中央研究院
資訊科學研究所
Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-08-007

An Analytical Study of Puzzle Selection Strategies for the ESP Game

Ling-Jyh Chen, Bo-Chun Wang, Chun-Yang Chen,
Irwin King, Jimmy Lee

July 7, 2008 || Technical Report No. TR-IIS-08-007
http://www.iis.sinica.edu.tw/page/library/LIB/TechReport/tr2008/tr08.html

An Analytical Study of Puzzle Selection Strategies for the ESP Game

Ling-Jyh Chen, Bo-Chun Wang, Chun-Yang Chen
Institute of Information Science
Academia Sinica
{cclljj, bcwang, cychen}@iis.sinica.edu.tw

Irwin King, Jimmy Lee
Department of Computer Science & Engineering
The Chinese University of Hong Kong
{king, jlee}@cse.cuhk.edu.hk

Abstract

"Human Computation" represents a new paradigm of applications that take advantage of people's desire to be entertained and produce useful metadata as a by-product. By creating games with a purpose, human computation has shown promise in solving a variety of problems that computer computation cannot currently resolve completely. Using the ESP game as an example, we propose an evaluation metric, called system gain, for human computation systems, and also study the properties of the ESP game using analysis. We argue that human computation systems should be played with a strategy. An Optimal Puzzle Selection Strategy (OPSA) is then implemented based on our analysis to improve

human computation. Using a comprehensive set of simulations, we demonstrate that the proposed OPSA approach can effectively improve the system gain performance of the ESP game, as long as the number of puzzles in the system is sufficiently large.

1 Introduction

"Human Computation" represents a new paradigm of applications that take advantage of people's desire to be entertained by outsourcing certain steps of the computational process to humans [5, 6, 13]. In [14], Ahn proposed the use of human computation by creating *games with a purpose* that provide entertainment and produce useful metadata as a by-product. By exploiting "human cycles" in computation, human computation has shown promise in solving a variety of problems, such as image annotation and commonsense reasoning, which computer computation has been unable to resolve completely thus far.

Several human computation systems have been proposed recently [8, 9, 15–19]. Among them, the ESP Game [15] was the first to successfully realize the advantages of human computation systems, and it was subsequently adopted as the Google Image Labeler [1]. The rationale behind the ESP game is to motivate people to label images because it is fun. It has been shown that the image labels collected through the ESP game are typically of good quality. Moreover, the game results allow more accurate image retrieval, help users block inappropriate (e.g., pornographic) images, and improve web accessibility (e.g., the labels can help visually impaired people surf web pages [3]).

In this work, using the ESP game as an example, we define a metric, called *system gain*, to evaluate the performance of human computation systems. The proposed metric considers two factors, namely the number of puzzles that have been played in the system and the average outcomes produced by each puzzle. Both factors are critical for human computation systems, but unfortunately they do not complement each other. We believe that human computation systems should be *played with a strategy*. Specifically, based on our analysis, we propose an *Optimal Puzzle Selection Algorithm* (OPSA) that can maximize the system gain performance by properly accommodating the two contrary factors. Using a set of simulations, we investigate the properties of the ESP game, and evaluate the proposed OPSA scheme on two widely used schemes, namely the *Random Puzzle Selection Algorithm* (RPSA) and the *Fresh-first Puzzle Selection Algorithm* (FPSA). The results demonstrate that, with the OPSA scheme, the ESP system can yield a much better system gain performance than the two compared schemes. In addition, the presented analysis is simple and applicable to other human computation systems.

The remainder of this paper is organized as follows. In Section 2, we review related works on human computation systems. In Section 3, we describe the rules of the ESP game and present our analysis. In Section 4, we compare three puzzle selection algorithms for the ESP game, namely the RPSA, FPSA, and OPSA schemes. Section 5 contains a comprehensive set of simulation results, which we analyze and explain in detail. In Section 6, we consider several issues arising from this work. We then summarize our conclusions in Section 7.

2 Background

"Human Computation" was pioneered by Luis von Ahn and his colleagues, who created games with a purpose [14] that people play voluntarily and produce useful metadata as a by-product. By taking advantage of people's desire to be entertained, Human Computation has shown promise in solving some problems that computer computation cannot currently resolve completely. In recent years, a substantial and increasing amount of research effort has been invested in the area, and several human computation systems have been developed for a variety of purposes [8, 9, 15–19].

Among them, the online ESP Game [15] was the first human computation system, and it was subsequently adopted as the Google Image Labeler [1]. In the system, two randomly selected players are paired to create a game, and a randomly selected image is displayed to both players simultaneously. To execute the task, the players must enter possible words to label the image until an "agreement" is reached (i.e., the same word is entered by both players). The agreed word is typically a good label for the image, and the system then displays another image as a new task to be solved. It has been shown that the collected labels facilitate more accurate image retrieval, help users block inappropriate (e.g., pornographic) images, and improve web accessibility (e.g., the labels can help visually impaired people surf web pages [3]).

In addition to image annotation, the Peekaboom system [19] can help determine the location of objects in images; and the Squigl system [2] and the LabelMe system [9] can provide the complete outlines of the objects in an image. Phetch [16, 17] can provide image descriptions that improve web accessibility and image searches, and the Matchin system [2] can help image search engines rank images based on which ones look the best. The concept of the ESP Game has been applied to other problems. For instance, the TagATune system [7], which provides annotation for sounds and music, can improve audio searches. The Verbosity system [18] and the Common Consensus system [8] collect "common-sense" knowledge that is valuable for commonsense reasoning and enhancing the design of interactive user interfaces. The Context-Aware Recognition Survey (CARS) system [20] uses ubiquitous sensors to monitor activities in the home, while the Gopher system [4] employs mobile social gaming for geospatial tagging. Moreover, [11] applies human computation to ontology alignment and web content annotation for the Semantic Web using a set of games, such as OntoPronto, SpotTheLinks, OntoTube, and OntoBay. Finally, Shenoy and Tan [10] showed that it is possible to design environments in which human cannot avoid processing some of the tasks (and produce some useful outcomes), even though they are actively trying not to.

3 The ESP Game

3.1 Game Description

The ESP Game [15] was the first human computation system to take advantage of people's desire to be entertained and provide useful metadata as a by-product. When a user logs into the system, he/she is automatically matched with a random partner. The two players have no idea about each other's identity as they cannot communicate.

Initially, a randomly selected image is displayed to both players simultaneously. The two players then input possible words to label the image until an "agreement" is reached (i.e., the same word is entered by both players), and a bonus score is awarded to both players based on the '*quality*' of the agreed word. In practice, the '*quality*' of a word is measured by its popularity; generally, words that are more popular receive lower scores. After the players agree on a word, they are shown another image, and they have two and a half minutes to label 15 images.

The word on which the two players agree becomes the label of the image, and it can not be used the next time that image is displayed in another game (the word is called a "taboo" word of the image). The rationale for using taboo words is to ensure that each image is labeled with a variety of words.

To be effective, the ESP game tries to collect outcomes with the largest possible aggregated score for each puzzle (image), and needs as many distinct puzzles as possible to be played. There is a trade-off between these two aspects. On the one hand, the system prefers to take as many labels as possible for each puzzle, which will result in the playing of fewer distinct puzzles; on the other hand, the system prefers that each puzzle is played only once, which can lead to the playing of the maximum number of puzzles. Thus, an optimal puzzle selection strategy that can accommodate the two goals is highly desirable. To this end, we propose a metric to evaluate the system gain of the ESP game, and analyze the puzzle selection problem. We discuss the analysis in the next subsection.

3.2 Analysis

Let N be the number of the puzzles that have been played at least once in the system, and let S be the total score of all the agreed labels. We define the system gain, G, of the ESP game as follows:

$$G = ln(N) \times ln(S/N). \tag{1}$$

Clearly, the system gain increases as the number of the games played increases, and/or as the average total score (per puzzle) increases. Suppose that in the system, each puzzle has the potential to yield K labels in total, and each tag is associated with one positive score value based on its popularity. For simplicity, we assume there are totally X distinct scores (i.e., $S_1, S_2, S_3, ..., S_X$) in the system, and $S_i = e^i$. Moreover, we assume that K_i labels have the score S_i, and $K_i = e^{X-i}$. Therefore the total number of potential labels per puzzle (K) can be obtained by Eq. 2, and the expected score value of each tag ($E[S]$) can be obtained by Eq. 3.

$$K = \sum_{i=1}^{X} K_i = \frac{e^X - 1}{e - 1} \tag{2}$$

$$E[S] = \frac{\sum_{i=1}^{X} e^i e^{X-i}}{K} = \frac{e^X(e-1)X}{e^X - 1} \tag{3}$$

Suppose the N puzzles have been played T rounds in total (one puzzle per round), and each puzzle has been played r times on average ($r = T/N$). We rewrite Eq. 1 as follows:

$$\begin{aligned} G &= ln(T/r) \times ln(E[S] \times r) \\ &= (ln(T) - ln(r)) \times (ln(E[S]) + ln(r)) \\ &= -(ln(r))^2 + (ln(T) - ln(E[S]))ln(r) + ln(T)ln(E[S]) \\ &= -\left(ln(r) - \frac{ln(T) - ln(E[S])}{2}\right)^2 + C, \end{aligned} \tag{4}$$

where C is a constant with a value equal to $ln(T)ln(E[S]) + \left(\frac{ln(T)-ln(E[S])}{2}\right)^2$. Note that C also represents the largest possible system gain, which occurs when

$$r = e^{\frac{ln(T)-ln(E[S])}{2}}. \tag{5}$$

4 Puzzle Selection Algorithms

In this section, we compare three puzzle selection algorithms for the ESP game, namely the *Random Puzzle Selection Algorithm* (RPSA), the *Fresh-first Puzzle Selection Algorithm* (FPSA), and the *Optimal Puzzle Selection Algorithm* (OPSA). We take the RPSA scheme's performance as the baseline (in terms of system gain). The heuristics-based FPSA scheme tries to maximize the first component of Eq.1 (i.e., $ln(N)$), while the proposed OPSA scheme tries to achieve the largest possible system gain based on our analysis (as discussed in Sec. 3.2).

Algorithm 1 The Random Puzzle Selection Algorithm (RPSA).

```
**Function RPSA**
$p \Leftarrow Select\_Random(P)$
Return $p$
```

Algorithm 2 The Fresh-first Puzzle Selection Algorithm (FPSA).

```
**Function FPSA**
$p \Leftarrow Select\_Fresh(P)$
Return $p$
```

We use P to denote the set of all puzzles in the system, and define three functions used by the puzzle selection algorithms: 1) $Select\_Random(P)$, which randomly selects a puzzle from the input puzzle set P; 2) $Select\_Played(P)$; which selects the puzzle in the input puzzle set P that has been played most frequently; and 3) $Select\_Fresh(P)$, which selects the puzzle in the input puzzle set P that has been played least frequently. We present the three algorithms in the following.

4.1 RPSA and FPSA

We present the *Random Puzzle Selection Algorithm* (RPSA) and the *Fresh-first Puzzle Selection Algorithm* (FPSA) in Algorithms 1 and 2 respectively. RPSA selects a puzzle at random from the puzzle pool P in each round. As mentioned earlier, it provides the baseline performance of the ESP game in this study. FPSA, on the other hand, selects the puzzle that has been played least frequently in the system. It is a greedy, heuristics-based approach that tries to maximize the first component of Eq. 1.

4.2 The Proposed Scheme: OPSA

In the proposed Optimal Puzzle Selection Algorithm (OPSA) for ESP games, N denotes the number of puzzles that have been played in the system, E denotes the expected score value of each label, and T is the total number of rounds that have been played. In addition, r denotes the optimal number of rounds (discussed in Sec. 3.2); and for each entry p of P, $p.r$ represents the round number in which the puzzle p was played. Suppose the puzzle set P_0 contains all the puzzles that have not been played; P_1 contains all the puzzles that have been played at least once, but less than r rounds; and the set $P_2 = P - P_0 - P_1$

Algorithm 3 The Optimal Puzzle Selection Algorithm (OPSA).

```
Function OPSA
$T \Leftarrow T + 1$
$r' \Leftarrow \lceil e^{\frac{ln(T)-ln(E)}{2}} \rceil$
if $r' > r$ then
    for each $p$ in $P_2$ do
        if $p.r < r'$ then
            Move $p$ from $P_2$ to $P_1$
        end if
    end for
    $P_1 \Leftarrow P_1 \bigcup P_2$
    $r \Leftarrow r'$
end if
if $\{P_1\}$ is NOT empty then
    $p \Leftarrow Select\_Played(P_1)$
    $p.r \Leftarrow p.r + 1$
    if $p.r = r$ then
        Move $p$ from $P_1$ to $P_2$
    end if
    Return $p$
else
    if $\{P_0\}$ is NOT empty then
        $p \Leftarrow Select\_Random(P_0)$
        $p.r \Leftarrow 1$
        if $p.r < r$ then
            Move $p$ from $P_0$ to $P_1$
        else
            Move $p$ from $P_0$ to $P_2$
        end if
        Return $p$
    else
        $p \Leftarrow Select\_Fresh(P_2)$
        $p.r \Leftarrow 1$
        Return $p$
    end if
end if
```

contains the other puzzles. We detail the OPSA algorithm in Algorithm 3.

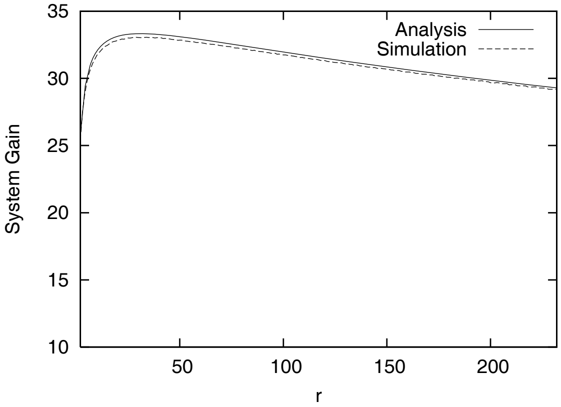

Figure 1. Comparison of the system gain with various r settings in both the simulations and the analysis. ($X = 6$ and $T = 10,000$)

5 Evaluation

In this section, we discuss the simulations performed to investigate the properties of the ESP game based on our analysis. We also evaluate the system gain performance of the three puzzle selection strategies. All the results are based on the average performance of 100 simulations.

5.1 The Optimal r

In the first set of simulations, we evaluated the accuracy of our analytical model for determining the optimal r value for the ESP game. We assumed that the number of puzzles in the system was infinite, and all of them were unsolved at the beginning of the simulation (i.e., no labels were discovered for any puzzles). Moreover, we set the total round number of games played (T) to 10,000. Figure 1 shows the evaluation results in terms of system gain for r values between 2 and 230, when the maximum score value X was fixed at 6. In the figure, the analysis curve is derived by Eq. 4, where $E[S]$ can be obtained by Eq. 3. It is equal to 10.3353 when $X = 6$. We observe that the analysis curve matches the simulation curve very well, and the optimal r values (i.e., those that yielded the largest system gain) of the two curves are also comparable.

Additionally, we varied the maximum score value X in the range 2 to 10 and compared the derived

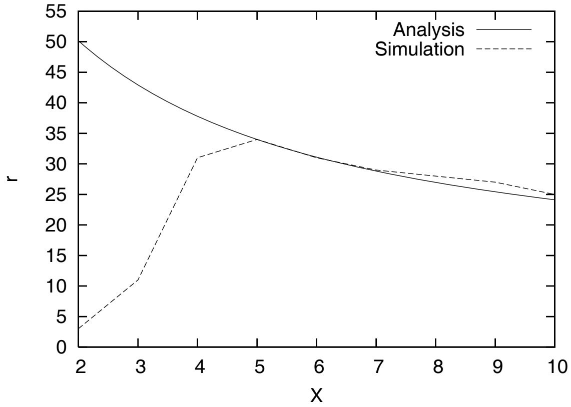

Figure 2. Comparison of the optimal r values derived by simulations and analysis, where $T = 10,000$ and X varies between 2 and 10.

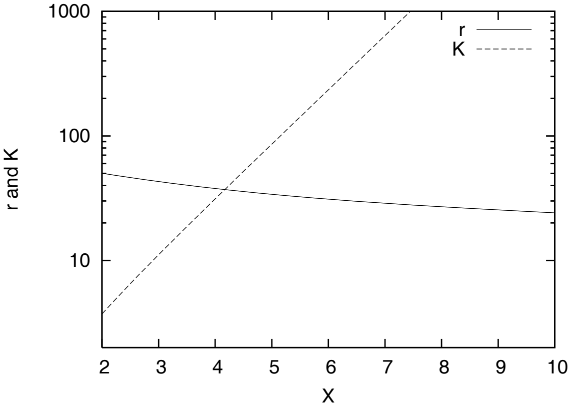

Figure 3. The relationship between the values of r and K with various X values.

optimal r values using both simulations and analysis, as shown in Figure 2. The results show that the analysis curve only matches the simulation results well when $X \geq 5$. The reason is that, in the analytical model, the optimal r value is larger than the total number of potential tags per puzzle (K in Eq. 2) when $X < 5$, as shown in Figure 3. Thus, the analytical model can not be used when $X < 5$ because the optimal number of rounds per puzzle (r) is larger than the number of tags (K) that a puzzle has in the system.

From Figure 2, we also observe that, when $X > 5$, the optimal r value decreases as the X value increases. This confirms our intuition that the value of $E[S]$ increases as X increases (cf. Eq. 3). As a

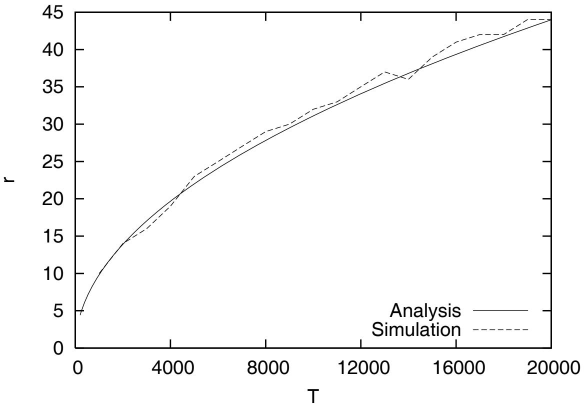

Figure 4. Comparison of the optimal r values derived by simulations and analysis, where $X = 6$ and T varies between 200 and 20,000.

result, based on Eq. 5, the optimal r value will decrease as X increases. We find that if there are many different scores in the system, more rounds of each puzzle should be played in order to achieve a better overall system gain.

5.2 The Relationship between T, N, and r

Next, we evaluate the relationship between the total number of game rounds T, the number of played puzzles N, and the number of game rounds required to maximize the system gain r in the proposed Optimal Puzzle Selection Algorithm. Figures 4 and 5 show the comparison results of r and N with various T values in the range 200 to 20,000 (X is fixed at 6).

Figures 4 and 5 show that our analytical model matches the simulation results very well in all the cases. In addition, we observe that both of the r and N values increase as the value of T increases. There are two reasons for this phenomenon: a) as the total number of game rounds increases, each puzzle tends to take more labels from the system; and b) a larger number of puzzles are played. Since $N = T/r$, the results confirm that the proposed OPSA approach can effectively balance the two goals, i.e., maximize the number of games played, while identifying as many labels per puzzle as possible.

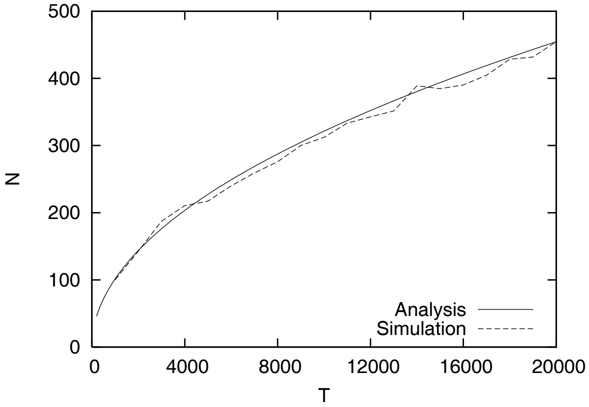

Figure 5. Comparison of the N values derived by the simulations and analysis, where $X = 6$ and T varies between 200 and 20,000.

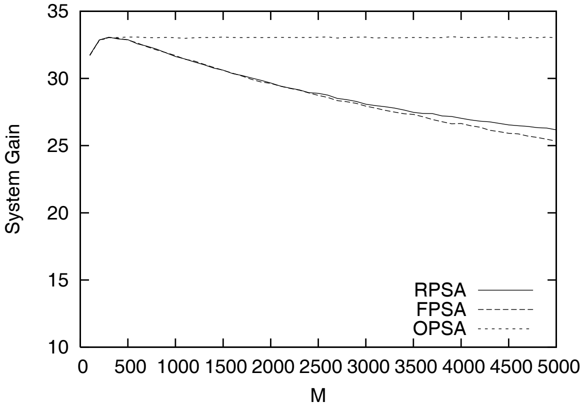

Figure 6. Comparison system gain achieved by the OPSA, FPSA and RPSA schemes with various numbers of puzzles, where T is fixed at 10,000 and X is set to 6.

5.3 Comparison of RPSA, FPSA, and OPSA

Here, we present the evaluation of the three puzzle selection algorithms in the ESP game. In the simulation, we set $T = 10,000$ and $X = 6$; M denotes the total number of puzzles in the system. The simulation results are shown in Figure 6.

The results in Figure 6 show that, when M is small (say, smaller than a threshold M'), the three algorithms are comparable in terms of the system gain performance achieved. However, when M is larger than M', the system gain of OPSA remains consistent regardless of the changes in the values of M. In contrast, the system gain of FPSA and RPSA degrades as the value of M increases, and RPSA slightly outperforms FPSA when M is very large. More precisely, the threshold M' represents the minimal number of puzzles required to achieve the maximum system gain (i.e., $M' = N = T/r$). Since $T = 10000$ and $X = 6$, we know that $E[S] = 10.3353$ and $r = 31$. Therefore, $M' = 10000/31 \approx 321$ in this case. The results indicate that, while using the OPSA scheme, the ESP game must maintain at least a certain number of puzzles to achieve the maximum system gain[1]; otherwise, it will favor the RPSA and FPSA schemes because their performance is comparable to that of OPSA and the ease of implementation.

6 Discussion

We have presented an analytical model that describes the system gain performance of the ESP game. In addition, we propose an Optimal Puzzle Selection Algorithm (OPSA) that can strategically select which puzzles should be played to achieve the largest possible system gain. However, there are some issues that have yet to be addressed. Here, we briefly discuss those issues together with possible solutions.

First, the proposed analytical model only considers the number of the games played and the quantity/quality of the game outcomes when measuring the system gain. However, the speed of generating metadata varies a great deal among different puzzles, and even different rounds of the same puzzle; thus, the proposed model may not be sufficiently representative to measure the "productivity" of the ESP game (i.e., the average quantity and/or quality of the outcomes in each time unit). A possible solution to this issue is to consider an additional factor, i.e., the time consumption of the played games in Eq. 1. We defer a detailed evaluation of this issue to a future work.

Second, in this study, we assume that each round of play produces an outcome with a positive score. However, this may not be always true, since it is possible to have a round without any positive outcomes (e.g., both players agree to pass a puzzle, or they fail to agree about the labels within the time limit).

[1] Fortunately this is not a problem in general, since the number of the puzzles can be increased easily by adding new puzzles from the Internet.

Even worse, the outcomes may be negative (e.g., incorrect labels are assigned either intentionally or accidentally). Thus, we need a validation mechanism to examine the outcomes of the games [12]. Moreover, it is necessary to extend the proposed analytical model to consider scenarios of zero and negative outcomes for more general cases. Again, we defer a detailed discussion of this issue to a future work.

7 Conclusion

In this paper, we have studied the ESP game, an emerging human computation system, and proposed an evaluation metric, called system gain, to evaluate the game's performance. Moreover, we argue that human computation games need to be *played with a strategy* in order to collect human intelligence in a more efficient manner. Based on our analysis, we propose and implement an Optimal Puzzle Selection Algorithm (OPSA) to provide guidelines for the ESP game. Using a comprehensive set of simulations, we have investigated the properties of the ESP game, and demonstrated that the proposed OPSA scheme substantially outperforms other schemes in all test cases. Moreover, the proposed analysis is simple and applicable to other ESP-like games, and the proposed puzzle selection strategy shows promise for the design and implementation of future human computation systems.

8 Acknowledgements

This material was based upon work supported by the National Science Council under grant number NSC 96-3113-H-001-012.

References

[1] Google image labeler. http://images.google.com/imagelabeler/.

[2] Gwap. http://www.gwap.com/gwap/.

[3] J. P. Bigham, R. S. Kaminsky, R. E. Ladner, O. M. Danielsson, and G. L. Hempton. Webinsight:

making web images accessible. In *ACM SIGACCESS Conference on Assistive Technologies*, pages 181–188, 2006.

[4] S. Casey, B. Kirman, and D. Rowland. The gopher game: a social, mobile, locative game with user generated content and peer review. In *International Conference on Advances in Computer Entertainment Technology*, pages 9–16, 2007.

[5] J. Howe. The rise of crowdsourcing. *WIRED Magazine*, 14(6), June 2006.

[6] A. Koblin. The sheep market: Two cents worth. Master's thesis, UCLA, 2006.

[7] E. L. M. Law, L. von Ahn, R. B. Dannenberg, and M. Crawford. Tagatune: A game for music and sound annotation. In *International Conference on Music Information Retrieval*, 2007.

[8] H. Lieberman, D. Smith, and A. Teeters. Common consensus: a web-based game for collecting commonsense goals. In *ACM Workshop on Common Sense for Intelligent Interfaces*, 2007.

[9] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, May 2008.

[10] P. Shenoy and D. S. Tan. Human-aided computing: utilizing implicit human processing to classify images. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 845–854, 2008.

[11] K. Siorpaes and M. Hepp. Games with a purpose for the semantic web. *IEEE Intelligent Systems*, 23(3):50–60, May/June 2008.

[12] Q. Su, D. Pavlov, J.-H. Chow, and W. C. Baker. Internet-scale collection of human-reviewed data. In *International World Wide Web Conference*, pages 231–240, 2007.

[13] J. Surowiecki. *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations*. Doubleday, May 2004.

[14] L. von Ahn. Games with a purpose. *IEEE Computer*, 39(6):92–94, June 2006.

[15] L. von Ahn and L. Dabbish. Labeling images with a computer game. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 319–326, 2004.

[16] L. von Ahn, S. Ginosar, M. Kedia, and M. Blum. Improving image search with phetch. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1209–1212, 2007.

[17] L. von Ahn, S. Ginosar, M. Kedia, R. Liu, and M. Blum. Improving accessibility of the web with a computer game. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 79–82, 2006.

[18] L. von Ahn, M. Kedia, and M. Blum. Verbosity: A game for collecting common-sense facts. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 75–78, 2006.

[19] L. von Ahn, R. Liu, and M. Blum. Peekaboom: A game for locating objects in images. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 55–64, 2006.

[20] D. H. Wilson, A. C. Long, and C. Atkeson. A context-aware recognition survey for data collection using ubiquitous sensors in the home. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1865–1868, 2005.