Human Computation (2015) 2:2:135-154

ISSN: 2330-8001, DOI: 10.15346/hc.v2i2.2

Target Acquisition and the Crowd Actor

JEFFREY P. BIGHAM, CARNEGIE MELLON UNIVERSITY

WALTER S. LASECKI, UNIVERSITY OF MICHIGAN

JACOB O. WOBBROCK, UNIVERSITY OF WASHINGTON

ABSTRACT

Work in human-computer interaction has generally assumed either a single user or a group of users working together in a shared virtual space. Recent crowd-powered systems use a different model in which a dynamic group of individuals (the crowd) collectively form a single actor that responds to real-time performance tasks, *e.g.*, controlling an on-screen character, driving a robot, or operating an existing desktop interface. In this paper, we introduce the idea of the *crowd actor* as a way to model coordination strategies and resulting collective performance, and discuss how the crowd actor is influenced not only by the domain on which it is asked to operate but also by the *personality* endowed to it by algorithms used to combine the inputs of constituent participants. Nowhere is the focus on the individual performer more finely resolved than in the study of the human psychomotor system, a mainstay topic in psychology that, largely owing to Fitts' law, also has a legacy in HCI. Therefore, we explored our notion of a crowd actor by modeling the crowd as a individual motor system performing pointing tasks. We combined the input of 200 participants in a controlled offline experiment to demonstrate the inherent trade-offs between speed and errors based on personality, the number of constituent individuals, and the mechanism used to distribute work across the group. Finally, 10 workers participated in a synchronous experiment to explore how the crowd actor responds in a real online setting. This work contributes to the beginning of a predictive science for the general crowd actor model.

1. INTRODUCTION

Work in human-computer interaction has generally assumed either a single user, or groups of users working together in a shared virtual space [15, 16, 34]. Recent crowd-powered systems explore a different model in which a dynamic group of individuals (the crowd) collectively form a single actor [23, 24, 36]. These models show benefits in speed, consistency, and ability to be applied to interfaces already in use today (which were likely designed not for group control but for control by an individual). Groups can now collectively play video games designed for a single person [27], collectively drive a robot [12, 23], and control existing user interfaces by collectively deciding what key presses and mouse movements should be sent to them [23].

Despite this interest, the focus thus far has primarily been on how to get collective control to work at all, rather than on understanding its benefits and tradeoffs. As one example, Twitch Pokemon[1], which allowed many thousands of users to collectively play Pokemon in a Gameboy emulator, attracted considerable interest because the collective was able to progress through and eventually even beat the game, with each individual supplying input at the level of individual button presses[2].

The systems discussed thus far share the notion of abstracting a group as an individual to perform real-time performance tasks on existing interfaces. In this paper, we formalize this idea as the *crowd actor*, and explore what we believe to be one of its most interesting and desirable qualities – the ability to outperform even the best human actor on real-time performance tasks. Nowhere is the focus on the individual performer more finely resolved than in the study of the human psychomotor system, a mainstay topic in psychology that, largely owing to Fitts' law [11], also has a legacy in HCI [29]. Therefore, in this paper we explore our notion of a crowd actor by modeling the crowd as a individual motor system.

What makes the crowd actor an interesting area for scientific inquiry is that while it is clear that a group should be able to outperform an individual, it is less than clear how to coordinate the group's effort to realize that potential. For instance, how should a group work together during a pointing task? The task is over quickly and the system state (cursor position) changes quickly; therefore, explicit coordination at each time step would be cumbersome and slow. On the other hand, a simple approach like letting individuals in the crowd choose a target and then taking a vote also has drawbacks. Namely, waiting for everyone's vote means slowing the process down to the speed of the slowest individual.

As we will see, a number of common characteristics define the crowd actor. Some apply generally across different domains – for instance, we expect potential performance to be a function of such qualities as the size of the crowd, the capabilities of its constituent workers, and the rate of turnover in the crowd. But, the realized performance depends also on choices made during the construction of the crowd actor and the domain on which is applied, *e.g.*, how work is distributed across individuals, and how their input is aggregated.

Each instantiation of a crowd actor model necessarily makes trade-offs. Diametric relationships exist in all human performance tasks, *e.g.*, the relationship between speed and error have been explored extensively in the context of pointing [41, 43]. Thus, the model for combining inputs of constituent workers into the output of the crowd actor can be tuned but cannot optimize all dimensions. The selected model is what we call the *personality* of the crowd[3]. For instance, if the model chooses the first input from among all inputs it receives, we may say the crowd actor's personality is *hasty*. Alternatively, if the model chooses an input only after observing all inputs from all crowd members, we might call its personality *cautious*.

[1] http://www.twitch.tv/twitchplayspokemon

[2] Twitch, which also allows individuals to screencast their own video game playing and subscribe to those of others, was recently sold to Amazon for nearly $1 billion.

[3] The definition of personality we use is: "the combination of characteristics or qualities that form an individual's distinctive character." We consider the crowd actor, comprised of many individuals, to be an individual actor.

We first explore the trade-offs of different crowd actor personalities in a series of pointing experiments with a total of 200 crowd workers drawn from Amazon Mechanical Turk. We combine the input of these workers offline in a controlled study designed to isolate the potential of the collective to outperform individuals on this task. We consider how measures like throughput, speed, and error rate are influenced by personality, the size of the crowd forming the crowd actor, and the mechanism used to solicit and combine their work. For instance, we find that the "hasty" personality completes the pointing task 33.8% faster than the average individual in the crowd, but this comes at the cost of 67.6% more errors. We find that a moderate personality that solicits a few responses (but does not wait for all of them) performs better than the more extreme "hasty" and "cautious" personalities, and out-performs even the best individual. Performance generally improves as the size of the underlying crowd grows, but this is not always true – for instance, a personality that utilizes the mean response performs worse than one that uses the median because it is more susceptible to outliers. In general, these results demonstrate the complexity of the crowd actor and the importance of the personality it takes on.

We then validate these results in a system to convert natural language instructions to clicks on a desktop interface.

This paper offers the following three contributions:

– We introduce the idea of the *crowd actor* model for modeling collective performance on real-time tasks, and formalize the notion of a crowd actor's personality as reflecting the algorithm by which the crowd actor makes choices and takes action.
– We demonstrate the idea of the crowd actor as applied to a target acquisition task, including variants that express different personalities.
– We compare different personalities of the crowd actor in a large-scale motor performance experiment, illustrating inherent trade-offs.

2. RELATED WORK

Our work on the crowd actor model for crowdsourcing was inspired by recent work in real-time human computation that puts a group (or crowd) collectively in control of a motor or cognitive task. Prior work in groupware has explored how groups of users could collaborate in shared online space [15], and many online games and multi-user dungeons (MUDs) likewise allow users to play or interact in the same space with one another. Crowd actors are different, however, because they synthesize the input of multiple workers to act as a single controller.

Some early web-based games allowed multiple users to control a single interface. For example, Massively Multiplayer Pong allows all of the current players to control the paddle [38]. Its interface displays both the "real" paddle position and the user-specific paddle position. Maynes-Aminzade *et al.* [32] have brought these techniques into the real world by enabling large audiences to collectively control a projected interface with collective actions like leaning to the left or right. Work such as Goldberg's Collaborative Telerobotics projects have looked at using group input to make decisions [12], and ShareCam asks a human actor to follow such crowd-derived commands in real time [13].

Human computation was introduced to integrate people into computational processes to solve problems too difficult for computers to solve alone, but most examples do not fall into the real-time domain. For instance, human computation has been shown useful in writing and editing [2], image description and interpretation [5, 40], and protein folding [8], among many other tasks. Existing abstractions focus on obtaining quality work, and generally introduce redundancy and layering into tasks so that multiple workers contribute and verify results at each stage, *e.g.*, guaranteeing reliability through answer agreement [40] or the find-fix-verify pattern of Soylent [2]. Unfortunately, these approaches take time, which makes these approaches unsuitable for real-time control. Naive solutions like recruiting a single online worker may allow for real-time control, but subvert existing methods of achieving reliability and are not robust to workers leaving (common in the crowd).

Several systems have explored how to make human computation interactive. As an example, VizWiz [5] answers visual questions for blind people quickly. It uses quikTurkit to pre-queue workers on Amazon's Mechanical Turk so that they will be available when needed. Crowd actors need multiple users to be available at the same time. Prior systems have also needed multiple workers to be available. For instance, the ESP Game encouraged accurate image labels by pairing players together and requiring them both to enter the same label [40]. Seaweed reliably got groups of Mechanical Turk workers to be available at the same time to play economic games by requiring the first worker to arrive to wait (generally for a few seconds) [7]. Systems like TurkServer make recruiting a group from Mechanical Turk easily [31]. LegionTools [14] is a publicly available[4] toolkit for retaining and routing workers to a task in about a second.

The work most related to crowd actors are systems that already employ what we would call crowd actors. For instance, the Legion system allows the dynamic crowd to control an existing desktop interface remotely through VNC [23]. Keyboard presses and mouse clicks from individuals in the crowd are merged together using what are called "input mediators." Legion introduces several input mediators, but the highest-performing ended up being the "leader" input mediator which used agreement in the inputs to elect temporary leaders who would assume full control of the interface. Legion's input mediators can be seen as instances of crowd actor personalities. Even without formally describing them as such, Legion had to deal with many of the same concerns, such as the trade-off between speed and accuracy. The results of experiments showed that while the "leader" input mediator allowed Legion to complete control tasks better than constituent crowd workers, they were not able to beat experts. Crowd actors help to formalize the notion of input mediators. Importantly, Legion demonstrates that groups can be brought together and coordinated around a given task. Rather than replicate this finding, we explore trade-offs on concrete target acquisition tasks.

A promising aspect of crowd actors is that they have the potential to outperform not only their constituent workers but also expert humans. For instance, the Rapid Refinement algorithm of the Adrenaline camera allows a synchronous crowd to quickly narrow in on a good frame of a video [3], generally much faster than a single person could because the crowd workers are able to consider different parts of the movie at the same time. The Legion:Scribe system is able to outperform even an expert stenographer at real-time captioning because each crowd worker only has to type part of what she hears [24]. Legion:AR used a similar approach to label actions in an activity recognition setting in real-time [26]. Apparition asked groups of workers to collectively improve and add functionality to an user-sketched interface within seconds of it being described [22]. Thus far, these systems have been one-off examples of what we believe to be a powerful underlying phenomenon that has not yet been explored.

[4]https://rochci.github.io/LegionTools/

Potential benefits of the crowd actor over individuals relate to the idea of collective intelligence and its application to human-computer interaction [4, 21]. Feedback between members of the crowd is not possible in our offline study because workers contribute their inputs asynchronously. Our synchronous experiment would allow for feedback to be given to crowd workers about collective performance, but the time demands of the clicking task (it requires low latency and is over quickly) make it difficult to provide meaningful feedback during a target acquisition trial. A rich area for future research would be in the collective dynamics that arise when the collective is able to see its own behavior and adjust accordingly [18]. Prior work on "self-correcting crowds" has shown that crowd members will often adjust their behavior to maximize collective rather than individual behavior [20]. In the case of the crowd actor, individuals may optimize for speed rather than reducing errors if they know that the aggregation strategy minimizes collective errors.

Our exploration of the crowd actor on a fundamentally individual task like mouse pointing represents our attempt to produce a common language and framework to discuss and compare such exciting developments with an eye towards furthering this nascent notion towards being a science.

3. THE CROWD ACTOR

A crowd actor is formed when input from a group of individuals who are trying to complete a matching task or subtask (e.g., [27]) are merged to form a single abstract individual, *e.g.*, a group collectively performing a target acquisition task as individuals usually do, or captioning a single stream of speech like individual professional stenographers do [24].

The output of a crowd actor appears as though it could have been produced via real-time interaction with an individual actor (including a super-performing individual actor). In practice, the motivation for the crowd actor model is that we believe it often introduces advantages that can be leveraged to exceed the maximum performance not only of constituent individuals but also of *any* individual. For instance, the crowd actor may be able to acquire a target more quickly and with less error than any one person. The crowd actor may be able to caption speech with higher quality and less latency than even a highly-trained stenographer.

Such performance gains are possible because constituent individuals need not perform the entire task or perform the task with as much care as he would need to if they were working alone. For instance, a constituent individual in the captioning task may type only part of what she hears, a constituent individual in a target acquisition task may operate more recklessly (with more error) than they would if working alone. It is thus incumbent on the mechanism used to merge work from the constituent individuals to correct for errors that otherwise may be introduced, *e.g.*, merging partial work together, or choosing correctly among inputs.

The "crowd agent" architecture [21] also attempts to allow a group to behave as a single individual, considering a broad range of necessary components, *e.g.*, constituent incentive mechanisms, collective memory and context, consistency of interaction, and task decomposition. The crowd actor focuses on action-level decisions and execution (*e.g.*, in motor tasks such as clicking) and the decision directly leading the system to take the action and *not*, for example, decision leading to the underlying intent.

3.1. Constituent Individuals

We believe several general qualities affect the quality of the crowd actor, although the particular mechanism for merging individual efforts into the crowd actor varies by domain. First, the quality and number of constituent individuals likely matters. This fits with expectations, as a good mechanism should allow a crowd actor composed of, for example, 10 workers to (on average) to perform better than a crowd actor comprised of only one. Moreover, a crowd actor composed of task experts should perform better than one composed of novices, all else being equal.

3.2. Personalities

Crowd actors are also impacted by what we call the "personality" endowed to them not by their constituent individuals but by the mechanism used to merge their work. Personalities represent how crowd actor mechanisms make trade-offs, *i.e.*, between speed and accuracy. Some general personality types include:

3.2.1. Hasty

"Hasty" personalities are characterized by their tendency to act as soon as possible, perhaps at the cost of not fully utilizing the input of multiple constituent individuals within the crowd actor. Redundancy is often used as a way to improve accuracy in human computation tasks, but redundancy takes time. For instance, in a pointing task, the hasty personality chooses the first click received to minimize latency without waiting to see whether other crowd members agree.

3.2.2. Cautious

"Cautious" personalities are the complement of hasty personalities. This personality is characterized by its tendency to wait until as much information is known as is possible before making a decision. In a pointing task, cautious personalities wait until the click locations of all constituent workers are known before deciding on a location itself to maximize accuracy at the cost of speed equal to the slowest worker.

3.2.3. Prudent

"Prudent" personalities fall in between hasty and cautious, and are characterized by their tendency to act not immediately but quickly, typically after receiving input from two or more crowd workers. For instance, in a pointing task, a crowd actor with a prudent personality may wait until it receives input from a few crowd workers before acting. If it has an *a priori* estimate on the time required for a crowd worker to provide input, it may wait a time equal to when it would expect to receive a small number of inputs before acting.

3.3. Mechanism

Finally, crowd actors are influenced by the mechanisms used to solicit and merge the work of constituent individuals. For instance, does the mechanism allow constituent workers to work on different sub-components of the problem? We have seen this scheme work in the real-time captioning domain, where each constituent individual is directed toward a different piece of the audio [24].

For target acquisition, asking constituent individuals to share the burden of moving the cursor along different portions of its trajectory likely does not make sense. Given multiple constituent workers we can change the problem in a different way – each individual can start from a different location, thereby making the task easier for some and increasing the likelihood that some will complete the task faster and with fewer errors.

The best mechanism varies by domain, but strongly impacts crowd actor performance. The best mechanisms either (i) divide work into smaller chunks that constituent crowd workers can perform better on, or (ii) distribute workers into different roles that they can focus on in order to lead to better crowd actor performance. For the pointing task, we distribute workers in space by assigning them different starting positions randomly. This way some crowd workers will be closer to new targets than they otherwise would be, allowing them to more quickly and accurately acquire a new target.

4. CROWD ACTOR MOTOR SYSTEM

In creating our crowd actor model of the human motor system and its personalities, we have built on prior work on modeling an individual human's motor system according to Fitts' law. When formulated for one dimension, Fitts' law [11] predicts the movement time MT to acquire a target of size W at distance A. Typically, MT is the *dependent* variable in Fitts' law. In Shannon formulation [29, 28], the law is written

$$MT = log_2\left(\frac{A}{W} + 1\right) \tag{1}$$

We can use this base theory to predict the performance of crowd actors with different parameters and personalities. For instance, Fitts' law captures what we observe in pointing tasks in practice: a subject makes a gross movement toward the target, usually misses, but then acquires the target with one or more corrective submovements [33]. Given this formulation, as the number of constituent workers in our crowd actor grows, we would expect the likelihood of one of the workers acquiring the target on the first gross movement without correction to increase. We therefore expect that as the number of constituent workers grows, both movement time and error should decrease because more workers will be expected to move to the target with the first gross movement. The effect may be similar to what we observe when W is made to be infinite, say by introducing an edge [44].

A primary parameter of our crowd actor is therefore N, the number of constituent workers forming the crowd actor. One of our hypotheses is that as N becomes larger, the better *potential* performance of the crowd actor.

4.1. Creating Personalities

In the fixed start condition, all individual crowd workers begin at the same position. Therefore, individually, their performance can be modeled by Fitts' law. The crowd actors however can perform better or worse than this depending on their personality. Below, we describe the crowd actor personalities we developed for pointing, and our expectations for their observed performance.

4.1.1. Hasty

The hasty (*fastest*) personality adopts the trial of the first worker to click. In a real system, target locations are unknown, and so this trial is chosen regardless of whether or not it hit the target. As N grows, we expect the hasty crowd actor's movement time to decrease substantially but at the cost of an increase in errors.

4.1.2. Cautious

We developed two variations on the *cautious* crowd actor for pointing. Both wait until all N crowd workers have clicked, and then choose either the *mean* or the *median* click position. In this case, the median is chosen as the geometric median, the point that minimizes the sum of the distances of all of the other points.

We expect that as N grows, movement time will increase because the actor cannot act until all constituent workers have finished. Accuracy will increase but quickly plateau at near zero errors.

4.1.3. Prudent

We have developed four *prudent* personalities for pointing, using the mean or median of the first 3 and 5 points received. The idea behind these personalities is that while the first click may often have errors, we expect the mean and median of a few fast clicks to be more accurate while retaining low latency. We expect that as N grows, observed movement time will go down, and errors will increase but stay low.

4.1.4. Best

Finally, we developed the *best* personality as a perfect crowd actor, although such a crowd actor would be impossible in practice. This actor adopts the click of the first constituent individual who successfully clicks on the target. We expect in real use cases, the location of the target will not be known by the system (if it were known, then the system could just click on it for the user). Nevertheless, this actor is a useful comparison to see how close to optimal are the crowd actors that could be used in practice.

4.2. **Random Start Mechanism**

Requiring all constituent workers to start from the same position does not fully utilize them. The second mechanism used by our crowd actor is to randomly start each worker at a different position. We expect this to reduce movement time as some workers will be closer to the target as a result (A will decrease), and some may even be randomly placed on top of the target ($A = 0$). These effects are likely to become more pronounced as N grows.

5. **EXPERIMENT**

Our experiment explored the trade-offs of the crowd actors that we developed, and tested our expectations regarding the performance of different personalities and the random start mechanism. We were particularly interested in throughput, movement time, and accuracy. We collected data on a

***Figure 1.** The web-based pointing task interface used for collecting trials from crowd workers on Mechanical Turk. The blue bar was the click target, and counted as a 'hit' if clicked. The number of hits and misses, along with the total time between target clicks were recorded.*

standard one-dimensional reciprocal pointing task [37], which we then combined using the crowd actors described in the previous section.

5.1. Method

5.1.1. Participants

The experiment was conducted with 200 workers recruited from Amazon Mechanical Turk[5], paid $0.50 USD for their participation, which required approximately 5 minutes (effective hourly rate of $6.00/hour).

5.1.2. Apparatus

We developed a web-based experiment framework (Figure 1). The testing area was 800x400 resolution. The bars alternated between being blue (the target) and gray. Workers were first shown a 15-second video, and then asked to complete a total of 160 target acquisition tasks. Points were awarded to workers for each successful click on a target bar, otherwise the screen flashed red on an error (a miss). The distance between the bars and the width of each bar was set appropriately for each condition. Workers used their own device and computer, which we did not control. Such a diverse setup is appropriate for crowd work, which is usually done with remote workers using their own equipment. Part of the goal of the crowd actor is to help overcome these individual differences.

5.1.3. Procedure

Prior to the task, workers were required to watch a 15 second video that demonstrated what they were to do (click the blue bar, alternating from the left to right side). During the task, the screen would flash red if an error was made. Upon success, points would be added to a scoreboard to reward workers. We did not enforce a particular error rate —4% is common in Fitts' law studies

[5]http://www.mturk.com

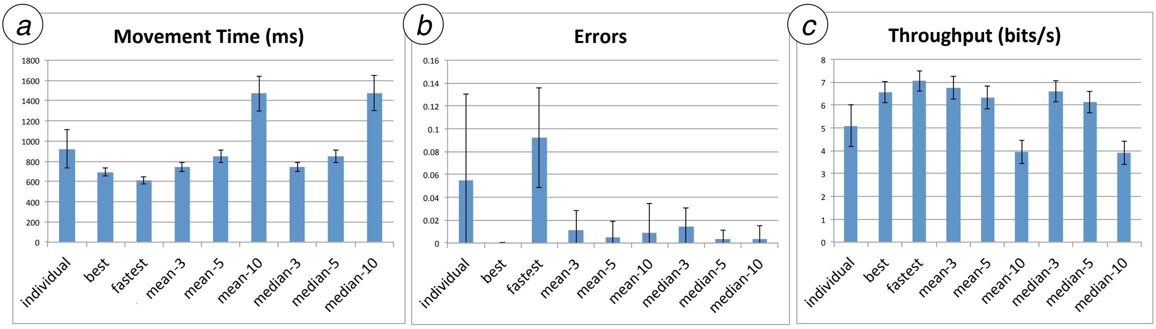

Figure 2. ***Arithmetic means of movement time, errors, and throughput for individuals and each of the 8 crowd actor personalities. Error bars are*** ± 1 ***Std Dev.***

[41] —because such adherence to protocol is difficult in crowdsourcing. We also wanted to see how our crowd actors responded to the natural error rates of the individuals in the crowd. Despite not controlling for error rate, the error rate for individual workers was near 4% at 5.50% (SD=7.56%).

Each subject performed 10 target acquisition tasks for 4 target sizes (W: 8, 16, 32, 64) $\times$ 4 target distances (A: 0, 128, 256, 512), which comprised 16 distinct Indexes of Difficulty (*ID*s) ranging from 1.58 to 6.02 bits. There were a total of $4 \times 4 \times 10 = 160$ target acquisitions per subject, and with 200 participants 32,000 target acquisitions overall. The 0 target distance does not make sense in standard pointing tasks modeled by Fitts' law, but was used here so that the random mechanism could simulate being assigned to a position right on top of the target, and was ignored for all other analysis.

5.2. Forming Crowd Actors from Individuals

We used the data to synthesize the following 8 crowd actors offline: *best*, *fastest*, *mean-of-3*, *mean-of-5*, *mean-of-10*, *median-of-3*, *median-of-5* and *median-of-10*. The random start condition was simulated by considering individuals in conditions with different distances (A). All of the participants did the same pointing tasks in the same order, and so comparable trials were found by looking at conditions with different target distances (A), the same target size (W), and the same trial number.

For each personality, we created 20 separate crowd actors with N, the number of constituent individuals forming the actors, fixed at 10. We then created an additional 20 actors for the random start mechanism. Finally, we created crowd actors for N from 2 to 10 for only the *best*, *fastest*, and *mean-of-3*. We chose these three because, as we will see in the next section, they were the best performing actors. Thus, we simulated data for $8 \times 2 \times 20 + 9 \times 20 \times 3 = 860$ different crowd actors. To create more comparable crowd actors, the constituent individuals (of 200 total) used to create each of the 20 crowd actors were chosen according to the following rule: $crowd\_actor = individual \pmod{10}$, where 10 was chosen because it was the maximum N that we explored.

We created crowd actors offline (post hoc) in order to *(i)* factor out time to route workers to tasks and time for them to understand the task, which is in keeping with experiments of motor performance (why reciprocal tasks are used), and *(ii)* to combine the data into many more crowd actors than would have been feasible to collect data for separately. Systems such as Legion [23] and WeGame [27] have shown that coordination can be solved in practice. Our experiments explore the broader potential for this approach by isolating motor performance.

5.3. **Measures and Analysis**

5.3.1. Measures

In pointing task experiments, we are primarily interested in three quantities: *(i)* movement time, *(ii)* errors, and *(iii)* throughput (which combines movement time and errors). Movement time in a reciprocal pointing task is the time between the end of the last trial and when the mouse is clicked again. A trial results in an error when the mouse is clicked outside of the target. Throughput is measured in terms of bits/second, and was calculated using the mean-of-means approach, advocated elsewhere [37, 42]. Crossman's adjustments for effective target width (W_e) and effective Index of Difficulty (ID_e) were used to control for speed-accuracy biases [9, 30].

We are primarily interested in observed differences between the crowd actors in terms of the metrics just described. We are also interested in how features unique to a crowd actor, i.e., the size of the crowd N and the mechanism used to combine their work, affects these measures.

5.3.2. Analysis

Statistical analyses were based on single-factor experimental designs. For our comparisons of crowd actor personality, *Personality* was a single-factor with 8 levels, one for each personality. For our comparison of non-random start positions to random start positions, *Random* was a single factor with 2 levels, one for the random start positions and one for standard start positions. For our analysis of how the performance of crowd actors changes with respect to the number of comprising humans, N (number of constituent individuals) was a single factor with 9 levels ranging from N=2 to N=10.

Movement time and throughput data were analyzed using standard ANOVA procedures, whereas error rate was analyzed nonparametrically using Kruskal-Wallis tests. When performing multiple pairwise comparisons, to correct for Type I errors, we used the Tukey-Kramer HSD test [19, 39] for parametric analyses, and the Steel-Dwass test [17] for nonparametric analyses.

We created a standard log file for each crowd actor of the same form as what would have resulted from an experiment with an individual. Doing so allowed us to analyze our crowd actor experiments along with the result from our constituent individuals with the widely-used FittsStudy[6][41]. Given a log file, FittsStudy computes movement time, errors, and throughput. It uses standard practices for discarding outlier trials, defined as trials in which movement distance was less than 0.5A to the target, or for which the trial endpoint was greater than 2W away from the target center [30]. It is also able to apply Crossman's (1957) [9] correction to account for varying speed-accuracy biases among participants.

[6]http://depts.washington.edu/aimgroup/proj/fittsstudy/

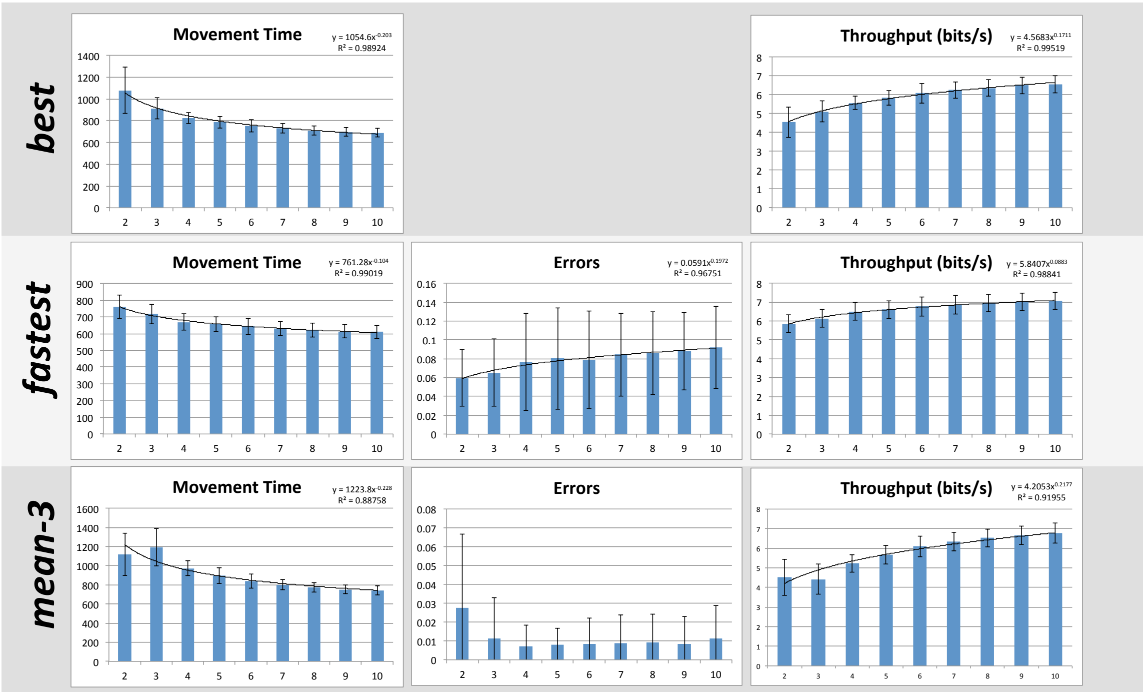

***Figure 3.** Arithmetic means of movement time, errors, and throughput as the number of constituent individuals (N) increases from 2 to 10 across the* **mean-3, fastest,** *and* **best** *personalities. Error bars are* ±1 *Std Dev.*

6. RESULTS

6.0.1. Performance of Individual Humans

On the whole, the individual participants in our study seemed to be fairly normal, which suggests that our web-based framework for capturing the trials and the crowd workers that we attracted were reasonable. For example, the submovements per pointing task were 2.12 (SD=0.44) on average, which comports exactly with the dominant movement model of [33]. The overall mean movement time was 921.4ms (SD=191.2); the overall error rate was 5.50% (SD=7.56%); and throughput was 5.09 bits/s (SD=0.91). The fit to the Fitts' law model is Pearson r=0.74 (SD=0.13). This result is non-obvious because unlike standard pointing experiments, the crowd used their own equipment from their own homes.

6.0.2. Comparing Crowd Actor Personalities

We found three groups of crowd actors that were significantly different from one another in terms of mean movement time ($F_{8,351} = 83.75$, p<.0001), as shown in Figure 2(a). The fastest group composed of the *fastest* (609.9ms, SD=38.3), *best* (690.6ms, SD=40.0), *mean-3* (742.0ms, SD=47.4), and *median-3* (742.0ms, SD=47.4) crowd actor personalities. These personalities correspond to our descriptions of perfect, hasty, and prudent personalities, although only the prudent actors utilizing the first 3 of 10 responses were part of this group. A middle group contained both the constituent individuals (921.4ms, SD=191.2) and the crowd actors with the *mean-5* (845.7ms, SD=62.6) and *median-5* (845.7ms, SD=62.6) personalities, indicating that these prudent crowd actor personalities did not offer a detectable advantage in movement time over the individuals comprising them. The slowest group contained the *mean-10* (1468.5ms, SD=173.5) and *median-10* (1474.9ms, SD=174.4) crowd actors, which was expected because these actors wait for all 10 constituent individuals to finish before deciding where to issue their click.

In terms of errors, all of the actors resulted in lower error rates as compared to the constituent individuals except for the *fastest* personality ($\chi^2_{(8)} = 174.6$, p<.0001), as shown in Figure 2(b). The mean error rate for individuals was 0.055 (SD=0.76), whereas the error rate for fastest was 0.090 (SD=0.044), which means that the crowd actors with the *fastest* personality made more errors than the constituent individuals (Steel-Dwass Z=3.99, p<.005). Interestingly, while including more constituent individuals in crowd actors with the *mean* and *median* personalities did decrease errors slightly (an observed error rate of 0.011 (SD=0.017) for the *mean-3* personality *vs.* 0.009 (SD=0.026) for the *mean-10* personality) these differences were small and not detectably different. The error rate of *best* was zero because it is defined as the theoretical upper bound.

Finally, in terms of throughput, all of the crowd actor personalities outperformed the constituent individuals (5.09 bits/s, SD=.91) with the exception of the cautious personalities (Figure 2(c)), *mean-10* (3.95 bits/s, SD=.52) and *median-10* (3.90 bits/s, SD=.50), which performed significantly worse than all other crowd actors and the individuals ($F_{8,351} = 61.67$, p<.0001). The crowd actors with the *fastest* personality seem to have done best here, but this is in part a result of how throughput is calculated. Standard practice removes outliers, as described above. Unfortunately, the *fastest* personality had a much higher outlier rate (0.018 outliers per trial, SD=0.012) compared to an outlier rate of only 0.009 (SD=0.015) for *individuals* and 0.003 (SD=0.012) for the *mean-3* personality. The *fastest* personality's outlier rate was significantly higher than all of the other actor's ($\chi^2_{(8)} = 100.1049$, p<.0001).

While *mean-3* had a slightly higher throughput (6.76 bits/s, SD=0.50) than *mean-5* (6.32 bits/s, SD=0.49), and a similar trend was observed between *median-3* (6.59 bits/s, SD=0.45) and *median-5* (6.13 bits/s, SD=0.47), this difference was not statistically significant.

Overall, we saw very little difference between the *mean* and *median* personality types, and so the remainder of our analysis focused only on *mean*. Furthermore, we found the *mean-3*, *fastest*, and *best* personalities to be most promising and so we explored other dimensions of the crowd actor for those personalities.

6.0.3. The Effect of Number of Humans N per Actor

We explored how the number of constituent individuals used to create each actor impacted the performance of the *best*, *fastest*, and *mean-3* crowd actors. As the number of constituent individuals increased from 2 to 10, the movement time decreased (Figure 3). The movement time of the crowd actors with the *best* personality decreased from a mean of 1077.5ms (SD=212.9) for $N = 2$ to 690.6ms (SD=40.0) for $N = 10$ ($F_{8,171} = 40.6$, p<.0001); decreased from a mean of 761.0ms

(SD=71.5) to 609.9ms (SD=38.3) for the *fastest* personality ($F_{8,171} = 21.6$, p<.0001); and decreased from 1117ms (SD=219.9) to 742.0 (SD=47.4) for the *mean-3* personality ($F_{8,171} = 42.1$, p<.0001).

The error rate went up slightly for the *fastest* personality from a mean error rate of 0.05 (SD=0.03) for $N = 2$ to 0.09 (SD=0.04), although the difference was not statistically significant ($\chi^2_{(8)} = 11.44$, p=.18). The very small error rates for the *mean-3* personality across all values of N were not detectably different from one another. The *best* personality had nearly zero errors: errors could occur rarely when all of the constituent individuals used to create a crowd actor with the *best* personality made an error, which is more likely when N is small.

The movement time performance of crowd actors seemed to decrease according to a power law, much like the effect long observed for practice [6], except that here – instead of practice time as the governing factor – the number of humans constituting to the crowd actor N was the cause of decreased movement time. We fit our observation of mean movement times to a power law of the form $MT = aN^b$ and found good fits as measured by R^2:

best:

$$MT = 1054.6N^{-0.203},\ R^2 = 0.9892 \tag{2}$$

fastest:

$$MT = 761.3N^{-0.104},\ R^2 = 0.9902 \tag{3}$$

mean-3:

$$MT = 1223.8N^{-0.228},\ R^2 = 0.8876 \tag{4}$$

Similarly, the performance of crowd actors as measured by throughput in bits/s seemed to increase according to a power law, as is typical for human performance in manual trials repeated over time [9]. But instead of practice time as the cause for improvement in an individual human, the number of constituent humans comprising the crowd actor was the cause of increased performance.

best:

$$TP = 4.6N^{0.1711},\ R^2 = 0.9952 \tag{5}$$

fastest:

$$TP = 5.8N^{0.0883},\ R^2 = 0.9884 \tag{6}$$

mean-3:

$$TP = 4.2N^{0.2177},\ R^2 = 0.9196 \tag{7}$$

6.0.4. The Random Start Mechanism

We explored the effect of using the random start mechanism on the performance of crowd actors created with the *best*, *fastest*, and *mean-3* personalities (Figure 4). Across the three crowd actors, the random mechanism significantly decreased movement time and increased throughput. The movement time decreased from 690.6ms (SD=40.0) to 462.8 (SD=48.7) for the *best* personality ($F_{1,38} = 261.6$, p<.0001); from 609.9ms (SD=38.3) to 309.5ms (SD=49.5) for the *fastest* personality ($F_{1,38} = 459.7$, p<.0001); and from 742.0ms (SD=47.4) to 577.7ms (SD=40.7) for the *mean-3*

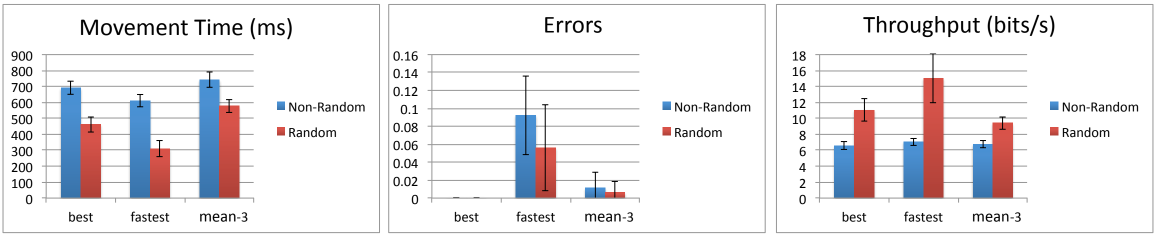

Figure 4. ***Arithmetic means of the random and non-random combination mechanisms on movement time, errors, and throughput for the*** **best, fastest,** ***and*** **mean-3** ***personalities. Error bars are*** ± 1 ***Std Dev. Note that no error graph is shown for*** **best** ***because it does not produce errors by definition.***

personality ($F_{1,38} = 138.0$, p<.0001). Throughput increased from 6.56 bits/s (SD=0.46) to 11.05 bits/s (SD=1.41) for the *best* personality ($F_{1,38} = 182.9$, p<.0001); from 7.05 bits/s (SD=0.4) to 15.05 bits/s (SD=3.10) for the *fastest* personality ($F_{1,38} = 130.44$, p<.0001); and from 6.76 bits/s (SD=0.50) to 9.42 (SD=0.81) for the *mean-3* personality.

6.0.5. The Best Individuals vs. *Crowd Actors*

One idea behind the crowd actor is that they should not only outperform their constituent individuals, but outperform what any human being could do alone. We compared the *best* individuals in our crowd of 200 to the crowd actors that we created. Individually, the best (least) individual average movement time was 602.75ms, the best individual error rate was 0.00%, and the best individual average Fitts' throughput was 7.42 bits/s. The fastest single crowd actor was *fastest-10*, at 547.00 milliseconds on average. All crowd actor personalities except *fastest-10* had at least one individual crowd actor that had 0 errors. Two of our crowd actors were better in terms of throughput: *mean-3* and *fastest-10*, which were 7.5757 and 7.8363 bits/s by comparison. *median-3* was close behind at 7.2725 bits/s. Thus, the best crowd actors were better than the best individuals, even when taking the fastest movement time, least error rates, and greatest throughputs from different individuals.

7. SYNCHRONOUS EXPERIMENT

The crowd actors presented thus far were constructed offline, which allowed the experiments to be tightly controlled. In order to better understand crowd actor performance in potential applications, we built a system to explore synchronous crowd actions in pointing on a desktop interface. This is similar to the setup in Legion [23] (Figure 5). Workers see a desktop interface and are able to click on it. The position of those clicks is sent to our server, which then combines them to determine a final click location. The latency and errors observed here are messy —a function of not only motor performance, but also recognition time, potential ambiguity of the label, and network latency.

All of the workers receive an instruction to click on the same element in the interface at the same time every 10 seconds (subject to network latency). They are rewarded according to both speed and accuracy, and can only issue one click per 10-second time period. Clicking correctly results in a fixed reward of $\$0.03 + (time/10,000) \times 0.03$, where *time* is the time after the start of the

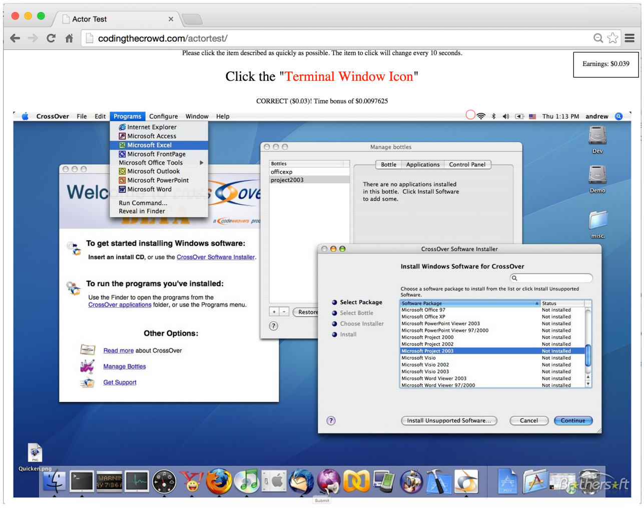

Figure 5. ***Our web-based experimental setup for synchronous crowd actor experiments. Workers were all asked to click on the same random element in their view of the desktop every 10 seconds.***

period when the server receives the worker's input, while clicking incorrectly results in a fixed $0.01 reward. This is a wage minimum of $3.60/hour; in practice, the task is reasonably easy and so our workers received closer to $15/hour.

We used LegionTools [25] to recruit a group of 10 workers and send them to the experiment web page. Once they arrived, they participated in 100 collective clicking tasks. We once again were concerned with our three actors: *best*, *fastest*, and *mean-3*. We added *median-3* when pilots suggested that workers were more likely to make large errors that could affect the mean because they didn't know where to click. We calculated the error rate and movement time as shown in the table below for each actor. We were unable to calculate throughput given that the experiment did not involve controlled target sizes, distances, or starting points. Results generally follow what we expect. *best* is slowest but generally makes few errors. The errors that are made resulted from when none of the workers clicked the correct region in the 10 second window. *mean-3* does poorly (worse in errors than *fastest*) because in these realistic tasks workers are more likely to click far from the target, although *median-3* is able to do much better.

actor	movement time	errors
fastest	1675ms (SD=1263)	0.69
median-3	3167ms (SD=1642)	0.59
mean-3	3167ms (SD=1642)	0.81
best	4322ms (SD=2377)	0.08

8. DISCUSSION

Overall, the crowd actor personalities behaved as expected. The *fastest* personality was quite fast, but the speed came at the cost of errors, as predicted by Fitts' law [41]. Personalities that make their pointing decision based on the input of multiple workers, such as the *mean* and *median* personalities, reduce errors. Interestingly, we found that combining the inputs of only the first 3 of 10 crowd workers already dramatically reduced errors while remaining fairly fast. In fact, the throughput of these actors was not detectably different from the *best* personality, suggesting they would be a good substitute for this "perfect" personality in practice. Furthermore, it suggests that in some domains, advantage in correctness may only require a few crowd workers, while performance in terms of speed may benefit from larger crowds. Personalities able to leverage the first few individuals to respond may show the best performance.

The crowd actor affords the tuning of dimensions unavailable in individual pointing tasks. For instance, the size of the crowd can be increased to improved performance, but not all crowd actor personalities leverage the availability of additional constituent individuals to the same extent as others. Personalities that take into account the input of all of the constituent individuals are necessarily slowed waiting on them, and response time is a clear function of the number of people being waited on. Fortunately, we found that personalities that determine the action to take based a small number of the first workers to act perform well, *e.g.*, *mean-3*, are able to balance accuracy with speed.

We also discovered that across our three most successful crowd actors (*best*, *fastest*, and *mean-3*), both movement time and throughput could be fit to a power law. This is important because it suggests that the performance of these actors could be predicted as a function of N. The number of constituent workers is likely to be proportional to cost. It also suggests a sort of space-time trade-off. What for an individual human performer is a matter of practice time may in fact become, for a crowd actor, a matter of "space"—that is, the number of human beings contributing to the crowd actor as N increases. It is intriguing that increasing N caused performance to follow a power law so familiar to human performance studies, but here with increasing constituent humans, not increasing the time any one human spends practicing.

Finally, as we transition from tasks being done by an individual to those being done by a crowd, we are afforded opportunities to deconstruct the problem to improve performance. The random start mechanism leverages this by allowing constituent individuals to start at arbitrary locations rather than at the same location. We found this improved performance—both lowered movement time and increased throughput. In addition to improving performance, this mechanism may be important for deploying the crowd actor in practice because it eliminates the constraint that all workers need to start at the same location or, potentially, even at the same time.

9. **FUTURE WORK**

Just as Fitts' Law has helped to define a science of human-computer interaction, it is our hope that the crowd actor model might be a start to a science for coordinated, real-time crowd work. In this paper, we have described the crowd actor model, and explored it through experimentation with crowd actors possessing different personalities and combination mechanisms. Modeling a dynamic group of people as a single crowd actor opens a number of avenues for future research.

Clearly, additional issues need to be addressed before we can be confident that the benefits of the crowd actor motor system can be extended to user interfaces. The results presented here illustrate that solving the practical issues with crowd actors are worthwhile because of the demonstrated potential for reward. That said, getting this set up to work in practice may be difficult for a variety of reasons, most notably the difficulty in coordinating a crowd around a common task. The Legion system has already demonstrated the potential for a crowd to collectively control existing interfaces [23]. Our model may help to predict expected performance in such a system given the size of the crowd or the personality chosen.

Future work may seek to separate the cognitive and motor functions of the crowd actor to enhance the abilities of an individual. For instance, working together with a crowd actor may allow a system to take tremor out of a mouse trajectory, thereby allowing someone to perform better than they otherwise could. This could be useful for people with motor impairments. A resulting challenge is to support different constituent individuals with different roles, *e.g.*, a user with a motor impairment could drive the system with others supporting.

Another avenue for future work is to explore how our model and way of describing the crowd actor may extend to other models of human performance, *e.g.*, the CRT choice reaction test[35], or the steering law[1]. Thus far, it seems that the idea of "personalities" is likely to extend across different kinds of problems, as the personalities express a fundamental trade-off between speed and accuracy seen in many areas. What seems to be more problem-specific is the method used to break up the problem. We have shown that multiple starting points are useful for pointing tasks, but other ways of breaking up problems have been found to be useful in other areas. Future work may consider the general principles that describe, demonstrate utility, and ultimately predict how to frame new problems for the crowd.

10. CONCLUSION

In this paper, we have introduced the *crowd actor* model for real-time crowdsourcing, and explored it in the context of modeling the crowd as an individual motor system. We have introduced the idea of the crowd actor *personality* for describing the trade-offs expressed in how the crowd actor is formed. We have shown how personality, crowd size, and the mechanism used to create the crowd actor affects observed performance, and suggested areas for future work that may generalize these concepts to new problems in this space. More fundamentally, we believe the crowd actor, and our modeling of its motor system, represent the beginning stages of a science to help understand, explain, and predict the performance of real-time crowdsourcing systems.

11. REFERENCES

[1] Accot, J. and Zhai, S. 1997. Beyond Fitts' law: models for trajectory-based HCI tasks. CHI 1997.

[2] Bernstein, M. S., Little, G., Miller, R. C., Hartmann, B., Ackerman, M. S., Karger, D. R., Crowell, D. and Panovich, K. Soylent: a word processor with a crowd inside. UIST 2010.

[3] Bernstein, M. S., Brandt, J. R., Miller, R. C., and Karger, D. R. 2011. Crowds in two seconds: Enabling realtime crowd-powered interfaces. UIST 2011.

[4] Bigham, J. P., Bernstein, M., and Adar, E. Human-Computer Interaction and Collective Intelligence. CI Handbook. 2014.

[5] Bigham, J. P., Jayant, C., Ji, H., Little, G., Miller, A., Miller, R. C., Miller, R., Tatarowicz, A., White, B., White, S., and Yeh, T. Vizwiz: nearly real-time answers to visual questions. UIST 2010.

[6] Card, S. K., Moran, T. P. and Newell, A. 1983. The Psychology of Human-Computer Interaction. Hillsdale, New Jersey: Lawrence Erlbaum, pp. 57–59.

[7] Chilton, L. 2009. Seaweed: A web application for designing economic games. Master's thesis, MIT.

[8] Cooper, S., Khatib, F., Treuille, A., Barbero, J., Lee, J., Beenen, M., Leaver-Fay, A., Baker, D., Popovic, Z. and Players, F. 2010. Predicting protein structures with a multiplayer online game. *Nature* 466(7307):756–760.

[9] Crossman, E. R. F. W. 1959. A theory of acquisition of speed-skill. Ergonomics 2 (2), pp. 153–166.

[10] Crossman, E. R. F. W. The speed and accuracy of simple hand movements. In *The Nature and Acquisition of Industrial Skills.* 1957.

[11] Fitts, P. M. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Pyschology* 47(6):381–391.

[12] Goldberg, K., and Chen, B. 2001. Collaborative control of robot motion: Robustness to error.

[13] Har-Peled, S., Koltun, V., and Song, D. and Goldberg, K. Efficient Algorithms for Shared Camera Control. SCG 2003.

[14] Gordon, M., Bigham, J. P. and Lasecki, W. S. Real-time captioning by groups of non-experts. In *Proceedings of the ACM Symposium on User Interface Software and Technology – Posters* (UIST 2012).

[15] Greenberg, S., and Marwood, D. Real time groupware as a distributed system: concurrency control and its effect on the interface. 1994.

[16] Kraut, R. E., Gergle, D. and Fussell, S. R. The use of visual information in shared visual spaces: informing the development of virtual co-presence. 2002.

[17] Hollander, M. and Wolfe, D. A. 1999. Distribution-free two-sided all-treatments multiple comparisons based on pairwise rankings. In *Nonparametric Statistical Methods*.

[18] J.A.S. Kelso and G. SchÃűner. 1988. Self-organization of coordinative movement patterns. Human Movement Science 7(1), pp. 27–46.

[19] Kramer, C. Y. 1956. Extension of multiple range tests to group means with unequal numbers of replications. Biometrics 12 (3), pp. 307-310.

[20] Lasecki, W. S. and Bigham, J. P. Self-Correcting Crowds. In *Proceedings of the ACM Conference on Human Factors in Computing Systems âĂŞ Works-in-Progress (CHI 2012).*

[21] Lasecki, W. S., Homan, C. and Bigham, J. P. Architecting Real-Time Crowd-Powered Systems. In *Human Computation Journal (1)(1).* 2014.

[22] Lasecki, W. S., Kim, J., Rafter, N., Sen, O., Bernstein, M. S. and Bigham, J. P. Real-time captioning by groups of non-experts. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI 2015).

[23] Lasecki, W., Murray, K., White, S., Miller, R. C. and Bigham, J. P. 2011. Real-time crowd control of existing interfaces. In *Proceedings of the ACM Symposium on User Interface Software and Technology* (UIST 2011).

[24] Lasecki, W. S., Miller, C. D., Sadilek, A., Abumoussa, A., Borrello, D., Kushalnagar, R. and Bigham, J. P. Real-time captioning by groups of non-experts. In *Proceedings of the ACM Symposium on User Interface Software and Technology* (UIST 2012).

[25] Lasecki, W. S., Mitchell, G., Koutra, D., Jung, M. F., Dow, S. P. and Bigham, J. P. Glance: Rapidly Coding Behavioral Video with the Crowd. In *Proceedings of the ACM Symposium on User Interface Software and Technology* (UIST 2014). 2014.

[26] Lasecki, W. S., Song, Y. C., Kautz, H., and Bigham, J. P. Real-time crowd labeling for deployable activity recognition. In *Proceedings of the ACM conference on Computer Supported Cooperative Work* (CSCW 2013).

[27] Loparev, A., Lasecki, W. S., Murray, K. I., and Bigham, J. P. Introducing Shared Control to Existing Video Games. In *Proceedings of the conference on the Future of Digital Gaming* 2014.

[28] MacKenzie, I.S. A note on the information-theoretic basis for FittsâĂŹ law. Journal of Motor Behavior, vol. 21, no. 3, pp. 323âĂŞ330. 1989.

[29] MacKenzie, I. Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction* 7(1):91–139. 1992.

[30] MacKenzie, I. S. and Isokoski, P. Fitts' throughput and the speed-accuracy tradeoff. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI 2008).

[31] Mao, A., Chen, Y., Gajos, K. Z., Parkes, D. C., Procaccia, A. D. and Zhang, H. 2012. Turkserver: Enabling synchronous and longitudinal online experiments. In *Proceedings of the conference on Human Computation and Crowdsourcing* (HCOMP 2012).

[32] Maynes-Aminzade, D.; Pausch, R.; and Seitz, S. 2002. Techniques for interactive audience participation. ICMI 2002.

[33] Meyer, D. E., Smith, J. E. K., Kornblum, S., Abrams, R. A. and Wright, C. E. 1990. Speed-accuracy tradeoffs in aimed movements: Toward a theory of rapid voluntary action. In Attention and Performance XIII, M. Jeannerod.

[34] Morris, M. R.; Huang, A.; Paepcke, A.; and Winograd, T. Cooperative gestures: multi-user gestural interactions for co-located groupware. 2006.

[35] Remington, Robert J. Analysis of sequential effects on choice reaction times. Journal of Experimental Psychology, Vol 82(2), Nov 1969, 250-257.

[36] Salisbury, E., Stein, S., and Ramchurn, S. Real-time Opinion Aggregation Methods for Crowd Robotics Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015).

[37] Soukoreff, R. W. and MacKenzie, I. S. 2004. Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI. International Journal of Human-Computer Studies.

[38] Thompson, C. 2006. Massively multiplayer pong.

[39] Tukey, J. W. 1953. The problem of multiple comparisons. Unpublished manuscript of 396 pages. Department of Statistics, Princeton University.

[40] von Ahn, L., and Dabbish, L. Labeling images with a computer game. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI 2004).

[41] Wobbrock, J. O., Cutrell, E., Harada, S. and MacKenzie, I. S. An error model for pointing based on fitts' law. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI 2008).

[42] Wobbrock, J. O., Shinohara, K. and Jansen, A. The effects of task dimensionality, endpoint deviation, throughput calculation, and experiment design on pointing measures and models. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI 2011).

[43] Wobbrock, J. O., Jansen, A. and Shinohara, K. Modeling and predicting pointing errors in two dimensions. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI 2011).

[44] Wobbrock, J. O., Myers, B. A. and Kembel, J. A. 2003. Edgewrite: a stylus-based text entry method designed for high accuracy and stability of motion. In *Proceedings of the ACM Symposium on User Interface Software and Technology* (UIST 2003).