

# Five design challenges for human computation

Stuart Reeves and Scott Sherwood  
Department of Computing Science  
University of Glasgow, UK  
{stuartr, sherwood}@dcs.gla.ac.uk

## ABSTRACT

Human computation systems, which draw upon human competencies in order to solve hard computational problems, represent a growing interest within HCI. Despite the numerous technical demonstrations of human computation systems, however, there are few design guidelines or frameworks for researchers or practitioners to draw upon when constructing such a system. Based upon findings from our own human computation system, and drawing upon those published within HCI, and from other scientific and engineering literatures, as well as systems deployed commercially, we offer a framework of five challenging issues of relevance to designers of systems with human computation elements: designing the motivation of participants in the human computation system and sustaining their engagement; orienting participants, framing and orienting participants; using situatedness as a driver for content generation; considering the organisation of human and machine roles in human computation systems; and reconsidering the way in which computational analogies are applied to the design space of human computation.

## Keywords

Human computation, design framework, games with a purpose, citizen science, crowdsourcing.

## ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

## INTRODUCTION

Recently within HCI, a strand of research has developed concerned with the ways in which we might harness human ‘computational power’ for the purposes of solving difficult computational problems. Most prominently this work has manifested itself within HCI via the so-called ‘games with a purpose’ (GWAP) genre (e.g., [28, 30, 31, 13, 9]), in which hard problems for computation (e.g., image labelling) are distributed to humans in the form of competitive games.

This work has a wider significance. Human activity has been envisaged, particularly within networked and ubiquitous systems (e.g., the internet, ad-hoc networks, etc.), as

offering vast resources of computation. In particular, it is this potential for solving hard computational tasks that has drawn researchers to designing systems which involve human activity as an integral part of their operation. These attempts have been characterised as ‘human computation’ systems, employing humans as “processing nodes for problems that computers cannot yet solve” [29], while others have likened this opportunity to a dynamically available “remote server rackspace” of “distributed human brainpower” [34]. Motivating this trend is the knowledge that, in theory, interactive computation provides greater computational power than non-interactive algorithmic systems [33].

Existing literature within HCI includes many impressive demonstrations of ways that human activity, coupled with computational processes, may be employed to solve varied problems, such as image recognition [28, 31] and audio tagging [17]. As yet, however, there is little research that offers generally applicable recommendations or design frameworks that might assist us in realising this potential power.

Furthermore, there are numerous systems outside the domain of HCI that draw upon similar strategies to human computation. So, when we consider more general design frameworks for human computation, it appears sensible to examine the larger domain of systems that involve complementary human-machine relationships (‘human-based computation’), in which computer systems become partners in interactive processes [11]. For example, we have seen the spread of systems employing similar techniques to human computation in scientific domains outside computer science. ‘Citizen science’ projects have employed human computation methods, particularly for processing large data sets. There are also elements of human computation concepts within popular notions of ‘crowdsourcing’ [14] and the ‘wisdom of crowds’ [25], as well as increasing numbers of mainstream and commercial crowdsourcing ventures. Work done in HCI also links to other disciplines within computer science, and interest has been generated in information retrieval, natural language processing, genetic and evolutionary computation, and artificial intelligence communities.

As a contribution to this growing literature, we develop five core challenges facing designers of human computation systems. Before presenting these distinct challenges, we review in more detail our own human computation game, ‘EyeSpy’, and the range of human computation systems within the literature. This literature, coupled with our own

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*NordiCHI 2010*, October 16–20, 2010, Reykjavik, Iceland.  
Copyright 2010 ACM ISBN: 978-1-60558-934-3...\$5.00.

human computation system design experiences, inform the design challenges.

## HUMAN COMPUTATION SYSTEMS

In this section we review various human computation systems within HCI, as well as our own mobile human computation system [1]. We shall also examine other system examples with similarities to human computation, and begin to piece together a sense of the boundaries of the design space.

### Games with a purpose

Human computation ‘games with a purpose’ within HCI typically involve simple game mechanics in order to produce fun and enjoyable activities for players. As a by-product of player activities, these systems generate useful data for other tasks. The ESP Game epitomises this [28]. Players are paired online via a website, and type relevant descriptions for a given image which, if matched with the other player’s keywords, score the players points. This activity results in the rapid collection of annotations for large numbers of images. In its design, the ESP Game addresses the hard computational problem of generating meta-data about large quantities of images in order to make them searchable.

Various other games with a purpose have been devised in the style of the ESP Game, such as Verbosity [30] and Matchin [13] (like the ESP Game, both are presented as web-based games). Verbosity attempts to collect a database of ‘commonsense’ statements within the context of a game structure similar to that of the ESP Game. The purpose of Verbosity is the reuse of these commonsense statements within AI applications. Matchin on the other hand involves each user selecting an image they think the player they are paired with will prefer. If both players match in their predictions about what the other player will prefer, then the players both score points. Games produced within this genre also involve the categorisation of other media than images. TagATune [17], for example, involves categorisation of audio clips, where players work to commonly agree upon textual descriptions for a section of music that is played to both of them at the same time.

von Ahn and Dabbish also characterise GWAP systems as follows: “output agreement”, where players must share the same input, and must match their outputs whilst being unable to see one another’s outputs; “input agreement”, where players receive different or identical inputs and, by sharing one another’s outputs, must determine between them whether those inputs are indeed different or the same; and finally, “inversion-problem” games, where one player receives an input, and the other player must determine this input, based upon the first player’s output [29].

It was within the context of human computation games that we designed our own system, EyeSpy [1]. EyeSpy is a simple mobile-based multiplayer game that generates photos and text labels for geographic locations. In order to test the game we recruited 18 participants who were each given a phone with the game software. The game itself ran for two

‘rounds’ lasting 3 weeks in total. Participants’ interactions with the game were extensively logged. After the game’s end we interviewed all participants, the transcripts of which were qualitatively analysed for key themes of player experience.

As a by-product of players’ activities, EyeSpy collects a high quality set of photos that are useful for navigation on foot around urban areas. In the game, players ‘tagged’ landmarks and other prominent urban features within their local environment. Other players then physically located and confirmed these ‘tags’ (see Figure 1) for which they earned points for themselves and the players who created the tags. Participants generated 257 georeferenced photo tags and 197 georeferenced text tags over the course of the trial. Through engaging in the game, players generated a set of photographic tags that consisted of varied pictures of buildings, streets, monuments, signs, and other objects in the environment. Crucially, in order to be successful within the game, players needed to ‘design’ tags of high quality for the purposes of play, i.e., to be very findable and recognisable images, that other players could rapidly go to the locations in question and align themselves so as to successfully confirm the tags. In this way, our study found that players demonstrated a concern for the navigational experiences of one another [1]. A follow-on experiment showed that the photo set’s navigational qualities, designed into them by players, significantly assisted their secondary use as navigational aids during a simple route finding test (subjects had to locate images from the photo sets, and corresponding geolocated images drawn from Flickr) [1].

Thus, as a by-product of player activities, EyeSpy offers one solution to the difficult problem of selecting ‘good’ images to assist navigation. The interaction of players, and their activities with and around the game, can thus be seen as a way in which to ‘process’ the physical urban environment and extract suitable imagery that is navigation-ready. Whilst such a task may well be partially possible via existing machine-based algorithms for processing image databases (e.g., extracting ‘good’ navigational images from Google Streetview or Flickr’s geotagged image sets), the human work employed within EyeSpy permits the designer to leverage local knowledge and player orientation to ‘what anyone knows’ regarding the navigational features of the

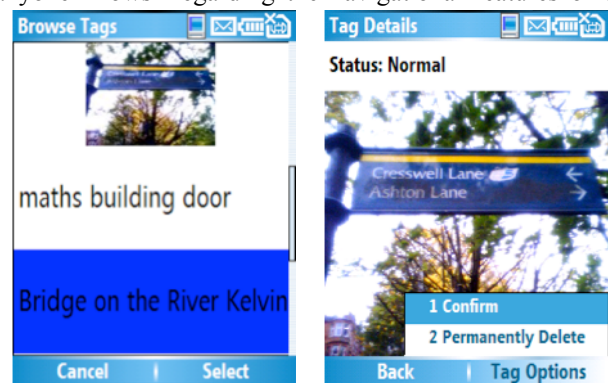


Figure 1: Browsing available tags (l); confirming a tag (r).

local area in which the game was played. This particular topic will be returned to in greater detail in our set of challenges for human computation systems designers.

### Citizen science

Games with a purpose are only one part of the range of systems that employ human computation. For example, there are increasingly many websites and applications that recruit members of the public in order to assist the processing of scientific data. This distributed ‘citizen science’ technique enables the analysis of large data sets that would be intractable via machine computation.

One of the most interesting examples of citizen science, Galaxy Zoo, is a website that invites users to engage in the recognition of galaxy types drawn from image data generated by the Sloan Digital Sky Survey [8]. Users voluntarily classify images according to specific attributes such as the number of spirals, general shape (e.g., ‘cigar’ shaped), and any unusual features that might be visible for a given galaxy (see Figure 2, left). Recognition of particular types may involve recourse to the community of Galaxy Zoo users, where users negotiate an agreed classification of images and develop common orientations to the methods of classification (i.e., develop competence). In this and other ways, Galaxy Zoo users may engage in the community of others via forums on the website. Findings of users are highlighted to the community via “object of the day” postings and so on.

Other groups of scientists with large data sets have taken a similar approach. Stardust@home [23], for instance, draws on visitors of its website to engage in the detection of interstellar dust particles embedded in the aerogel collector on NASA’s Stardust spacecraft (see Figure 2, right). For various reasons, these dust particles are very hard to detect automatically, and so analysis is conducted using humans.

In contrast with the above two examples’ use of relatively raw data, citizen science systems also may involve humans manipulating abstract representations of data. FoldIt [7] in particular recruits participants to manipulate protein structures as represented in a downloadable application. Large numbers of participants ‘fold’ these protein structures into stable shapes—something that is a hard computational problem—thus leading to the mass collection of appropriate protein configurations.

Finally, citizen science systems may also be mobile-based. Paulos et al. [20], for example, employ participatory data



Figure 2: Classifying galaxies with Galaxy Zoo (l); a slide from Stardust@home (r)

collection techniques in urban environments in order to map out pollution data. Crucially for human computation, such systems support selection of data that is meaningful to urban inhabitants by through drawing upon human agency and local practices tied up in urban life. In contrast, using a fully automated system that drew information from dense arrays of static sensors in the environment might collect similar data, however may well entail great computational complexity in determining which readings were most meaningful to local inhabitants.

### Crowdsourcing

Systems popularly characterised as ‘crowdsourcing’ applications often rely on human computation methods in their configurations of humans and machines. One of the best-known examples is Amazon’s Mechanical Turk website [19] in which tasks are distributed to users who subsequently receive payment based on their completion of the task. Many of these tasks involve hard computational problems, for instance, labelling images, the generation of summarisations of text documents, or transcribing audio and performing recognition on video streams.

Other services provide mobile versions of the Mechanical Turk, such as TxtEagle, which recruits developing world populations for various tasks such as translation and transcription work (that often require highly localised knowledge). These tasks are delivered to participants by mobile phone in return for payment [27].

A further example, perhaps better known to HCI audiences, is reCAPTCHA [32]. Web CAPTCHAs are used to secure websites against spammers, by challenging a potential user with a hard computational problem that is easy for humans to solve (in particular, the recognition of distorted text). The reCAPTCHA system modifies this by sourcing the distorted text from words selected from digitised books via an OCR process. Since OCR processes may fail to perform correct recognition, human activity (i.e., reading, recognising and rendering CAPTCHAs into plain text) may be employed to solve the problem. Thus, reCAPTCHAs use the ‘side effects’ or ‘by-products’ of human activity directed towards another purpose (e.g., accessing a portion of a website), as we saw with EyeSpy and other games with a purpose.

### Interactive computations

Our final systems to review differ in scale and form to citizen science, crowdsourcing and games with a purpose. Interactive evolutionary computation (IEC) and interactive machine learning systems [6] are employed for computational problems in which selecting optimal configurations that are best for a given process is computationally hard. So, in order to solve the problem of deciding upon an optimisation index, IEC systems rely upon human interaction, bringing a user’s activity to bear upon the task, and exploiting human judgement in order to guide the ongoing machine-computational process.

Early work on ‘biomorphs’ [5], in which users select a particular evolutionary path from various machine-suggested paths, led to various other demonstrations of IECs. In an

extensive review of the field, Takagi discusses how human evaluative abilities can be brought to bear on computational problems, such as image detection, in which current algorithms benefit from human guidance in order to finesse results [26]. In another example, human scientific expertise is brought to bear upon geophysics problems, such as determining the location of natural mines through modelling mantle convection within the earth [26]. (This is, interestingly, an inverse of the well-known expert systems found in artificial intelligence.)

Interestingly, and unlike other systems discussed so far, IECs typically do not rely on mass participation, in contrast with ‘citizen science’, ‘games with a purpose’ and ‘crowdsourcing’ applications (although mass participation is not inconceivable). Instead they invert the model: multiple results are machine-generated and then presented to a single user for selection, rather than one machine distributing many tasks to multiple users.

### **What do we mean by human computation?**

Based on our brief review, there are a number of systems that might plausibly be ‘human computation’ systems. As such, we can begin to piece together what human computation systems ‘look like’, albeit in an approximate way.

Firstly we can say that the systems we have reviewed generally rely upon interaction between machine and human (e.g., via some kind of division of labour between the two [16]). Secondly, we can see that another important aspect shared by these systems is that work is done by a human that might otherwise be represented algorithmically for use in a computer system. Indeed, other authors ask similar questions of whether an algorithm could possibly produce similar results to their system in order to present it as being in the ‘human computation’ genre [3]. Bound up in this is the question of the level of output quality of an equivalent machine-computable version of the task, which itself often contrasts drastically with human computation system outputs. So, for example, whilst machine-based translation is possible, results remain poor outside of domain-bound tasks (e.g., weather reports) when compared with human-based translation systems.

We note that the classes of problems delegated to humans tend to be what are seen as computationally ‘hard’ tasks, such as computer vision problems, or natural language parsing, which are currently difficult for computers to perform, but easy for humans [29] (although there may be reasons to delegate computationally ‘easy’ or tractable tasks, which we shall discuss later). Generally we suggest that human computation tasks can often be characterised by knowledge-based procedures, expertise or skilled activity, or activity that is highly contextualised. Such tasks often draw upon commonsense and practical reasoning that everyday members of society engage in routinely.

However, as this approximate ‘definition’ might indicate, the boundaries of human computation remain fuzzy, particularly with regard to what constitutes a reasonable task. Thus there are some caveats in this rough characterisation

of these systems. Some systems we have covered in our review push the boundaries. For example, systems like Matchin begin to involve more aesthetic judgements that are less clearly amenable to algorithmic approaches, i.e., determining human preferences for attractive imagery, although we note that services such as Flickr’s “interestingness” rating for photos [4] indicate algorithmic approaches. In a similar way, IEC systems may also combine more obviously subjective processes, such as guided music composition (Sonomorphs and GenJam [26]) or 3D lighting design, in which computer-generated arrangements of lighting in virtual scenes are selected by human intervention (i.e., to decide the appropriate configuration for the scene).

Citizen science has also been used to describe systems that appear to sit further outside the boundaries of what we might consider human computation. For example, some projects involve large numbers of geographically distributed participants collecting biological data, e.g., Ebird or NestWatch. Such tasks are not obviously computational matters and have no clear algorithmic representation. Similarly, crowdsourcing systems also emphasise the blurry boundaries of human computation. The Mechanical Turk website, for instance, offers many tasks that fall well outside what we have considered so far as a relevant computational problem for human computation systems. These may be obviously non-computational tasks, such as writing reviews of music, or, instead, computational tasks that are relatively simple (i.e., not ‘hard’), such as scraping web content. Finally, some TxtEagle tasks, such as citizen journalism, also do not fit within the boundaries as they would stretch the notion of ‘computation’ in the term ‘human computation’.

### **DESIGN CHALLENGES FOR HUMAN COMPUTATION**

In our short review above, the use of human computation methods within standalone, networked and mobile systems is very varied. With the increasing diversity of applications of human computation, there is a need for developing a design-oriented understanding for the ways in which humans and their interactions together may be employed as computational ‘components’ within machine processes.

Drawing from our study of EyeSpy and the systems reviewed in the previous section, here we develop five challenges facing designers of human computation systems. Some of these challenges also have relevance outside the domain of human computation we have loosely defined and in concluding we shall discuss this issue.

#### **Challenge one: designing for motivation and sustainability**

Human computation tasks will require groups of participants to be recruited and then motivated and sustained in particular ways. As part of configuring player activity in EyeSpy, we had to consider how players could be motivated to engage in playing the game as well as reflecting upon the inherent ‘saturation point’ that was reached after a certain amount of play (i.e., when all the ‘obvious’ land-

marks had been tagged). In this way a considerable issue for designers is the need to *choose appropriate motivation and sustaining strategies* for the task at hand and likely participant base. Here we will examine the motivation and sustaining strategies used by other systems we have reviewed as well as our own system.

#### *Competition and gaming*

In EyeSpy, we configured the motivations of players primarily by offering them a competitive game and scoring structure, which most players reported as a major feature of their own interest in taking part. This approach is common among games with a purpose. At the core of these game-based and game-like systems is the notion of competition (often in the form of scoreboards or ranking schemes), and social interaction between players (via in-game chat, forums, etc.). So, in using the combination of competition and social interaction to form a key role in motivation to play, such human computation systems will as a by-product enable the processing of large amounts of data.

Characteristic of a competition or game-based motivational strategy is the increased separation between the role of the objects manipulated by players within the game, and the role of those same objects outside of the game. Thus, for instance, for ESP Game images, players interact with one another using the game objects (i.e., images) as the central focus of the work of playing the game. However, subsequently these images and descriptive tags are then used in a very different context, namely image search. This again is similar to EyeSpy, with players generating images for other players to confirm, whereas subsequently the same images were repurposed for navigational tasks of little relevance within the game for players.

Although a citizen science project, FoldIt is explicitly marketed to users as a “puzzle” game and includes game-like components such as leaderboards of most prolific ‘folders’. Further to this, FoldIt users may form teams in order to compete in a more collaborative way with other users. However, FoldIt is not framed purely as a game to potential users, and, like Galaxy Zoo and Stardust@home, highlights the scientific importance of a user’s activity (in this case helping find medical advances in treatments for HIV / AIDS, cancer and so on). In this way, FoldIt merges motivational strategies of gaming enjoyment, and at the same time implies that a participant’s altruism has some relevance for their engagement with the system. The ESP Game and others in the genre also highlight that the player is not only “having fun”, but helping “computers get smarter” [9], thus contributing to some scientific or engineering achievement. This leads us to examine the role of altruism—implied or otherwise—within human computation systems.

#### *Altruism and status payoffs*

A common motivation relied upon by the citizen science projects we have examined in previous sections is a sense of altruism on the part of the participants. In systems such as Galaxy Zoo and Stardust@home, a considerable part of

the way in which user motivation is framed appears to be altruistic, in that participants are asked to contribute towards scientific endeavours. Of course, other motivational factors such as individual fun and enjoyment may still play a vital role in user engagement in such systems.

There are also potential situations in which users may gain social status amongst other participants. For Galaxy Zoo, highlighting unusual images offers a ‘payoff’ for participants (e.g., an anomaly, “Hanny’s Voorwerp”, was named after the Galaxy Zoo user who detected it, Hanny van Arkel, bringing them to prominence and generating further scientific investigation [18]). Thus, for Galaxy Zoo, gaining participants relies on user self-motivation initially, but the configuration of the system maintains these users both through potential payoffs and social interaction in the form of community forums in which findings may be shared and discussed (indeed, this is how “Hanny’s Voorwerp” was first brought to attention).

Compensatory payoffs are perhaps made clearer for Stardust@home users. Besides a more prominent user rankings system helping foster clearer competition between participants, any participants discovering a particle of interstellar dust are offered co-authorship on publications related to that particle, in addition to naming the particle themselves. This more explicit competition compares with strategies employed in EyeSpy and other ‘games with a purpose’. Stardust@home, whilst not being a ‘game’, presents an explicit motivational scoring structure with further motivation provided by the accolade of finding a particle track.

Here we can return again to the notion of game structures and their merging with other motivational forces, such as altruism. At the same time we are also led to consider the separation between the payoffs for the designer, and the payoffs for the user. We find a range of choices with regard to how much human computation systems contain game-like components.

One side of this range is typified by citizen science projects that have hints of game qualities, and yet involve fairly explicit integration of the products of the task, i.e., payoffs for the designers, and what it is that users are achieving, i.e., the payoffs for the user. In Stardust@home, for example, we see a clear and explicit focus on the examination of technical data and the potential resulting scientific accolades for a discovering user. This payoff coincides with the payoff for the designers as well (e.g., a publication). At the same time, we also see on the Stardust@home site game-like elements such as a leaderboard, which promotes competition and potential payoff mostly between users (since it is irrelevant for the scientific results how the details of any competition between users works out).

The opposite side of this range are systems in which user payoffs and manipulation of task objects (e.g., enjoyment in the case of playing the ESP Game, and tagging its images) are very separate from the payoffs for the designer, and the subsequent repurposing of those objects (e.g., the ESP Game’s link to image search).

### *Monetary motivations*

For EyeSpy, we also offered a nominal payment in order to compensate players for any inconvenience generated by playing the game (e.g., going out of their usual routines). This was also a motivational feature of the game, however was balanced with players' desire to compete with one another and to win. Such monetary motivations are also present in crowdsourcing systems such as the Mechanical Turk and TxtEagle, except in these cases, providing a core component of participants' interaction with the system (i.e., that, on the completion of a task, participants then receive payment). It is interesting to see that identical human computation tasks found in the ESP Game and on the Mechanical Turk website may differ greatly in the strategy employed for motivation: the ESP Game is presented as a simple, fun game with the payoff being enjoyment (or perhaps a momentary distraction), with no monetary motivation whatsoever, whereas similar results are achieved through explicit monetary means as Mechanical Turk tasks.

### *Sustaining human computation*

Coupled with these issues, and raised in our analysis of EyeSpy, is one aspect of sustaining engagement, in particular the problem of how 'complete' work done by participants during some human computation task may be. As we found with EyeSpy, due to the players saturating a small geographic area with photo and text tags, play within that area (i.e., finding locations that had not been tagged already) became more difficult with time. Participants explicitly mentioned this as a demotivating factor in playing the game. Thus, sustaining play came at the expense of the motivation strategies we employed (fun, enjoyment and competition).

Sustaining motivation is also important in other systems. Projects such as FoldIt, for example, are enriched by participants becoming more skilled in their task, so in this case, fostering community and competition, and thus sustained engagement, becomes important. Contrastingly, systems like the ESP Game may benefit from a more rapidly churning user base in order to assist the reduction of players becoming very familiar with the system and potentially developing ways to 'game the system'.

However, within the literature, there are few accounts of how existing systems sustain participants over long periods of time, and the challenges that are faced by designers over this matter (or how the user base itself may change with time). Players may get bored by the simplicity of a game, or confused by over-complex rules (or perhaps confused by system 'framing', as in EyeSpy). Alternatively they may suffer from fatigue, drop out, or have problems weaving their play into everyday life. Other significant social factors will influence the success of human computation systems, such as how to 'market' them to potential users. Users may move on, and be replaced by entirely different people with markedly different practical engagements with the system. Many of these and other questions remain unexplored, but remain important factors when designing human computation systems.

### *Discussion*

Various obvious motivational strategies are offered in human computation systems: *fun*, *altruism*, *social interaction*, *payment* and *competition*, to name a few (as there are surely others). Strategies such as these also may be employed more generally in trials of interactive systems (e.g., paying compensation for participation is a common technique). We are also reminded that, for many of the human computation systems reviewed in this paper, social interaction is a core motivation for participation even when this appears peripheral to the actual task.

Configuring motivation using these strategies forms a key design consideration for human computation systems, and will have significant knock-on effects for the quality or characteristics of the objects that come to be manipulated or analysed by users. Designers must take into account the various vectors of motivation that may come into play when engaging participants in their design, and be aware of motivations that are emergent and have not explicitly been designed into the system.

In these instances designers may attempt to minimise the possibility of deliberate disruption, motivating or assisting users with engaging in the task at hand. Designers are also constructing potential social interactions that may be folded into use with and around the system, and this is a further key component of motivation for users. For games this is most apparent when rules of competition provide explicit support for social interactions that motivate engagement in the human computation task.

Finally, considering whether we might want a continually changing group of participants or a stable group is an important decision to address in system design. It may be a virtue for the human computation task at hand to have a rapid 'churn' of participants. This 'lifetime' of a user, combined with what the design requires for successful outputs of participant task work, may at one extreme maximise lifetime through maintaining user interest in the task, or at the other extreme ensure a continual flow of new users. Alternatively a significant component in the sustained success of a human computation task may be how to fit the system into existing everyday routines and interactions.

### **Challenge two: balancing system design and user practices through orientation and framing**

The second challenge involves understanding the nature of balance within human computation system design. This includes understanding participant use of that design, in practice. Once again, this challenge has relevance for other HCI systems.

We found that EyeSpy's game rules oriented players toward a strong concern for two aspects of navigation—recognisability and findability—which were in accord with our repurposing those images for navigation. Nevertheless, the game rules and context encouraged rather than enforced such an orientation. Through this we came to appreciate the importance of considering the context in which the game was presented. For example, even the language we used in

order to introduce and frame the game configured certain expectations about how to play (using the name ‘EyeSpy’ confused some players initially due to the similar name of the children’s game ‘<Children’s game>’). Oriented by this, some players initially began creating ‘riddle’ tags that involved cryptic descriptions of locations, requiring ‘detective work’ on the part of other players (i.e., in keeping with the notion of ‘<Children’s game>’). Rapidly, however, players ceased to create such tags, since the game encouraged an orientation to findable and recognisable tags. This ‘framing’ of the ways in which we presented the system to our participants thus challenged the system’s premises.

The way participants in human computation systems practically interpret system rules is influenced by the way in which the system itself is presented to them. By being aware of and even explicitly designing a process of *framing* in the deployment, we can attempt to orient participants to a particular way of conducting themselves in their interactions with and around the system, and with others. One practical way to do this is to take care with the language that is presented to users of the system, and perhaps employ scripted introductions and debriefings for users to carefully configure their expectations in ways designers intend.

Researchers in other domains have also described the relevance of system framing for users and how it influences their interpretation of the system and directs subsequent action [12]. This also has relevance for the design of other human computation systems we have reviewed; how the attitudes, norms and particular approaches of users in their interaction with one another are shaped by the system’s very construction and design. Matchin is of particular interest here because it throws into relief some of the mechanics present in other ‘games with a purpose’. In their study of Matchin’s results, Hacker and von Ahn suggest that cultural norms come to be reflected within the image selections, particularly the way in which considerations of gender came to feature in players’ interactions with one another (such as treating some images as ‘masculine’ or ‘feminine’, so influencing the other player’s preference) [13]. Thus, orientations to other participants as found in systems like EyeSpy and Matchin, forms a key part of the way that game objects are produced and manipulated.

The designer’s intent in constructing system rules, and how that intent is interpreted by participants, is also relevant. For instance, reCAPTCHAs present their purpose to users (e.g., to sign up to a service on a website), but hide their secondary purpose (e.g., to digitise scanned text). Instead of motivating participants with a game mechanic or relying on monetary or altruistic motivation, reCAPTCHAs exhibit an almost complete separation between the intentions of the participant in manipulating the object (i.e., rendering the image of text to plaintext) and the ways in which that activity is then used (i.e., to manually digitise books).

The relationship between system rules and user practices is delicate and hard to predict (as are the specific sequences of

action users engage in [21]). The design of the system will shape, be appropriated, and be subverted by the practical and mundane activities of users.

There are further complications for framing and orienting users human computation systems, however, particularly within systems that produce ‘by-products’. Once again, designer and user orientations may involve potentially conflicting or competing concerns, as well as more complementary or harmonious concerns. In systems with by-products, the framing of the objects manipulated by users within the system, the intentions in that manipulation, and the role and intentions of use of those same objects outside of the system (say, within some scientific practice), must be carefully balanced.

There are of course many such dimensions and factors that play a role in successful configurations of these relationships. The challenge might be, for instance, balancing creativity and fun (as normally associated with gaming and more widely, systems that are enjoyable to use) with the requirement for quality system by-products. In order to meet such challenges, highly adaptive software design processes that involve significant levels of user feedback and offer the possibility of rapid updates in response, may form one way to quickly and iteratively fine-tune the relationship between system rules and human computation products. In this way a suitable balance between motivation strategies (e.g., enjoyment) and the quality of participant’s work may be reached more rapidly than for more traditional iterative design involving successive static software deployments.

Finally, we note that many systems are inherently ‘open’ in their construction; in particular the under-constrained nature of game rules fosters the construction and continual negotiation of shared understandings of ‘appropriate play’ in and around the game. Often within human computation systems design, this openness has often led to a preoccupation with issues such as accuracy, ‘cheating’ and ‘gaming the system’, and thus a concentration on the ways in which such activities may be curtailed in order to promote a particular quality of game objects (e.g., by-products). Whilst this is clearly an important feature, it should be coupled with a more general concern for how framings and orientations may shape user activity, in practice.

### **Challenge three: using situatedness as a resource**

A key feature employed within the design of EyeSpy involved exploiting the situated, local understandings of players. In order to generate our useful data from EyeSpy, the design drew on ‘what anyone knows’ [10] about the local area in order to successfully select (and capture) navigationally relevant images. Players concerned themselves with other players’ potential routes, places locally considered to be central and so on. Social roles were also important to local understandings. This was demonstrated particularly by one of the participants in his orientation to ‘students’ as hypothesised recipients of his images (our trial involved mostly students, however non-students also played). However, although in EyeSpy the role of practical

knowledge as deployed by players in the game featured as judgements regarding what people ('anyone') could find in a local area, we found that even this depended on the cultural positioning of players (e.g. as pedestrians in the city rather than drivers). More generally, then, in EyeSpy, exploiting the local knowledge of participants meant producing more culturally relevant images.

Other systems we have reviewed also purposefully exploit local knowledge. TxtEagle, for instance, relies upon the localised and practical understanding of its participants for a given area, in order to address highly specific linguistic problems such as providing local dialect terms to finesse translations.

This third challenge, like many of the others outlined in this paper, also offers an opportunity. EyeSpy demonstrated to us how human computation does not just involve producing 'objective' results, but can also be about using situated understandings to produce content that draws upon subjective, creative and practical knowledge. The centrality of situatedness in interaction has been a conceptual interest within HCI for some time (e.g., Suchman's ethnomethodological analysis of situated action with technology [24]) and it is key for human computation system design that it both takes into account situatedness and even takes advantage of it.

Thus as designers we must take note of *everyday commonsense and localised understandings*. These understandings may be a resource to exploit or a hindrance to work around. The situated nature of participant interactions with, via and around the system will offer the opportunity of taking advantage of local knowledge and practical understandings of the world that are often very difficult to access otherwise. EyeSpy and TxtEagle both demonstrate ways in which this may be exploited to the advantage of the system, and is particularly pertinent for any mobile human computation systems (of which we may see increasing numbers).

Although there is a clear opportunity here for design, so there is a corresponding potential danger in not accounting for such commonsense, culturally specific or highly localised understandings. This is especially true when deploying human computation systems on the web (e.g., GWAP, Mechanical Turk tasks). For instance, what might be the ways in which conflicts in 'what anyone knows' can come to bear when categorising the content of images found in the ESP Game? The geographic position of web-based participants is of relevance and we can imagine how one symbol may mean very different things to different groups of users (e.g., a swastika or manji, commonly used in Japanese maps to mark temples).

#### **Challenge four: organising human-machine relations**

A fourth challenge to designers is the *organisation of the human and machine components* in their system, particularly how to design their roles. One way, following Kosorukoff [16], is to analyse human computation systems as being based on variations of 'selection' and 'innovation' roles for humans and computers. So, for instance, 'citizen

science' systems may involve computer-based selection, organisation and distribution of the data to be processed, and alongside this, human innovation to conduct piecemeal analysis of the data. However, innovation work, i.e., reporting what can be 'seen' in, say, astronomical imagery in Galaxy Zoo, is, of course, the job for the human participants. IEC and interactive machine learning systems invert selection and innovation roles, with humans providing the selection work in order to guide 'innovation' on the computer's part.

In EyeSpy the machine role was selection of tags, using a strategy that distributed them randomly and without identifying information amongst players (who innovated by generating content). In our interviews, we found that players were often concerned with who else was playing the game. One player, for instance described it as "walking in the footsteps" of others whereas another "saw other people that [she] thought were doing the same thing". In this way, organisation of the 'thin channel' between the selection interface and players helped motivate play, increasing the sense of intrigue reported by players and maintaining interest in the game.

For systems in which 'innovation' is the human role, we have also highlighted how human computation systems may be discussed in terms of *content creation* or *content analysis* (for humans in a 'selection' capacity, such as in an interactive machine learning system, the role will always be analysis). For EyeSpy, players created content which was then analysed by other players for validity, whereas for players of the ESP Game, the primary job is content analysis, and it is players' various analyses of images which are then validated through gathering large numbers of results.

#### **Challenge five: reconsider the utility of machine analogies in human computation**

Our final challenge is conceptual. As mentioned in the introduction, much existing discussion on human computation systems has been based on information processing models of human activity. However, we suggest there are problems when using machine computation or abstract algorithmic processes as a design analogy for collected human activities. If we wish to use humans as 'algorithms' and "networked brainpower" [34] for hard computational problems, *analogies between the two can potentially obscure the considerable design differences*.

In what ways do the challenges for designing for machine algorithmic components and to designing for human 'components' differ? One immediate issue is that algorithms are generally deterministic and have known upper bounds calculation time (computational complexity). They are highly 'accountable' in that one can examine in detail precisely how an output was created. In comparison, within human computation systems, the time needed to obtain information is nondeterministic, and subject to the vagaries of human participation, motivation and conformity with regard to norms of interaction. For example, we were unable to predict how much content would be created during the 3 weeks



of trialling EyeSpy. In contrast to the fixed accountability of an algorithm, accountability in human computation systems is negotiated continuously between users themselves, which, again, was a key feature in the production of navigable images in EyeSpy—players’ activities were made accountable to one another in the game via tagging [1].

Indeed, existing design techniques for human computation systems have been concerned with a very human problem – that of ways to preclude ‘gaming the system’ or feeding the system spurious data (e.g., [28]). This has highlighted the importance of moderation, quality control and ‘orchestration’ activities as vital components in keeping the system running successfully (see [15]), especially with large numbers of participants. However, this is only one part of a broader conceptual challenge in which human roles in computational systems are not seen as interchangeable with machine-based algorithmic components. So, as designers we should question the utility (and applicability) of simply repackaging design strategies that are useful for machine computation, when approaching human computation systems, their design, implementation, and evaluation.

## DISCUSSION

This paper has offered five challenging dimensions along which to design human computation systems, providing a framework for designers that is derived both from our own study and wide ranging analysis of various systems in HCI and beyond. This framework serves multiple purposes: it provides both strategies and opportunities, but also sensitises designers to conceptual issues. The challenges we have covered in this paper vary in type from general interaction design issues, to particular opportunities that exist for human computation systems specifically, to questions that are hard to currently address fully. We can briefly reflect upon this framework for EyeSpy. (1) Designing for motivation and sustainability: we motivated our users with fun, competition and money, however we found the game design did not ensure it was sufficiently sustainable. This may have been solved with a more adaptive game design. (2) Balancing system design and user practices through orientation and framing: we experienced conflicts between our inadvertent framing and orientation of users (via the language we chose) and the way we intended users to engage with the system. Induction ‘rituals’ [2] and careful naming may have been of benefit. (3) Using situatedness as a resource: we created useful by-products (photos of landmarks, etc.) as a product of our system making a virtue of local knowledge as a game resource, although we note that the by-products were also a function of the particular group and mobility they employed—cyclists for instance would ‘see’ things differently (e.g., see [22]). (4) Organising human-machine relations: in EyeSpy, we maintained player interest by restricting the machine interface between players, making tags into ‘clues to be found’. (5) Reconsider the utility of machine analogies in human computation: we relied on the accountability of human activity, balanced via the restricted communication between players (see 4), in order to help drive successful tag creation.

A final topic for discussion is the inherent imprecision of the boundaries and limits of what constitutes a human computation system. There are complexities in any potential definition, mostly due to the diffuse nature of what may or may not reasonably constitute a ‘machine-computable task’. We argue for the value of not providing too narrow or objectifying a definition, and that arriving at such a definition may not only be very difficult but perhaps inappropriate to the phenomenon under study.

In addition, our discussion retains some ambiguity partly because it is vital to understanding the relevance and scope of the challenges outlined in this paper. We have noted how some of the challenges have relevance both within and towards the boundaries of human computation tasks. For example, the importance of motivation strategies applies to citizen science projects such as NestWatch, which we have argued probably sits outside the limits of human computation.

We note also that there are computational tasks which are very ‘easy’ for machines, however, the contexts that address such computational situations may well be ‘hard’ for other, non-computational reasons. The associated costs of developing machine-based algorithms might well be more costly than the more rapid development potential of human-based solutions. Some tasks on the Mechanical Turk website, for example, rely on this fact (e.g., harvesting online data on businesses). Alternatively, logistics may mean that human computation methods work for the collection of data that could easily, but (in a monetary sense) expensively, be gathered. In these cases we might employ the non-monetary methods of human computation systems design, such as enjoyable game mechanics or altruism, in order to motivate and sustain participation.

## CONCLUSION

This paper presented one way of understanding and structuring the emerging design space of human computation, and offers five key issues for consideration when designing such systems. We examined various *strategies* for designing and sustaining motivation, framing and orienting users, and organising human and machine components in terms of innovation and selection relationships. *Opportunities* such as exploiting everyday commonsense and localised understandings may also present themselves in human computation systems. We also explored *conceptual issues* such as the use of machine analogies.

In this paper we have moved forward from demonstrations of particular designs and have begun to develop frameworks that offer more generalised directions for researchers and practitioners. This has not been without limitations, however; for instance, a restriction of our study of EyeSpy was the size and nature of the trial, and it is clear that mass participation is key in deepening our understanding of human computation system design. We are also particularly interested in further validating the framework by employing it in design formatively. Through understanding this space we hope to empower diverse groups of non-experts in de-

veloping their own human computation systems, as well as expanding human computation into areas it does not currently address.

#### ACKNOWLEDGMENTS

This research was funded UK EPSRC (EP/F035586/1, EP/E04848X/1, GR/N15986/01). Thanks also to the other members of the SUMGroup who helped directly or indirectly with this work.

#### REFERENCES

1. Bell, M., et al. EyeSpy: supporting navigation through play. In *Proc. CHI*, pp. 123-132, ACM Press, 2009.
2. Benford, S. et al. The frame of the game: Blurring the boundary between fiction and reality in mobile experiences. In *Proc. CHI*, pp. 427-436, ACM Press, 2006.
3. Bernstein, M. et al. Collabio: a game for annotating people within social networks. In *Proc UIST*, pp. 97-100, ACM Press, 2009.
4. Butterfield, D. S. et al. Interestingness ranking of media objects. *U.S. Patent application 2006/0242139 A1*, February 8, 2006.
5. Dawkins, R. *The Blind Watchmaker*. Longman, 1986.
6. Fails, J. A. and Olsen, D. R. 2003. Interactive machine learning. In *Proc. IUI '03*. ACM, New York, NY, 39-45.
7. FoldIt website, <http://fold.it>, verified 17/09/09.
8. Galaxy Zoo website, <http://www.galaxyzoo.org>, verified 26/02/10.
9. Games With A Purpose website, <http://www.gwap.com>, verified 26/02/10.
10. Garfinkel, H. Some rules of correct decision making that jurors respect. In *Studies in Ethnomethodology*, pp. 104-115, 1967, Prentice Hall.
11. Goldin, D. et al. *Interactive Computation: The New Paradigm*. Springer Verlag: 2006.
12. Grimes, A. et al. EatWell: sharing nutrition-related memories in a low-income community. In *Proc. CSCW '08*. ACM, New York, NY, 87-96, 2008.
13. Hacker, S. and von Ahn, L. Matchin: eliciting user preferences with an online game. In *Proc. CHI '09*. ACM, New York, NY, 1207-1216, 2009.
14. Howe, J. The Rise of Crowdsourcing, *Wired*, June 2006, <http://www.wired.com/wired/archive/14.06/crowds.html>, verified 26/02/10.
15. Koleva, B. et al. Orchestrating a mixed reality performance. In *Proc. ACM CHI*, pp. 38-45, 2001.
16. Kosorukoff, A. Human-based genetic algorithm, <http://www.geocities.com/alex+kosorukoff/hbga/hbga.html>, verified 26/02/10.
17. Law, E., von Ahn, L. Input-agreement: A New Mechanism for Data Collection using Human Computation Games. In *Proc. CHI*, pp. 1197-1206, 2009.
18. Lintott C.J., et al. Galaxy Zoo: 'Hanny's Voorwerp', a quasar light echo?, submitted to MNRAS.
19. Mechanical Turk website, <http://www.mturk.com>, verified 26/02/10.
20. Paulos, E., Honicky, R. and Hooker, B. Citizen science: Enabling participatory urbanism. In M. Foth, editor, *Handbook of Research on Urban Informatics*. IGI Global, 2008.
21. Robinson, M. Design for unanticipated use. In *Proc. ECSCW*. Kluwer, pp. 187-202, 1993.
22. Rowland, D. et al. Ubiquitous computing: designing interactive experiences for cyclists. In *Proc. MobileHCI*. ACM Press, pp. 1-11, 2009.
23. Stardust@home website, <http://stardustathome.ssl.berkeley.edu>, verified 26/02/10.
24. Suchman, L. *Plans and situated actions: The problem of human-machine communication*. Cambridge University Press, 1987
25. Surowiecki, J. *Wisdom of Crowds*, Random House, 2004.
26. Takagi, H. Interactive evolutionary computation: fusion of the capabilities of EC optimization and human evaluation, In *Proc. of the IEEE*, vol.89, no.9, pp.1275-1296, Sep 2001.
27. TxtEagle, <http://www.newscientist.com/article/mg20126956.600-translations-by-text.html>, verified 26/02/10.
28. von Ahn, L. and Dabbish, L. Labeling images with a computer game. In *Proc ACM CHI*, pp. 319-326, 2004.
29. von Ahn, L. and Dabbish, L. Designing games with a purpose. *Communications of the ACM*, 51(8): 58-67, 2008.
30. von Ahn, L., Kedia, M., and Blum, M. Verbosity: a game for collecting common-sense facts. In *Proc. CHI '06*. ACM Press, pp. 75-78, 2006.
31. von Ahn, L., Liu, R., and Blum, M. Peekaboom: a game for locating objects in images. In *Proc. CHI*, pp. 55-64, ACM Press, 2006.
32. von Ahn, L., Maurer, B., McMillen, C., Abraham, D. and Blum, M. reCAPTCHA: Human-Based Character Recognition via Web Security Measures. *Science*, 321 (5895): 1465-1468, 2008.
33. Wegner, P. Why interaction is more powerful than algorithms, *Communications of the ACM*, 40(5), May 1997, pp. 80-91.
34. Zitttrain, J. Ubiquitous human computing, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 366, No. 1881, pp. 3813-3821, 2008.