Design Strategies for Efficient Access to Mobile Device Users via Amazon Mechanical Turk

Jason T. Jacques and Per Ola Kristensson
Department of Engineering
University of Cambridge
Cambridge, United Kingdom

ABSTRACT

It is often challenging to access a pool of mobile device users and instruct them to perform an interactive task. Yet such data is often vital to provide design insight at various stages of the design process of a mobile application, service or system. We propose accessing a pool of mobile device users via the microtask market Amazon Mechanical Turk (MTurk). While mobile device users are still a minority on MTurk, they provide unique opportunities for requesters. Not only does catering to mobile device users expand the potential worker pool, but also offers access to realistic *in situ* mobile sensor data for a variety of pervasive computing applications. We present a design pattern that makes it easy to crowdsource tasks to mobile device users. We demonstrate this pattern to obtain sensor availability information for some of the most popular mobile device sensors, using HTML5 web APIs to access them. We find that there is a sizable pool of crowd workers on MTurk and we present two design strategies for efficiently accessing them. The first design strategy enables direct fast access to a small pool of mobile-only device users. The second strategy exploits our finding that a large number of desktop-based crowd workers are prepared to switch to their mobile device if prompted by a well-designed mobile task. Finally we discuss solution principles for sensor-dependent mobile systems and studies that will be underpinned by mobile device crowd workers.

CCS CONCEPTS

• **Human-centered computing → Systems and tools for interaction design**; • **Hardware → Sensor applications and deployments**;

KEYWORDS

Crowdsourcing, web, mobile, devices, sensors

ACM Reference format:
Jason T. Jacques and Per Ola Kristensson. 2017. Design Strategies for Efficient Access to Mobile Device Users via Amazon Mechanical Turk. In *Proceedings of First ACM Workshop On Mobile Crowdsensing Systems And Applications, Delft, The Netherlands, November 2017 (CrowdSense '17),* 6 pages.
https://doi.org/10.475/123_4

1 INTRODUCTION

It is often beneficial to gain direct access to mobile device users and have them *interact* with a particular sensor-based task. First, such data is often required to train machine learning algorithms that handle user interaction with mobile device sensors. Second, iterative design of new mobile applications and services is reliant on easy access to mobile device users for rapid prototyping. Third, it is often desirable to micro-survey users *in situ* to gather contextual information, for instance, to improve a mobile service. However, it is difficult to achieve these objectives as existing solution strategies typically either rely on recruiting paid local volunteers or rely on designing an app and releasing it on the app store.

This paper explores the idea that it is often possible to access mobile device users via an existing microtask market in which requesters pay crowd workers to carry out a mictoask. This idea is underpinned by the fact that Amazon Mechanical Turk (MTurk) has a small, but growing, potentially untapped workforce of mobile device users. However, at present, available tasks do not offer a suitable experience for these workers. Further, the demographics, distribution, and capabilities of this potential workforce and their devices are all poorly understood. Viewing a website, such as MTurk, on a mobile device requires frequent panning and zooming of the interface. Additionally, tasks embedded in the MTurk interface frequently require a second level of window-in-window scrolling, which further exacerbates the poor user experience. This may discourage mobile users from participating in a given task, or using MTurk at all.

Besides expanding the potential workforce for existing tasks, engaging with mobile MTurk users opens up several new possibilities for research and design. First, it allows user studies and surveys on MTurk to be carried out by users in off-the-desktop *in-situ* situations. Second, by leveraging mobile devices' built-in sensors, MTurk HITs can augment study data with additional automatically-sensed contextual information. Third, by interacting with mobile MTurk users it is possible to quickly gather contextual sensor data for new mobile interaction techniques, such as detecting deice taps using accelerometer data [9]. Finally, it can also enable researchers to deploy contextual survey methodologies such as the Experience Sampling Method [4], which have been shown to reveal novel insights not gained via traditional lab-studies (e.g. [13]).

This paper offers the following practical contributions for engaging with mobile device users via a microtask market:

- We identify some of the difficulties experienced in undertaking sensor-based tasks using MTurk, and demonstrate potential mitigation techniques.
- We provide sensor availability information for some of the most popular sensors using HTML5 web APIs.

CrowdSense '17, November 2017, Delft, The Netherlands
© 2017 Copyright held by the owner/author(s).
ACM ISBN 123-4567-24-567/08/06...$15.00
https://doi.org/10.475/123_4

- We present design strategies to engage with mobile device users to either encourage desktop users to switch to their mobile device or to exclude switchers to gather data exclusively from genuine routine mobile device user.

2 BACKGROUND

Crowdsourcing has become a significant source of data for behavioral research [8], image analysis [12], and linguistic tasks [1]. With the ever increasing number of mobile devices in use, rising 526 million to 7 billion in 2013 alone and expected to reach 10.2 billion by 2018 [3], the use of mobile devices by crowd workers is inevitable. Not only do mobile devices present additional opportunities to reach the existing workforce, such as when commuting, but they also allow new classes of worker to be engaged. The majority of the MTurk workforce are US citizens [14]. As a developed market, two-thirds of Americans own a smartphone [15]. It is expected that by 2018, the majority of American mobile device connections will be using 4G technology [3], allowing an increasingly rich internet experience. This will expand the opportunities for people to undertake casual work in the form of crowdsourced tasks.

Modern mobile devices are highly capable, with fast and efficient JavaScript interpreters and offer functionality such as maintaining state, local storage APIs, and client-side support for efficient compression and decompression of data. Even where desktop computing is available, modern smartphones, tablets, and other mobile devices can offer a richer computing environment for both the worker and requester. The plethora of sensors available in mobile devices—such as cameras, microphones, touch screens, accelerometers, and more—offer the opportunity for requesters to introduce creative interfaces, improve interactivity, and explore new avenues for research.

Where users can be encouraged to install dedicated application, this low-level access to the device hardware may offer additional interaction and data gathering opportunities. However, not all crowdsourced data collection can be “gamified” in this way and developing apps and reaching a critical mass of users requires a substantial investment of time and effort. As the app market has developed, the difficulty of attracting enough users to download and install an app has become an intractable problem [18]. Further, there has been some concern about the ethics of large-scale “app store” distribution of instrumented research applications [10], even where successful. When a remote end-user downloads a free “game” or application, as with commercial software, they may not read or understand the terms being imposed on them [6].

While dedicated applications may have advantages with regard to gathering *in-situ* data with potentially more direct access to hardware, web-based systems, such as MTurk, continue to predominate large-scale crowdsourcing investigations. The ubiquity of web-browsers, preinstalled on modern smartphones, tablets, and other mobile devices, creates a large user-base with ready uptake. Platforms such as MTurk make on-demand access to this ready and waiting workforce fast and easy.

Unlike apps, which are often launched for just minutes, or even seconds, at a time [17], limiting their opportunity to gather substantial data, crowd work platforms create a more formal relationship, where workers are required to carryout a given activity and compensation is offered for their work. Using crowdsourcing platforms such as MTurk and paying workers sets expectations and ensures that each participant will provide a minimum data contribution.

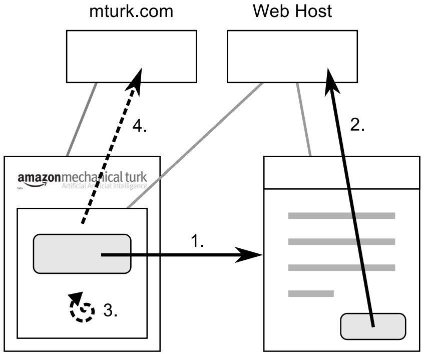

Figure 1: Task setup design pattern for a mobile device optimized task: 1. Worker clicks oversized button in task frame to open mobile-optimized task; 2. Worker completes task and submits page to web host; 3. Task frame polls for task completion; 4. Task frame redirected to the `mturk.com` submission URL, ready for approval and payment.

As web services have pushed the web itself forward as a platform, more and more technologies are accessible through the browser. For many tasks, modern web standards provide access to many of the same input sources as dedicated applications and frameworks. These new and evolving web-standards mean that an increasing number of sensors can be accessed through web-based platforms, including MTurk, with no app to install on the workers’ devices. Even where permanent access to participants’ devices is desirable, the use of web-based environments, such as MTurk, to prototype interactive systems has been recognized for some time [16]. These developments allow zero-install sensor based tasks to be rapidly prototyped and launched.

3 DESIGN PATTERN FOR MOBILE TASKS

Historically, MTurk has poorly supported mobile device users. Recently, however, Amazon has offered a new “beta” site http://worker.mturk.com/ which offers a more responsive UI suitable for smaller screens. Despite this change, while the newer search interface offers an improved experience in finding tasks, embedded tasks are still not typically optimised for mobile device users.

To support mobile sensor-based tasks on MTurk, we developed a design pattern that supports minimizing the time mobile device users spend using the desktop-optimised Amazon MTurk interfaces. Similar to previous work such as *CrowdStudy* [11], the approach described below and visualised in Figure 1 minimizes interference with the task itself, however our technique does not depend on a particular framework or complex client/server infrastructure. This hands-off technique reduces constraints, allowing any web based task to be used, and introduces little overhead for the on the task administrator and developer.

Our tasks, when accessed from a mobile device and embedded in the MTurk interface, present a large, easy to tap, button to open the task in a new window (Figure 1.1). This allows a fully mobile-optimized page to be presented, outside of the confines of MTurk. On completion of the task, a marker is set indicating completion of the task (Figure 1.2) and the page is automatically closed. This marker is polled periodically by the embedded window (Figure 1.3) which verifies completion and automatically submits the task to Amazon for approval and payment (Figure 1.4). This automated submission removes the common requirement to copy-and-paste a completion code from one page to another, further reducing the non-task related burdens on the worker.

Amazon does not offer a mechanism to specify that workers accepting a task should be mobile device users, and the MTurk qualification system is limited to statically assigned values. As users may, for example, switch between using desktop computers at home to mobile devices on the move, this limitation is significant. To verify that a mobile device is currently being used requires dynamic real-time detection, and can only be carried out at preview time when users first engage with the task. To detect mobile devices, we parse the browser user agent string. However, for tasks dependent on specific sensors, access should also be verified.

For tasks which are exclusively for mobile device users, workers accessing the task from their desktop computer are shown a page explaining that they are ineligible for the task. Depending on the needs of the requester this page may disclose the mobile nature of the task. Identifying the task as mobile may encourage users to switch to a mobile device. Where *in situ*, "real" mobile device users are required, requester may choose not to explain the qualification to discourage such behavior. These alternative interfaces are readily hidden, exposed, and modified using basic styling techniques.

4 THE MOBILE CROWD WORKER POOLS

Our first objective was to test our design pattern and identify if we can efficiently access a pool of crowd workers on MTurk that are willing to carry out a sensor-dependent mobile task.

We recruited 200 unique participants using MTurk. Participants were paid $0.10 (USD) for completing the task. Workers used their own mobile device, and the browser user agent string was verified and recorded as part of the detection and acceptance process.

The tasks was listed on MTurk in 25 batches of 8, one after another. Workers were prevented from participating in later batches by identifying duplicate worker IDs, asking workers to "return the HIT." This approach improves the likelihood of the task showing near the top of the available tasks lists [2].

The task collected sensor data from a variety of HTML5 APIs that might be available on a mobile device: ambient light, battery, device motion (typically accelerometers and gyroscope), orientation (typically compass), and network information. We intentionally avoided collecting audio-visual data to minimize any privacy concerns for the participants, however for activities that require multimedia sensors, camera and audio input devices are expected to be almost universal on modern smartphones, tablets, and mobile devices. After automatic data collection, the task required users to answer five questions regarding:

1. Age (18-24, 25-34, 35-44, 45-54, 55-64, 65+)
2. Gender (Male, Female, Other)
3. How many hours per week do you spend working on Mechanical Turk?
   - Less than 1, 1-5, 6-10, 11-15, 16-20, 21-25, 26-30, 31-35, More than 35.
4. What percentage of that time is spent using a mobile device to complete Mechanical Turk tasks?
   - 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100
5. Why do you carry out tasks using a mobile device?
   - I only own a mobile device.
   - I am working away from my desktop or laptop computer.
   - I find it convenient to work with my mobile device.
   - Other

In addition we asked users to provide their location via the HTML5 geolocation API (typically using A-GPS). We elected not to make geolocation automatic, and users were asked to click a button to initiate the device geolocation request. Geolocation can be denied using either a prearranged denial at OS- or browser-level, or interactively with the OS- or browser-provided dialogue. While we required users to activate the geolocation request to proceed, users declining to provide location information were permitted to submit the task and approved for payment.

Finally, we provided a space for optional open comments. When developing new and unusual tasks, which may have complex dependencies, offering the workforce the ability to comment on a task may provide additional insights or indicate problematic design decisions that may need revisiting.

We found that worker demographics were evenly split between male and female (105 male, 95 female) and skewed towards a younger population. This data is summarized in Figure 2.

The conversion rate, the ratio of exposed workers who went on to complete the task [7], was 9.15% (200 completions, 2,185 unique previews). This was higher than expected as only 7% of workers reported exclusively using a mobile device to carry out tasks.

Interestingly, 28.5% accepted the task from their desktop device and switched to their mobile device to "continue" the task. This process reserves the task for the user and makes it much easier to find later. While it seems likely that other users first encountered the task on a desktop and then switched to their mobile device before accepting the task, identification and quantification is hampered by MTurk previews being anonymous. In addition, the IP address used to access the task may well be different or transient due to the multihomed nature of mobile devices.

Sensor capabilities varied among participants' mobile devices. All targeted devices supported touchscreen input. While 93% of devices included some support for sensing device motion events via the Device Orientation Event Specification[1], only 79.5% provided values for all sensor data properties that we probed. Device orientation was provided by 85.5% of devices, although two devices provided only zero values. Device orientation was reported to be absolute, relative to true North, in 62% of the devices supporting these events. The battery status API was supported by 50% of devices, 21 devices were currently charging while 79 indicated that they were not. The mean charge of the reporting devices was 62.86%. Network type information was provided by 48.5% of devices with 82 devices indicating 'wifi', 14 indicating 'cellular', and 1 curiously indicating 'none'. Five devices reported the ambient light level, and just one reported a value for the proximity sensor.

[1] https://www.w3.org/TR/orientation-event/

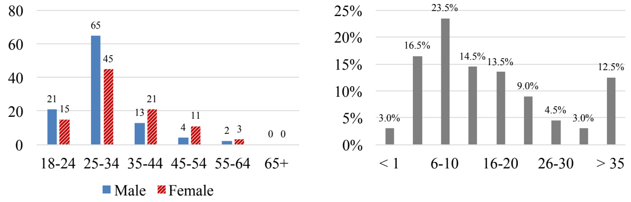

Figure 2: Left: Age and gender of respondents; Right: Time spent working on MTurk.

For the 71% of participants providing geolocation API access, 81% were from the United States, 15.5% from India, and 3.5% from elsewhere. For those that did not allow access, IP-based geolocation indicates that 62% were from the United States, 20.7% from India, and 17.3% from elsewhere. Similarly for workers who previewed the task but did not accept, IP-based geolocation indicates that 57.7% were from the United States, 27.4% were from India, and 14.9% from elsewhere. This difference in willingness to provide potentially specific location data may hint at cultural differences or concerns regarding biases.

The reported number of hours spent using MTurk per week peaked at 6–10 hours, with a second peak for workers using the platform "full-time" (greater than 35: 12.5%). This data is summarized in Figure 2.

Mobile device usage was skewed to the lower values with 17% indicating that 0% best represented their mobile device usage for MTurk tasks. The majority of workers, 51.5%, indicated that 10% of their MTurk tasks were completed on a mobile device, and less than one-third (31.5%) of workers indicated that they used their mobile device for 20% or more of their MTurk tasks. The majority of participants (57.5%) indicated using their mobile device for MTurk tasks when away from their computer, 26% indicated that using a mobile device was more convenient, 7% stated they only owned a mobile device, and 9.5% chose "Other" as their reason.

While the majority of the the 48 optional and undirected comments received, were incidental (e.g. "none", "n/a"), 10 workers indicated that they use their mobile device when away from the computer. However, another seven workers indicated a desire for more tasks to be mobile compatible, and a further three indicated that they regularly used their mobile device to complete tasks.

5 DESIGN STRATEGY 1: DIRECT ACCESS TO MOBILE-ONLY CROWD WORKERS

Our survey confirmed that we are able to access mobile device users on MTurk. However, it also revealed that the majority of crowd workers temporarily switched to their mobile device to carry out the task. Sometimes it is desirable to locate the subpool of crowd workers that *only* use a mobile device. This may for instance be beneficial when the majority of crowd workers are expected to be in an off-the-desktop context. We therefore devised a design strategy for accessing this subpool of workers.

To determine the viability of recruiting only regular mobile device users, who were not switching device on request, we carried out a follow-up to our first ask. While the task as carried out by the worker was the same in all regards, the setup was modified slightly: workers visiting the task from a non-mobile device where informed that they were ineligible to participate. However, for this task, the reason why users where asked to return the task was not disclosed. This approach was designed to dissuade otherwise-interested desktop workers from switching to a mobile device.

We recruited 56 unique participants using MTurk over three days. The tasks was listed on MTurk in 7 batches of 8. Workers were prevented from participating in later batches by identifying duplicate worker IDs. Participants were paid $0.20 (USD) for completing the task.

We found that the conversion rate for this task was 2.88% (56 completions, 1,946 unique previews). This confirmed our expectation that the conversation rate would be low as the subpool of exclusive mobile device users is a small subset of MTurk workers.

All participants first previewed, accepted, and went on submit from their mobile device. This ability to exclusively recruit mobile device users may be significant for some requesters and tasks dependent on an *in situ* workforce.

The low conversion rate may be partially explained as a consequence of the geolocation request. Users who routinely use their mobile device may be more aware of the privacy implications of allowing a geolocation data to be recorded. Only 48% of users provided geolocation data, compared with 71% in Task 1. Despite all workers electing to use their mobile device to search for, find, and complete our task, only 12.5% reported *only* having a mobile device. 37.5% of participants reported using their mobile device only when away from their computer, with the remaining 50% favouring their mobile device for the convenience.

Further, just 9 participants (16%) accepted the task from the new mobile-friendly MTurk interface. At the time of writing Amazon does not automatically redirect mobile device users to the new search interface, and this may impact this surprisingly low figure.

The study confirmed we can access a subpool of mobile device users *in-situ* via MTurk. As expected, this subpool of crowd workers is relatively small. Nonetheless, within just three days we successfully recruited 56 crowd workers, which demonstrate that it is viable to crowdsource a task to a reasonably large number of participants and receive results under relatively tight time constraints.

6 DESIGN STRATEGY 2: ENCOURAGING SWITCHING TO A MOBILE DEVICE

For many tasks, the fact that users switch device does not impact the usefulness of the user interactions or sensor data recorded. Such tasks include collecting machine learning data for certain interactive tasks (such as gestures or device rotations) and for iterative prototyping of mobile apps and services.

We noted the readiness of workers to switch devices in our first task and attempted to encourage this behavior. We carried out a follow-up survey to offer additional insights and help to better understand this phenomenon, as well as identify likely real-world location where workers may be able to offer *in-situ* data.

We recruited 104 unique participants using MTurk. The task was listed on MTurk in 13 batches of 8. Workers were prevented from participating in later batches by identifying duplicate worker IDs. Participants were paid $0.20 (USD) for completing the task. For this task, we added a static preview image to the error page shown to desktop users. This allowed workers to preview the task and evaluate their interest without committing to their mobile device at this stage. However, it is important to note that this error page included an explicit instruction to "return the HIT." Workers who chose to transition to a mobile device after accepting were actively circumventing this direction to participate in this mobile-only task.

The task asked participants to answer four questions about their device usage, and to provide context to their current location. Workers were asked to specify the device they used to find the task, how easy it was to switch device, how likely they would be to switch again in future, and an identifier for their location: home, workplace, school/college/university, internet cafe, library, or other. Participants were also offered the opportunity to provide additional comments on the task.

The conversion rate for this task was 25.06% (104 completions, 415 unique previews).

82.7% accepted the task from their desktop device and switched to their mobile device to continue the task. This higher rate may reflect the additional appeal of a preview image for the task, encouraging workers to "lock in" and transition to their mobile device.

All participants, except one, completed the location question. 84.6% of workers classified themselves as at "home," 12.5% at their "workplace," 1% at a "library," and one worker (1%) chose "other" clarifying that they were in a "hotel room." 101 participants identified their initial device as follows: 34.6% desktop; 26.9% laptop (no touch); 5.8% laptop (with touch); 22.1% phone; 7.7% tablet. This offers an incongruous contrast to data collected from the user agent strings. 35.6% of workers would have been eligible to immediately engage with the task, having claimed to have found the task using a touch capable device, as noted, however 82.7% of participants switched devices according to user agent string data.

Participants were asked to rate their experience of switching device on a five point scale between 1 (very easy) and 5 (very difficult). The mean response was 1.84, with a mode of 1 selected by 48.1% of participants. 6.7% of workers indicated that they did not switch device, however user agent string data indicates that three of these seven workers did in fact switch. Together, comprising over 50% of participants, these switching workers highlight the readiness and ease of transitioning to mobile optimized tasks, even when explicitly instructed to return the task instead.

Further, participants were asked to specify how likely they were to switch device in future on a five point scale between 1 (very unlikely) and 5 (very likely). The mean response was 4.27, with a mode of 5 selected by 58.7% of participants. These results show a very high readiness to switch device on request, and suggest that workers would be willing to switch device should a task depend on sensors or an environment not available on a typical desktop or laptop computer. These responses are summarized in Figure 3.

7 DISCUSSION

Our work highlights the potential of web-based microtasks to support successful interactive mobile crowdsensing applications. Using MTurk for mobile tasks makes it potentially much faster and easier to prototype, evaluate and collect data for applications that rely on mobile device sensors.

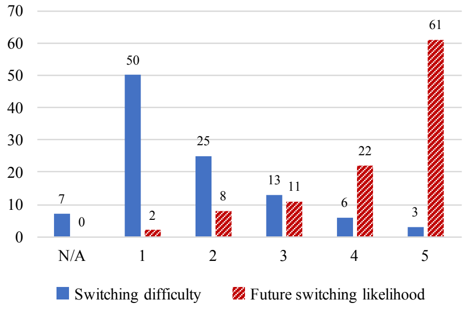

Figure 3: Self reported switching difficulty (1: very easy; 5: very difficult) and future switching likelihood (1: very unlikely; 5: very likely) of participants in Task 3. Participants indicated that they did not switch device by selecting "N/A."

We found that 7% of crowd workers only use a mobile device to access MTurk and that workers on non-mobile devices showed surprising readiness to change device when requested.

The ability to get users to switch opens additional avenues for data collection. Mobile workers have at their disposal compact and powerful devices, packed with a multitude of sensors. Task 1 demonstrated the wide range of data available through modern HTML 5 APIs. The variety of the available data will only expand as more and more devices incorporate these sensors and implement the emerging web standards. Web-based data collection through platforms such as MTurk with mobile device users offers access to potentially larger and more diverse datasets than what can be collected in typical lab settings. The first task we carried out alone collected data from 139 unique device configurations (device, operating system version, and web browser).

7.1 Solution Principles

Based on our evaluation of the basic MTurk interface and the three tasks we explored in this paper we suggest the following solution principles for using MTurk in mobile systems and studies:

Fast Collection of Sensor-Based User Interface Data Our first ask demonstrated a wide range of sensor APIs accessible via MTurk, and our third task revealed crowd workers' readiness to switch to a mobile device, if requested. In conjunction, these findings pave the way for designers and researchers to crowdsource large collections of sampling points from a variety of mobile device sensors. This is critical when developing new user interface techniques underpinned by machine learning algorithms relying on large amounts of sensor data to train on.

Switching from Desktop to Mobile Workers demonstrate a surprising readiness to change from a desktop computer to a mobile device. As shown by the demographic data collected with our tasks, this allows designers and researchers to quickly and easily reach a diverse set of users and mobile devices, which should improve the quality and representativeness of the collected user interaction data.

Mobile-Only Crowd Worker Pool Our first task discovered that 7% of workers only use a mobile device to access

MTurk. This worker pool enables MTurk to be used for *in situ* studies of mobile users. We also caution against naïvely sampling mobile crowd workers, as we have found that many crowd workers are ready to switch from their desktop to their mobile device to carry out a task. A study relying on *in-situ* mobile crowd workers therefore needs to be careful to access the correct worker pool. Crucially, for tasks which are sensitive to genuine *in-situ* data, we demonstrate techniques to limit this workforce to routine mobile device users.

7.2 Potential Applications and Future Work

Our work has show that mobile workers using MTurk can be thought of in two distinct groups, routine mobile device users and potential mobile device users. For those workers willing to switch between devices this opens new avenues of research. For many tasks, simply having ready access to rich data from the sensors typically available in modern mobile devices may be sufficient. Touch screen gesture tasks are just one example of an activity which cannot be practically or accurately carried out using a typical desktop. Improving the touchscreen experience is especially important for tasks such as text entry [13]. The ability to generate interactive prototypes, which can then be used without installation, removes a significant barrier to entry. Sensor based tasks can be deployed to numerous different devices running various underlying software without the overhead of explicitly porting to various platforms and devices.

Exclusively targeting only routine users would be detrimental to uptake. However, for some tasks capturing data *in-situ* may have benefits or be required. For example, Das et al. [5] demonstrating their work on *PRISM* consider how *in-situ* collection can be used for citizen journalism. Verifying that a worker is in place to report on an event is crucial to citizen journalism. Equally, the road bump monitoring carried out by *PRISM* could easily be prototyped using our design pattern. Combining the GPS and the device motion data demonstrated in Task 1 could be used to verify location, speed, direction, and monitor sharp vertical movements. While we chose not to *require* geolocation data in our first task, it would be trivial to do so. Task 1 demonstrates that many users are ready to provide geolocation data on request. Workers who declined to provide geolocation data could then be pre-filtered from the task.

8 CONCLUSIONS

A long-standing problem when designing pervasive and ubiquitous sensor-based interaction techniques is the efficient collection of representative user data. In this paper we demonstrate simple techniques to support rapid prototyping, evaluation, and data collection for sensor-based tasks using Amazon Mechanical Turk. We provided sensor availability information for some of the most popular sensors and show how they can be accessed via HTML5 web APIs on MTurk. We have shown that there is a sizable pool of crowd workers on MTurk and we have presented two design strategies for efficiently accessing them.

ACKNOWLEDGMENTS

This work was supported by EPSRC grant EP/N010558/1.

REFERENCES

[1] Chris Callison-Burch and Mark Dredze. 2010. Creating speech and language data with Amazon's Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (CSLDAMT '10)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1–12. http://dl.acm.org/citation.cfm?id=1866696.1866697

[2] Lydia B. Chilton, John J. Horton, Robert C. Miller, and Shiri Azenkot. 2010. Task search in a human computation market. In *Proceedings of the ACM SIGKDD Workshop on Human Computation (HCOMP '10)*. ACM, New York, NY, USA, 1–9. https://doi.org/10.1145/1837885.1837889

[3] Cisco. 2014. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013-2018*. Technical Report. http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html

[4] Sunny Consolvo and Miriam Walker. 2003. Using the experience sampling method to evaluate ubicomp applications. *IEEE Pervasive Computing* 2 (2003), 24–31.

[5] Tathagata Das, Prashanth Mohan, Venkata N. Padmanabhan, Ramachandran Ramjee, and Asankhaya Sharma. 2010. PRISM: Platform for Remote Sensing Using Smartphones. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys '10)*. ACM, New York, NY, USA, 63–76. https://doi.org/10.1145/1814433.1814442

[6] Denzil Ferreira, Vassilis Kostakos, Alastair R. Beresford, Janne Lindqvist, and Anind K. Dey. 2015. Securacy: an empirical investigation of Android applications' network usage, privacy and security. ACM Press, 1–11. https://doi.org/10.1145/2766498.2766506

[7] Jason T. Jacques and Per Ola Kristensson. 2013. Crowdsourcing a HIT: Measuring Workers' Pre-Task Interactions on Microtask Markets. In *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2013)*. AAAI Press, 86–93. http://www.aaai.org/ocs/index.php/HCOMP/HCOMP13/paper/view/7540

[8] Winter Mason and Siddharth Suri. 2011. Conducting behavioral research on Amazon's Mechanical Turk. *Behavior Research Methods* 44, 1 (June 2011), 1–23. https://doi.org/10.3758/s13428-011-0124-6

[9] William McGrath and Yang Li. 2014. Detecting tapping motion on the side of mobile devices by probabilistically combining hand postures. ACM Press, 215–219. https://doi.org/10.1145/2642918.2647363

[10] Alistair Morrison, Donald McMillan, and Matthew Chalmers. 2014. Improving Consent in Large Scale Mobile HCI Through Personalised Representations of Data. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational (NordiCHI '14)*. ACM, New York, NY, USA, 471–480. https://doi.org/10.1145/2639189.2639239

[11] Michael Nebeling, Maximilian Speicher, and Moira C. Norrie. 2013. CrowdStudy: General Toolkit for Crowdsourced Evaluation of Web Interfaces. In *Proceedings of the 5th ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS '13)*. ACM, New York, NY, USA, 255–264. https://doi.org/10.1145/2494603.2480303

[12] Cyrus Rashtchian, Peter Young, Micah Hodosh, and Julia Hockenmaier. 2010. Collecting image annotations using Amazon's Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (CSLDAMT '10)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 139–147. http://dl.acm.org/citation.cfm?id=1866696.1866717

[13] Shyam Reyal, Shumin Zhai, and Per Ola Kristensson. 2015. Performance and User Experience of Touchscreen and Gesture Keyboards in a Lab Setting and in the Wild. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 679–688.

[14] Joel Ross, Lilly Irani, M. Six Silberman, Andrew Zaldivar, and Bill Tomlinson. 2010. Who are the crowdworkers?: shifting demographics in mechanical turk. In *Extended Abstracts of the 28th ACM Conference on Human Factors in Computing Systems (CHI EA '10)*. ACM, New York, NY, USA, 2863–2872. https://doi.org/10.1145/1753846.1753873

[15] Aaron Smith and Dana Page. 2015. *U.S. Smartphone Use in 2015*. Technical Report. Pew Research Center. http://www.pewinternet.org/2015/04/01/us-smartphone-use-in-2015/

[16] Jaya Vaidyanathan, Jason E. Robbins, and David F. Redmiles. 1999. Using HTML to Create Early Prototypes. In *CHI '99 Extended Abstracts on Human Factors in Computing Systems (CHI EA '99)*. ACM, New York, NY, USA, 232–233. https://doi.org/10.1145/632716.632860

[17] Tingxin Yan, David Chu, Deepak Ganesan, Aman Kansal, and Jie Liu. 2012. Fast app launching for mobile devices using predictive user context. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 113–126. http://dl.acm.org/citation.cfm?id=2307648

[18] Nan Zhong and Florian Michahelles. 2013. Google Play Is Not a Long Tail Market: An Empirical Analysis of App Adoption on the Google Play App Market. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing (SAC '13)*. ACM, New York, NY, USA, 499–504. https://doi.org/10.1145/2480362.2480460