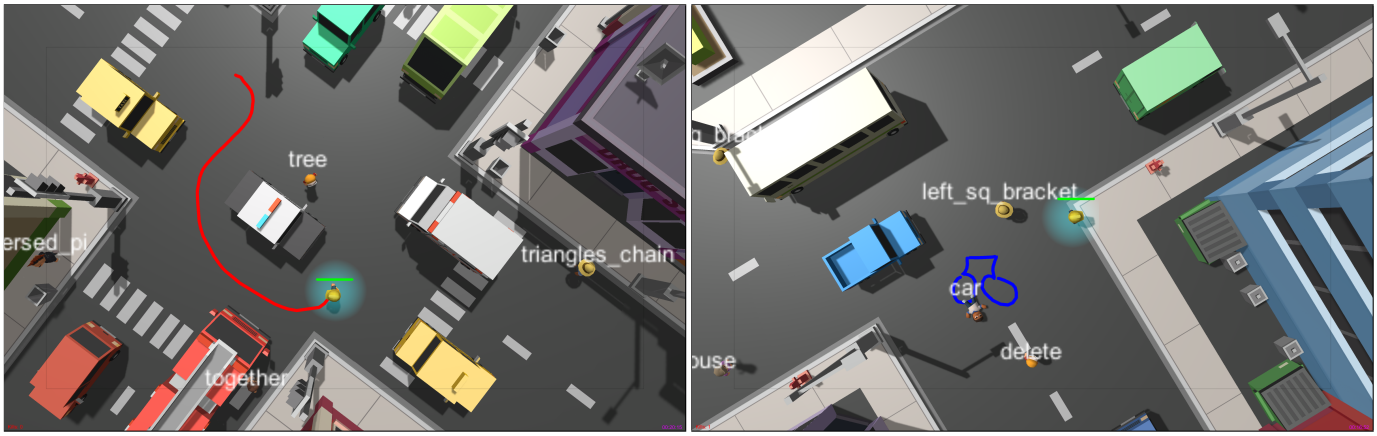

Figure 4: Sleepy Town: Left, player sketches a path that Leopold follows through the scene. Right, player draws a car gesture that causes an NPC to fall asleep as he crosses over its threshold.

gesture production variability as a player must attend to multiple dynamic tasks. FTL is further motivated by our desire to minimize developer effort should one decide to use an FTL approach in their own data collection work. Specifically, our goal was to increase variability by leveraging machinery already present in most data collection tools such as to display, capture, and randomize gestures. To use FTL, one will first define a gesture dataset, render leader sequences in random order, ask participants to follow leader actions as closely as possible, periodically make gesture request that participants immediately perform, and continue as such until all training data is collected.

Game: Sleepy Town

Our second game, Sleepy Town, offers an entirely unique experience relative to FTL by introducing comparatively greater game play complexity. In this fantasy-based game, the local government has decreed that its citizens must adopt an Uberman sleep schedule[2] so as to become a more productive town. One week after this law has gone into effect, Leopold happens across the city, where he finds that its citizens appear to be wandering aimlessly about without purpose. Further, in a state of sleep deprived delirium (because there could be no other reason), its citizens pursue and attack outsiders on sight. Luckily, Leopold is a magician who knows a variety of sleeping spells.

In Figure 4, we present the player's top-down view of a simple city scene along with an illustration of its two game play mechanics. First, to navigate through Sleepy Town, one sketches a path rendered in red that begins from within Leopold's halo. He will immediately begin to run along the player's sketched route until he reaches the end or until the player constructs a new route. As one approaches their display's physical boundary, we automatically rotate the camera around their point of contact in a way that allows him or her to continue sketching a continuous path; and when not routing, the camera automatically follows Leopold as he traverses his assigned route. When one instead sketches outside of his halo, our game transitions to gesture mode. Specifically, we project the player's input into the environment, rendering blue spell strokes. Above each non-player character (NPC) is written a gesture name—the spell that puts him or her to sleep. Once drawn, if an NPC walks over this gesture, he or she will immediately collapse into some much needed slumber. Like before, gesture assignments are randomized, and players continue to evade or put citizens to sleep until we collect a predetermined number of samples from each gesture class.

Design Considerations

In designing Sleepy Town, we were concerned with providing a realistic game play experience to ensure ecological validity. While a game can take on almost any form, a common design approach is to mimic elements found among successful games [1]. In this regard, we provide a clear objective, visual feedback, interactive elements, risk, reward, and variability through navigation, gestures, and health pack pickups. Further, Sleepy Town adheres to the playability heuristics described by Desurvive *et al.* [4] and Pinelle *et al.*'s usability principles [17]. For example, the camera is never obscured. Relevant game state information is always presented in the form of an overlay with health information. Enemy behavior and user movement are similarly consistent and were found to be fair by users. In this way, we ensure an ecologically valid game play experience. Further, Sleepy Town's design is inspired by prior work [3, 27]. In our design, we ensure that gestures are also spatially relevant and require interaction with virtual objects. To illustrate, gesticulation requires that players align their gestures to intercept citizens along their trajectory. We believe this can influence shape, time, size, and sloppiness. And depending on the relationship between citizen, camera, and environment, the ability of a player to intercept a citizen will vary. Although we did not collect player experience metrics, we anecdotally received unsolicited positive feedback and participant requests for us to publish our

[2]A polyphasic sleep schedule in which individuals nap for 20 minutes at equidistant intervals throughout the day, usually six times per day. This schedule is notoriously difficult to adopt.

game on an app store, which suggests that we provided a compelling gameplay experience.

PERFORMANCE MEASURES

To understand gesture production differences between data collection applications, we employ a variety of performance measures: relative, global, and coverage. Each type offers a unique perspective on how users gesticulate within a given environment, which we discuss in this section.

Relative Accuracy Measures: One set of twelve designed by Vatavu *et al.*[33] for stroke gestures are the so-called relative measures. Given a gesture set, one first selects a representative task axis, typically the distribution's centroid. One then measures each remaining sample against this axis to understand how gestures vary within the random sample relative to a canonical form. An example relative measure referred to as the shape error follows:

$$\mathrm{ShE}(p) = \frac{1}{n}\sum_{i=1}^{n} \| p_{\sigma(i)} - \bar{p}_i \|, \tag{1}$$

where p is a stroke uniformly resampled to n points, $\bar{p}$ is the similarly resampled task axis, σ is a permutation function that aligns points in p with points from $\bar{p}$. In words, ShE measures the average difference between corresponding points. A brief description of each relative measure is given in Table 1, though the associated mathematics are omitted (see [33] for more information). In addition to then ten listed relative measures, we also report their geometric mean (Mean Measure) as a summary of error and variance across all measures, enabling one to quickly see the aggregate effect between conditions.

One issue with the relative measures in their specified form is that they do not allow for a direct comparison between distributions collected by unique protocols and/or hardware. For example, when one collects a distribution of samples with a large display compared to that of a smaller display, then the Euclidean distance-based shape error results will report a larger dispersion in the large display condition compare to that collected with the smaller display apparatus. For this reason, we z-score normalize both position and time data for all samples residing within the same distribution. Specifically, we measure the bounding box size for all samples within a distribution and subsequently rescale all samples by the largest z-score normalized extent, so as to preserve aspect ratio. This normalization step allows us to directly compare the intra-distribution dispersion of those measures reported in Table 1 when collected amongst different data collection devices and protocols. We refer to these as *scaled* relative measures.

Global Measures: In addition to those relative measures just discussed, we also collect and report on a variety of absolute global measures. Namely, we examine the bounding box area, path length, gesture production time, and indicative angle variance, which are classic measures commonly used in gesture production analysis [21]. In our context, bounding box area informs one about the size of gesticulations across protocols as does path length. Differences in size and length force one to consider possible explanations for why users choose to vary their size with respect to the given task and apparatus. Gesture production time provides insight into how hurriedly a population produces gestures under a given condition, and variation in the indicate angle gives insight into orientation consistency under the same conditions. This latter measure is especially important given that a recent trend in recognizer research has been to drop rotation invariance [26, 28, 30, 32, 34].

Name	Abbr	Description
Shape Error	ShE	Average difference between corresponding points
Shape Variability	ShV	Standard deviation of shape error differences
Length Error	LE	Sum of differences between gesture arc-lengths across corresponding points
Size Error	SzE	Difference in bounding box sizes
Bending Error	BE	Average of differences between turning angle at corresponding points
Bending Variability	BV	Standard deviation of turning angle difference between corresponding points
Time Error	TE	Difference in gesture production time
Time Variability	TV	Standard deviation of time difference between corresponding points
Speed Error	VE	Average difference in speed between corresponding points
Speed Variability	VV	Standard difference between differences in speed
Mean Measure	MM	Geometric mean of above relative measures

Table 1: Subset of the relative measure defined by Vatavu *et al.* [33] that we use in this work. The Mean Measure, however, is new in this work.

Coverage Measures: Both relative and global measures yield important information on dispersion, yet fail to provide insight on form differences between distributions. For instance, although two unique random samples are identically self similar according to a given relative measure, this does not guarantee that their shapes are similarly identical, which directly impacts a recognizer's ability to match patterns. For this reason, we also report coverage via the modified Hausdorff distance [5], defined as:

$$H(\mathcal{A},\mathcal{B}) = \max\left(d(\mathcal{A},\mathcal{B}), d(\mathcal{B},\mathcal{A})\right) \tag{2}$$
$$d(\mathcal{A},\mathcal{B}) = \frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}} \min_{b\in\mathcal{B}}\left(f(a,b)\right),$$

where $\mathcal{A}$ and $\mathcal{B}$ are independent gesture sets, and f is a dissimilarity measure. In words, we calculate the average dissimilarity of each sample in a first dataset to its nearest neighbor in the second dataset, and we do this again in the opposite direction, which yields two averages. The maximum of these average dissimilarities then gives us a sense of how well the distributions cover each other.

A number of recognizers utilize Euclidean Distance in their pattern matching approach. For this reason, we employ the squared Euclidean Distance over normalized samples as our primary dissimilarity measure (f in Equation 2). Similar to \$1 [36], we resample, scale, and rotate all samples before making our measurement to ensure we are comparing differences in underlying form.

USER STUDY

We conducted an experiment to better understand differences in gesticulation between a standard data collection protocol and games using the applications described in our previous section. We designed our experiment to test the following hypotheses:

Hypothesis 1 (H1): *Gesture production variability is application dependent.*

Hypothesis 2 (H2): *Standard practice yields the least amount of gesture production variability.*

Hypothesis 3 (H3): *Sleepy Town yields the most amount of gesture production variability.*

Subjects and Apparatus

We recruited 18 participants (12 male and 6 female) from a local university, all were right handed, all had prior experience with touch-based electronic interfaces and 14 had prior experience with pen-based electronic interfaces. Further, the population's mean age was 20.3 years old and ranged from 18 to 28. The experiment duration ranged from 30 to 50 minutes, and each participant was compensated \$10 for their time. For a pen-based interactive display we used a Wacom Cintiq Pro 16 with display size of 15.6 inches (37.62 cm) and resolution of 3840x2160 pixels (UHD). We used the stylus (6 inch, 15.5 cm) included with the Wacom device for data collection.

FTL: Implementation

To facilitate offline processing, we record which leader or gesture request command is displayed at each moment in time. When FTL makes a gesture request, we inform the participant, and once the associated gesture is complete, we press a key that logs the request as complete. Afterward, we use automation implementing Penny Pincher [24] to classify each stroke recorded during the session. We further visually confirm all results and manually correct any errors.

Sleepy Town: Implementation

Actions are fast and sporadic in this game environment, so it is not possible to classify strokes in real-time as users play our game. Instead, using key button presses, we count when we believe players produce certain gestures. Once we collect a sufficient sample count for each gesture based on investigator key-press feedback, the game terminates and all strokes are saved to disk. In a post-processing step, all strokes are classified and errors are manually corrected.

Procedure

We presented each participant with a consent form that explained our experimental procedure and informed him or her of their rights. We then gave each individual a pre-questionnaire so as to collect demographic information, after which we explained our research. Participants were next introduced to the ten gesture classes shown in Figure 1 and allowed to practice them on paper until they were satisfied with their performance, though a reference sheet was kept nearby throughout the entire session. Once comfortable, participants used each of the three data collection applications in a counterbalanced order. For each application, we first introduced its mechanics to the participant and then allowed him or her to practice until they were confident. Thereafter, we recorded at least six samples of each gesture class. Finally, we asked participants to fill out the NASA Task Load Index (TLX) questionnaire upon completion of each data collection task so as to assess subjective workload.

We chose to use the ten gestures shown in Figure 1 because of their prevalence throughout the custom gesture recognition literature. Although there were many to choose from, these ten also vary in familiarity and difficulty, and have good separability, which facilitates the use of our offline post-processing tools. Our choice to limit the gesture class count to ten was driven only by logistics. To produce a minimum of six samples per class over three conditions took the longest participants approximately one hour. We feared that more samples or classes would lead to fatigue or effortless gesticulation.

Design and Analysis

We chose a within-subject design for our experiment in order to compare writer-dependent gesture production variations across each of the three data collection applications. In this way, we had one independent variable, *application*, with three levels: Standard, FTL, and Sleepy Town. Our dependent variables are the global, relative, and distribution coverage measures discussed in the previous section.

For each participant, per gesture, we first computed each measure. We then averaged together the individual gesture class results per participant, which provided us with eighteen responses per condition. We thereafter used Friedman omnibus testing to detect differences between treatments, and exact Wilcoxon signed-rank testing for post-hoc analysis. Finally, we used the Holm–Bonferroni step down procedure to control family-wise error rates [9].

RESULTS

Relative Measures

Relative measure results are shown in Figure 5, and Friedman tests showing significant differences across all conditions for all metrics are shown in Table 2. Post-hoc analysis of the relative measures provide additional insight into the differences between different applications and their distributions, where the pairwise comparison results are presented in Table 3.

We first note that the Shape Error and Shape Variability measures between all applications are significantly different, increasing from standard practice to FTL and again from FTL to Sleepy Town. This result indicates that the position of corresponding points in a normalized space after spatial resampling are less varied under standard data collection practices.

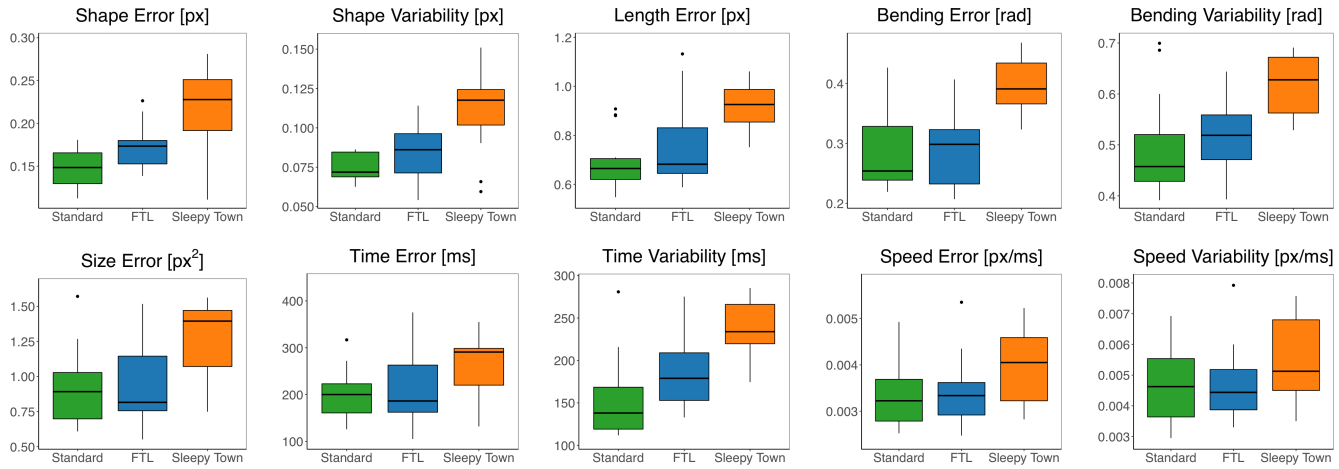

Figure 5: Collection of relative metrics. The top row, from left to right, includes Shape Error, Shape Variability, Length Error, Bending Error, and Bending Variabilility. The bottom row contains Size Error, Time Error, Time Variability, Speed Error and Speed Variability.

Name	Test Statistic and Significance
Shape Error	$\chi^2(2) = 25.33, p < 0.00001$
Shape Variability	$\chi^2(2) = 30.33, p < 0.00001$
Length Error	$\chi^2(2) = 14.88, p < 0.001$
Size Error	$\chi^2(2) = 10.11, p < 0.01$
Bending Error	$\chi^2(2) = 23.11, p < 0.00001$
Bending Variability	$\chi^2(2) = 22.33, p < 0.0001$
Time Error	$\chi^2(2) = 16.78, p < 0.001$
Time Variability	$\chi^2(2) = 25.44, p < 0.00001$
Speed Error	$\chi^2(2) = 19.44, p < 0.05$
Speed Variability	$\chi^2(2) = 23.18, p < 0.00001$
Mean Measure	$\chi^2(2) = 28.78, p < 0.00001$

Table 2: Friedman test results for relative measures. All results were significant.

We see similar trends in Bending Error and Bending Variability, which shows that the angles between corresponding point triplets yield increased curvature differences with increased game complexity. Size and Length Error are not significantly different between standard practice and FTL, but both differ from Sleepy Town, where we observe less consistency. That is the relative variation in size and temporal alignment between points is greatest in Sleepy Town. We see a similiar result for Speed Error and Variability in that Sleepy Town exhibits the most error and variability. Finally, the Mean measure (Figure 6 left) clearly echos the individual relative measure results—standard practice gesture productions are most consistent and Sleepy Town least, leaving FTL in the middle.

Global Measures

Global measure comparisons results are shown in Figure 6. Results of our Friedman tests showed significant differences across conditions for Area ($\chi^2(2) = 20.11, p < 0.05$), Angle Variance ($\chi^2(2) = 25, p < 0.05$), Length ($\chi^2(2) = 21.78, p < 0.05$), and Duration ($\chi^2(2) = 27.44, p < 0.05$) only. Post-hoc analysis provided further insight about differences between the conditions. For Area, there was a difference between standard practice and Sleepy Town ($Z = 3.79, p < 0.001, r = 0.63$), as well as FTL and Sleepy Town ($Z = 4.08, p < 0.0001, r = 0.68$). For Angle Variance, there was a difference between standard practice and Sleepy Town ($Z = -3.87, p < 0.0001, r = 0.64$), as well as FTL and Sleepy Town ($Z = -2.95, p < 0.01, r = 0.49$) and FTL and standard practice ($Z = -2.41, p < 0.05, r = 0.40$). Finally, Duration showed a difference between standard practice and Sleepy Town ($Z = 4.23, p < 0.0001, r = 0.71$), as well as FTL and Sleepy Town ($Z = 2.12, p < 0.05, r = 0.35$) and FTL and standard practice ($Z = 4.23, p < 0.0001, r = 0.71$). Note that although FTL and Sleepy Town measure lower in Area and Duration, their relative measures show greater error and variability.

Coverage Measures

Coverage measure comparison results are shown in Figure 7. Our Friedman tests showed significance both between conditions ($\chi^2(2) = 32.44, p < 0.05$), and within conditions ($\chi^2(2) = 36, p < 0.05$) for our coverage metric based on modified Hausdorff distance. Post-hoc analysis shows significant differences between each pairwise comparison (Standard-FTL, Standard-Sleepy Town, and FTL-Sleepy Town). For intra-condition distances, all pairwise combinations were equally significant ($Z = -4.23, p < 0.0001, r = 0.71$). For inter-condition distances, Standard-FTL vs FTL-Sleepy Town and Standard-FTL vs Standard-Sleepy Town were equally significant ($Z = -4.23, p < 0.0001, r = 0.71$), while Standard-Sleepy Town vs FTL-Sleepy Town was slightly less significant ($Z = 3.96, p < 0.0001, r = 0.66$). These results further support our prior findings that the distribution of samples generated by an individual application do not necessarily cover the space of other applications, and standard practice produces the least variable results.

Name	Standard-FTL	Standard-Sleepy Town	FTL-Sleepy Town
Shape Error	$Z = -3.79, p < 0.001, r = 0.63$	$Z = -4.08, p < 0.0001, r = 0.68$	$Z = -3.79, p < 0.001, r = 0.63$
Shape Variability	$Z = -3.96, p < 0.0001, r = 0.66$	$Z = -3.96, p < 0.0001, r = 0.66$	$Z = -4.23, p < 0.0001, r = 0.66$
Length Error	$Z = -1.29, p = 0.20$	$Z = -3.37, p < 0.001, r = 0.56$	$Z = -3.01, p < 0.01, r = 0.50$
Size Error	$Z = 1.16, p = 0.25$	$Z = -3.16, p < 0.01, r = 0.53$	$Z = -2.65, p < 0.01, r = 0.44$
Bending Error	$Z = -2.33, p < 0.05, r = 0.39$	$Z = -4.08, p < 0.0001, r = 0.68$	$Z = -4.08, p < 0.0001, r = 0.68$
Bending Variability	$Z = -2.17, p < 0.05, r = 0.36$	$Z = -3.16, p < 0.01, r = 0.53$	$Z = -3.16, p < 0.01, r = 0.53$
Time Error	$Z = 0.212, p = 0.83$	$Z = -2.89, p < 0.01, r = 0.48$	$Z = -2.76, p < 0.01, r = 0.46$
Time Variability	$Z = -2.61, p < 0.001, r = 0.43$	$Z = -4.16, p < 0.0001, r = 0.69$	$Z = -4.23, p < 0.0001, r = 0.71$
Speed Error	$Z = -1.15, p = 0.25$	$Z = -4.08, p < 0.0001, r = 0.68$	$Z = -3.32, p < 0.001, r = 0.55$
Speed Variability	$Z = 0.45, p = 0.65$	$Z = -3.30, p < 0.001, r = 0.55$	$Z = -3.20, p < 0.01, r = 0.53$
Mean Measure	$Z = -2.98, p < 0.01, r = 0.50$	$Z = -4.23, p < 0.0001, r = 0.71$	$Z = -4.16, p < 0.0001, r = 0.69$

Table 3: Pairwise Wilcoxon signed rank test results for relative measures.

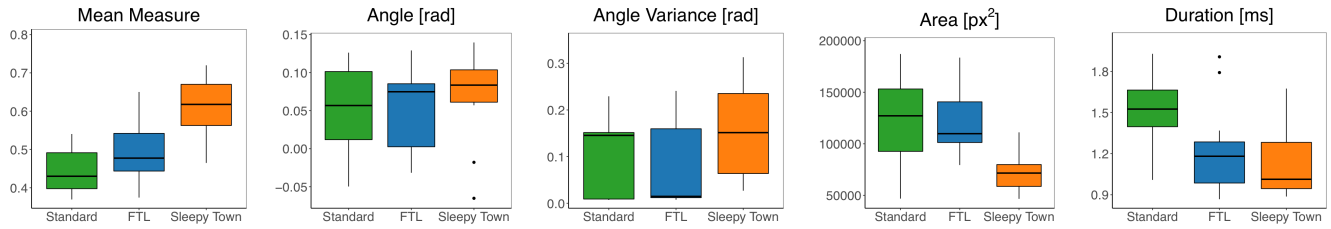

Figure 6: Mean Measure (geometric mean) of all relative measures, followed by the global measures that are Indicative Angle, Indicative Angle Variance, Gesture Area, and Duration from left to right.

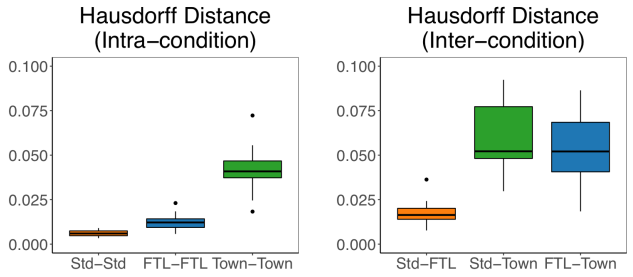

Figure 7: Results of comparisons betweeen gesture form from different conditions using Hausdorff Distance.

Perceived Workload

To analyze perceived workload, we ran a Friedman test on the raw NASA TLX data collected from the users after completing each task [7]. The Standard collection tool showed lower workload ($M = 31.22, SD = 20.0$) than both Sleepy Town ($M = 48.44, SD = 23.7$) and FTL ($M = 80, SD = 25.0$). There was a significant difference between the three conditions ($\chi^2(2) = 23.07, p < 0.05$). Our post-hoc analysis found that there were differences between Standard collection tool and FTL ($p < .0001$), Sleepy Town and FTL ($p < .001$), and Standard collection tool and Sleepy Town ($p < .0001$). The increased task load introduced by FTL did not introduce additional gesture variability, as seen in the global and relative measures.

DISCUSSION

From our experiment, it is clear that data collection protocols influence gesture form, and by introducing even minor game play mechanics to a protocol, we can expect a significant increase in variability. Specifically, by analyzing global measures, we observe differences in production time, where both game conditions elicited faster responses relative to standard practice. Since speed has been correlated with recognition accuracy [33], and we find that participants gesticulate with greater haste under our alternative conditions, future data collection protocols should work to elicit speed variability.

Second, we found a significant difference in size and orientation variability between Sleepy Town and the remaining conditions. Interestingly, participants were as likely to draw large gestures when playing FTL as they were when using standard practice. However, in Sleepy Town, players were forced to interact with dynamic content in a timely manner. We believe that these targeted interactions had an influence on gesture size. For a similar reason, we visually observed that some players oriented their gestures toward Sleepy Town citizens so as to intercept individuals along their apparent trajectories and to put spells on specific characters. Such orientation variability has an important effect on recognizer accuracy with respect to alignment. In recent years, custom gesture recognizers have dropped support for rotation invariance [26, 28, 30, 32, 34]; however, with this new data in hand, practitioners might consider again adopting rotation invariance, especially for circumstances that involve interactions with dynamic content. Intuitively, our findings also suggest that future data collection protocols should work to elicit variability in both orientation and size.

Relative accuracy measures also reveal a number of insights. Across all twelve measures analyzed, Sleepy Town elicited significantly higher variation compared to FTL and standard practice. Of particular interest are Shape Error and Variabil-

ity, because like speed, these measures have been correlated with recognizer accuracy [33], where lower values lead to better performance. Specifically, these measures inform us about the average deviation and variation between corresponding points of a given sample against the distribution's centroid, both of which are significantly higher in our game environments. In other words, we find that players are less consistent in their productions when reacting to dynamic content. Because standard practice does require such interactions, data captured with such tools are unlikely to exercise recognizers to the same extent. Bending error and variability tell a similar story.

Finally, coverage measures reinforce what we already learned from relative accuracy measure analysis—-standard practice yields a more consistent random sample. We find that the distance between nearest neighbors within the same distribution are furthest apart in the game environments, with Sleepy Town being greatest by a considerable margin. Consequently, standard practice data is unable to provide adequate coverage for either game environment. This finding is consistent with prior work, where researchers found accuracy drops in video games relative to tests conducted with training data [3, 27, 28].

Findings across all measures confirm our hypothesis that gesture production variability is application dependent, as across each condition, we observe unique variations in size, speed, orientation, and form. We further confirmed our second hypothesis that standard practice yields the most consistent sampling. And finally, we also confirmed that Sleepy Town, being the most complex game with respect to interaction style, elicits the most variable responses. Based on these findings, we recommend that user interface designs and pattern recognition researchers validate their work with data collected from within rather than outside of their target environment, or adopt new data collection protocols. Given that standard practice does yield sufficient variability, results reported with such data represent optimistic upper bounds on performance rather than provide clear expectations.

About Follow the Leader

We designed FTL as a data collection application for low cost implementation effort in that many design choices were driven by practical logistical considerations. Consider that a typical standard application already has the ability to collect and render gesture data. Using this functionality, a practitioner can easily collect leader samples that their system replays while it displays text commands and collects new input. Subsequently, using any off-the-shelf recognizer such as $Q [34], one can post process their data to classify new input using designer made templates, after which one just manually corrects any minor missclassification error. Inline with simplicity, FTL does not time, score, nor provide any user performance feedback, though one could if they so chose. Despite this apparent feature scarcity, FTL is still able to elicit highly variable responses relative to standard practice, and for these reasons, we believe FTL is a good starting point for future data collection efforts as we begin to move toward more ecologically valid protocols.

Limitations and Future Directions

In this work we were able to show that differences in data collection protocols lead to differences in gesture production variability. However, we did not identify which specific factors cause variability. For instance, Sleepy Town gestures were on average significantly smaller than those produced within the standard data collection and FTL applications. Does size impact variability? We also saw differences in speed, orientation, and time. Do these differences result from how players interact with objects in the virtual environment? To what extent does cognitive load impact gesture form? We intend to explore these factors in greater detail.

We further focused on unistroke gestures of varying complexity to facilitate user study duration, but we intend to follow up with multistroke as well as hand and full body gesture analysis. We also intend to explore additional FTL game mechanics that may yield even more variability, including orientation and timed response requirements. Last, we also believe FTL will be especially useful for high activity continuous data acquisition, as players are forced to maneuver between gesture and non-gesture interactions. In this way, FTL will capture challenging datasets that researchers and practitioners can use to test gesture segmentation and recognition.

CONCLUSION

We have presented results from a user study demonstrating that standard data collection protocols do not capture the true variability of gesticulation within a game environment. This result holds for even our simplest game, Follow the Leader (FTL), which yielded variation significantly different from that of standard practice. Our second game, Sleepy Town, generated even greater variability, which was again significantly different from both. Differences between these distributions were validated using seventeen global, relative, and coverage measures. Our findings motivate the need for researchers and designers to move away from standard practice, and for the community to develop new ecologically valid data collection protocols. We believe that FTL is a good first step solution, as FTL requires little effort to implement, being built on tools already present in standard data collection applications, and that elicits greater variability.

ACKNOWLEDGMENTS

This work is supported in part by NSF Award IIS-1638060 and Army RDECOM Award W911QX13C0052. We also thank the anonymous reviewers for their insightful feedback. We are further grateful to the Interactive Systems and User Experience lab at UCF for their support.

REFERENCES

[1] Luke Ahearn. 2000. *Designing 3D games that sell!* Charles River Media, Inc.

[2] Xiang Cao and Shumin Zhai. 2007. Modeling Human Performance of Pen Stroke Gestures. In *CHI Conference*. IBM Almaden Research Center and University of Toronto, 1495–1504.

[3] Salman Cheema and Joseph J LaViola. 2011. Wizard of Wii: toward understanding player experience in first

person games with 3D gestures. In *Proceedings of the 6th International Conference on Foundations of Digital Games*. ACM, 265–267.

[4] Heather Desurvire, Martin Caplan, and Jozsef A Toth. 2004. Using heuristics to evaluate the playability of games. In *CHI'04 extended abstracts on Human factors in computing systems*. ACM, 1509–1512.

[5] M-P Dubuisson and Anil K Jain. 1994. A modified Hausdorff distance for object matching. In *Proceedings of 12th international conference on pattern recognition*, Vol. 1. IEEE, 566–568.

[6] Chris Ellis, Syed Zain Masood, Marshall F Tappen, Joseph J Laviola Jr., and Rahul Sukthankar. 2013. Exploring the Trade-off Between Accuracy and Observational Latency in Action Recognition. *International Journal of Computer Vision* 101, 3 (feb 2013), 420–436.

[7] Sandra G Hart. 2006. NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, Vol. 50. Sage publications Sage CA: Los Angeles, CA, 904–908.

[8] Javier Hernandez-Ortega, Aythami Morales, Julian Fierrez, and Alajandro Acien. 2017. Predicting Age Groups from Touch Patterns based on Neuromotor Models. In *International Conference of Pattern Recognition Systems*. BiDA Lab, 1–6.

[9] Sture Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics* (1979), 65–70.

[10] Luis A Leiva, Daniel Martín-Albo, and Réjean Plamondon. 2015. Gestures À Go Go: Authoring Synthetic Human-Like Stroke Gestures Using the Kinematic Theory of Rapid Movements. *ACM Trans. Intell. Syst. Technol.* 7, 2 (nov 2015), 15:1—-15:29.

[11] Luis A Leiva, Daniel Martín-Albo, and Radu-Daniel Vatavu. 2017. Synthesizing stroke gestures across user populations: A case for users with visual impairments. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 4182–4193.

[12] Laurence Likforman-Sulem, Anna Esposito, Marcos Faundez-Zanuy, Stéphan Clémençon, and Gennaro Cordasco. 2017. EMOTHAW: A Novel Database for Emotional State Recognition From Handwriting and Drawing. In *IEEE TRANSACTIONS ON HUMAN-MACHINE SYSTEMS, VOL. 47*. IEEE, 273–284.

[13] Jiayang Liu, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. 2009. uWave: Accelerometer-based personalized gesture recognition and its applications. *Pervasive and Mobile Computing* 5, 6 (2009), 657–675.

[14] Gil Luria and Sara Rosenblum. 2010. Comparing the Handwriting Behaviours of True and False Writing with Computerized Handwriting Measures. In *Applied Cognitive Psychology, issue 24*. Department of Human Services, Haifa University, 1115–1128.

[15] Mehran Maghoumi and Joseph J LaViola Jr. 2018. DeepGRU: Deep Gesture Recognition Utility. *CoRR* abs/1810.1 (2018).

[16] Daniel Martín-Albo, Réjean Plamondon, and Enrique Vidal. 2014. Training of on-line handwriting text recognizers with synthetic text generated using the kinematic theory of rapid human movements. In *2014 14th International Conference on Frontiers in Handwriting Recognition*. IEEE, 543–548.

[17] David Pinelle, Nelson Wong, and Tadeusz Stach. 2008. Heuristic evaluation for games: usability principles for video game design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1453–1462.

[18] Réjean Plamondon. 1995a. A kinematic theory of rapid human movements. Part I: Movement representation and control. *Biological Cybernetics* 72, 4 (mar 1995), 309–320.

[19] Réjean Plamondon. 1995b. A kinematic theory of rapid human movements. Part II. Movement time and control. *Biological cybernetics* 72 4 (1995), 309–20.

[20] Réjean Plamondon and Moussa Djioua. 2006. A multi-level representation paradigm for handwriting stroke generation. *Human movement science* 25, 4-5 (2006), 586–607.

[21] Dean Rubine. 1991. Specifying Gestures by Example. *SIGGRAPH Computer Graphics* 25, 4 (jul 1991), 329–337.

[22] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. 2016. NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[23] Steven W Smith. 1997. *The scientist and engineer's guide to digital signal processing*. California Technical Pub. San Diego.

[24] Eugene M Taranta II and Joseph J LaViola Jr. 2015. Penny Pincher: A Blazing Fast, Highly Accurate $-family Recognizer. In *Proceedings of the 41st Graphics Interface Conference (GI '15)*, Vol. 2015-June. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 195–202.

[25] Eugene M Taranta II, Mehran Maghoumi, Corey R Pittman, and Joseph J LaViola Jr. 2016. A rapid prototyping approach to synthetic data generation for improved 2D gesture recognition. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, ACM, New York, NY, USA, 873–885.

[26] Eugene M Taranta II, Amirreza Samiei, Mehran Maghoumi, Pooya Khaloo, Corey R Pittman, and Joseph J LaViola Jr. 2017. Jackknife: A Reliable Recognizer with Few Samples and Many Modalities. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 5850–5861.

[27] Eugene M Taranta II, Thaddeus K Simons, Rahul Sukthankar, and Joseph J Laviola Jr. 2015. Exploring the Benefits of Context in 3D Gesture Recognition for Game-Based Virtual Environments. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 5, 1 (2015), 1.

[28] Eugene M Taranta II, Andrés N Vargas, and Joseph J LaViola Jr. 2016. Streamlined and accurate gesture recognition with Penny Pincher. *Computers & Graphics* 55 (2016), 130–142.

[29] Jean Vanderdonckt, Paolo Roselli, and Jorge Luis Pérez-Medina. 2018. ! FTL, an Articulation-Invariant Stroke Gesture Recognizer with Controllable Position, Scale, and Rotation Invariances. In *Proceedings of the 2018 on International Conference on Multimodal Interaction*. ACM, 125–134.

[30] Radu-Daniel Vatavu. 2017. Improving Gesture Recognition Accuracy on Touch Screens for Users with Low Vision. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 4667–4679.

[31] Radu-Daniel Vatavu, Lisa Anthony, and Quincy Brown. 2015. Child or adult? Inferring Smartphone users' age group from touch measurements alone. In *IFIP Conference on Human-Computer Interaction*. Springer, 1–9.

[32] Radu-Daniel Vatavu, Lisa Anthony, and Jacob O Wobbrock. 2012. Gestures As Point Clouds: A $P Recognizer for User Interface Prototypes. In *Proceedings of the 14th ACM International Conference on Multimodal Interaction (ICMI '12)*. ACM, New York, NY, USA, 273–280.

[33] Radu-Daniel Vatavu, Lisa Anthony, and Jacob O Wobbrock. 2013. Relative Accuracy Measures for Stroke Gestures. In *Proceedings of the 15th ACM on International conference on multimodal interaction (ICMI '13)*. ACM, ACM, New York, NY, USA, 279–286.

[34] Radu-Daniel Vatavu, Lisa Anthony, and Jacob O Wobbrock. 2018. $ Q: a super-quick, articulation-invariant stroke-gesture recognizer for low-resource devices. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*. ACM, 23.

[35] Radu-Daniel Vatavu, Daniel Vogel, Géry Casiez, and Laurent Grisoni. 2011. Estimating the Perceived Difficulty of Pen Gestures. In *Proceedings of the 13th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part II (INTERACT'11)*. Springer-Verlag, Berlin, Heidelberg, 89–106.

[36] Jacob O Wobbrock, Andrew D Wilson, and Yang Li. 2007. Gestures Without Libraries, Toolkits or Training: A $1 Recognizer for User Interface Prototypes. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology (UIST '07)*. ACM, New York, NY, USA, 159–168.

[37] L. Xia, C.C. Chen, and JK Aggarwal. 2012. View invariant human action recognition using histograms of 3D joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*. IEEE, 20–27.