DBSJ Journal, Vol.7, No.4
March 2009

Oblivious CAPTCHA: A Fifth-Factor Technology for Practical CAPTCHA Use

Hiroaki OZEKI ♥

The CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) idea is widely used as a HIP (Human Interactive Proof) for distinguishing between humans and computer programs. These are automated tests that humans can pass, but that current computer programs can't handle. Breaking a CAPTCHA generally involves solving a difficult Artificial Intelligence problem. There are demands for new technologies that are stronger against automatic attacks by machines, without making it too hard for humans to pass the tests. In this paper, we propose a concept called the Oblivious CAPTCHA, as a fifth-factor technology for a practical CAPTCHA. The Oblivious CAPTCHA uses tasks such as identifying the English alphabetic characters in a set of mixed alphabetic and non-alphabetic characters, or counting the English alphabetic characters in a string. In experiments we found that the Oblivious CAPTCHA was easy for users, because human beings can recognize images of alphabetic characters quickly and accurately, but this is difficult for computers, because OCR tech-niques tend to misrecognize non-alphabetic characters as though they were alphabetic. This shows our approach is practical. We also describe novel algorithms for enhancing the skill gap between humans and computers that can be used with many existing CAPTCHAs.

1. Introduction

1.1. Background

A CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) [1,2] is an automated test that humans can pass, but that current computer programs can't satisfy. Most CAPTCHAs are based on unsolved Artificial Intelligence (AI) problems, so CAPTCHA breaking is regarded as an AI task. A CAPTCHA is also regarded as a HIP (Human Interactive Proof) that proves an active agent is a human rather than a machine.

A good CAPTCHA should satisfy several requirements. First, a human being should be able to solve it quickly and with little effort, but automated techniques, such as software bots should not be able to solve it easily. Automated attacks are expected to produce many incorrect answers.

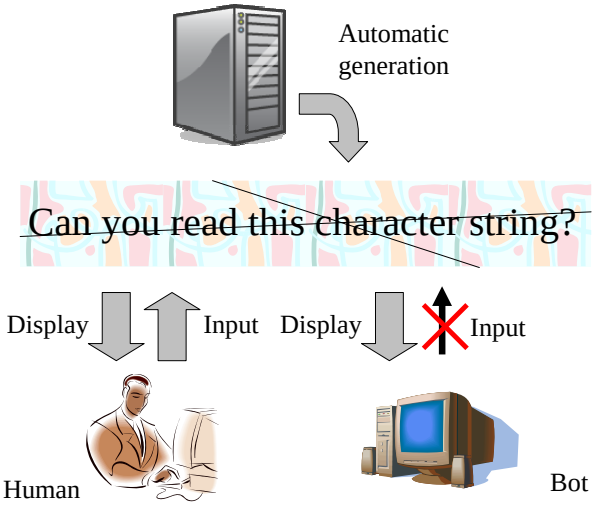

Figure 1: What is a CAPTCHA?

Figure 1 is a simple example of a CAPTCHA used to block software bots (non-humans).

Here are some typical use cases of CAPTCHAs :

- Registration for a new ID or user account (such as the IDs or accounts used for free email, blogs, or storage services).
- Authenticating comments (such as in a blog, wiki, or BBS (Bulletin Board System)).
- Validating online voting (to prevent automated ballot stuffing).

Currently, there are four major CAPTCHA technologies: Text identification (Text) [3-8], Logical puzzles (Puzzle) [9], Photograph identification (Photo) [10-13], and Sound identification (Sound) [14] (Figure 2). Each existing CAPTCHA technology has strengths and limitations. Table 1 is a comparison table of CAPTCHAs.

At present, text identification is the most mature CAPTCHA technology and the kind that is most widely used. However, the attack technologies for text identification have been developing rapidly simply because it is also the biggest target [15-19]. Text identification has improved its resistance to attacks by adding more processing to the character strings made of alphanumeric characters. Typical processing transformations used with the character strings to help disguise them are called Arc, Deform, Undividable, and Special effects (Figure 3).

Text identification faces two dilemmas, one for length and one for character distortion. Shorter strings are easier and more convenient for human beings, but shorter strings are also easier to attack with automated methods. Using more automated processing can add more complexity to the characters and make it harder to attack them with automatic methods, but that also makes the strings harder and less convenient for people to recognize. The goal is create new technologies that retain strength against attacks from machines, but without making it more difficult for humans to prove their humanity.

1.2. Our Contribution

In this paper, we propose an Oblivious CAPTCHA (Figures 4, 5, 6), using fifth-factor technologies for

♥ Member. Tokyo Research Laboratory, IBM Japan, Ltd. ozkmail@gmail.com

practical CAPTCHAs. Our Oblivious CAPTCHA asks users to answer questions like these:

- Question: "Which of the characters in the 12-character string 'C A Л P T Ж Ч C ∀ H И A' are English letters?" (Correct answer: "CAPTCHA".)
- Question: "How many English letters appear in that string?" (Answer: "7".)

Our experiments showed the Oblivious CAPTCHA is easy for users, because humans can quickly and accurately distinguish between characters of their own alphabet and non-alphabetic characters, while this is a difficult task for computers, because all OCR approaches tend to misrecognize non-alphabetic characters as belonging to the target alphabet. Therefore our approach seems simple and practical.

This paper is structured as follows. We describe related work about CAPTCHAs in Section 2 and explain the Oblivious CAPTCHA in Section 3. Section 4 evaluates our approach through actual experiments. Finally, we conclude in Section 5 with a summary of our results and consider issues that still remain to be addressed.

2. Related Works

Here are the four primary techniques now used for CAPTCHAs [1,2][20-26]:

- Text identification (Text) [3-9,15-18]: This method involves recognizing and reading alphanumeric characters within images.
- Logical puzzles (Puzzle) [9]: This method asks questions about logical puzzles.
- Photograph identification (Photo) [10-13]: This method calls for recognizing photographs, such as asking whether a certain picture is a cat or a monkey.
- Sound identification (Sound) [14]: This involves asking about sounds, usually synthesized voices or recordings.

Figure 2 is an example of four variations of CAPTCHA.

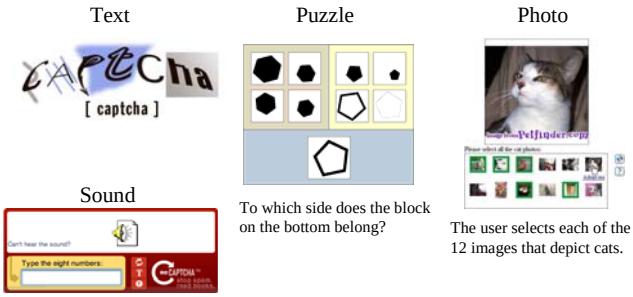

Figure 2: Varieties of CAPTCHAs.

For all of these methods, humans are better than computers at answering the questions. Figure 2 illustrates these four CAPTCHAs.

Each existing CAPTCHA technology has strengths and weaknesses. (△: strengths, ▼: weaknesses)

- Text identification (Text):
  - △ This is currently the most commonly used method. There are several efficient automatic generation algorithms. Human can easily recognize the answers in a few moments. The keyboard response method is quick and convenient. It works effectively even with low quality images and small-screen devices such as mobile phones. The questions are easy to understand and the cognitive loads for the users are small.
  - ▼ Attacks based on improvements in OCR technologies are becoming increasingly effective [15-19]. To respond to the increasingly sophisticated attacks, the images are becoming more complicated, noisy, and distorted, so even human beings find it increasingly hard to see the answers.
- Logical puzzles (Puzzle):
  - △ There are automatic generation algorithms.
  - ▼ There is limited variation in the automatically generated questions or questions may take too long to solve. It is difficult to design questions that are easily understood, because some questions are not intuitive for certain users. Also, some people find certain problems hard to solve.
- Photograph identification (Photo):
  - △ These tasks are especially challenging for artificial intelligence, so the questions are especially resistant to machine attacks. Also, the questions are easily understood, so the cognitive load is small.
  - ▼ Larger images of higher quality are required, so this method is not suitable for many mobile devices (with small screens and low quality images). There are no efficient algorithms to generate the images automatically. It is difficult to prepare photographs with large numbers of elements. Also, the method requires constructing a large and costly database of photographs.
- Sound identification (Sound):
  - △ This is a method that visually handicapped people can also use. There are several efficient automatic generation algorithms. It works effectively even with low quality images and small-screen devices such as mobile phones.
  - ▼ Listening to instructions and questions takes a relatively long time (perhaps 30 seconds). Also, attacks based on improving in voice recognition technology technologies are becoming increasingly effective.

Table 1: Comparison of CAPTCHA.

	Text	Puzzle	Photo	Sound
Present adoption (diffusion and share).	◎	△	△	△
Humans can solve it quickly and easily.	◎	△	○	△
Bots cannot solve it.	○	○	◎	○
Low cost of generating questions.	◎	△	×	◎
Effective for mobile devices.	◎	△	×	◎

Table 1 is a comparison table of the four types of CAPTCHA. (Legend: ◎: Very good. ○: Good. △: Not good. ×: Bad.) At present, text identification is the most developed and most widely used CAPTCHA technology.

Users have become accustomed to text CAPTCHAs, and the competing methods are relatively less advantageous. Puzzles and sounds are slow and can be difficult, and images are expensive and have limited applicability. Therefore it seems clear that text CAPTCHAs will remain popular for the time being.

Preparing a text identification CAPTCHA involves several steps. A string of characters is converted into a small image, and then some operations such as those

shown in Figure 3 are used on that image.

- arc: Thin lines and arcs are added on top of the characters.
- deform: The characters are deformed or warped.
- undividable: Characters are pushed together to make them hard to separate.
- special effects: Decorations or borders are added around the characters.
- combination: Various operations are combined.

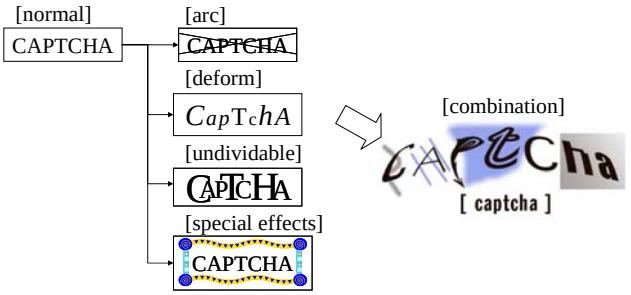

Figure 3: Operations used for text identification images.

3. Oblivious CAPTCHA

3.1. Concept

In this paper, we propose the concept of Oblivious CAPTCHA as an enhancement to the text identification type of CAPTCHA.

Using Oblivious CAPTCHA involves three steps. First, a character string that mixes dummy characters with the characters of the correct answer is prepared. Second, the human is asked to pick the characters of the correct answer out of the mixed character string. Third the results are checked to determine whether a human being properly recognized the correct answer or a machine failed to do so.

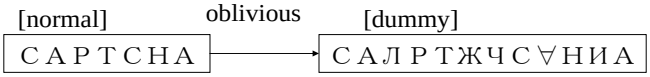

Figure 4: Oblivious CAPTCHA.

Figure 4 shows an example of Oblivious CAPTCHA. The correct answer is written with letters of the English alphabet, and the dummy characters are Russian letters (selected from those that do not appear in English). A human being who reads English selects the correct letters, while a machine using OCR techniques will include some Russian characters (misrecognized as the most similar English letters). The Oblivious CAPTCHA can also be combined with the existing arc, deform, undividable, and effect operations for more strength.

There are two basic assumptions of the Oblivious CAPTCHA approach. One is that literate human beings can quickly and accurately recognize the written form of their own language. The second assumption is that a machine using OCR techniques will tend to force foreign characters into the target alphabet. These two assumptions are described in more detail in the Evaluation and Discussion.

Oblivious CAPTCHA can be combined with existing techniques. Figure 5 shows combinations with Oblivious CAPTCHA.

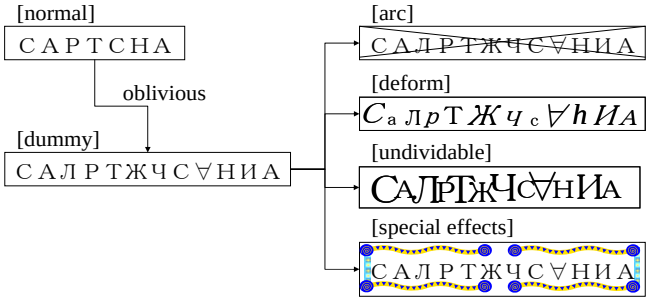

Figure 5: Combinations with Oblivious CAPTCHA.

Here are two examples of Oblivious CAPTCHAs, showing that people can quickly and easily respond to this kind of CAPTCHA. These are high-level questions that humans can answer quickly, even on a mobile device.

- Question: “Select the English letters in the following 12-character string: ‘С А Л Р Т Ж Ч С ∀ Н И А’.” (Answer: “CAPTCHA”.)
- Question: “How many English letters appear in the string?” (Answer: “7”.)

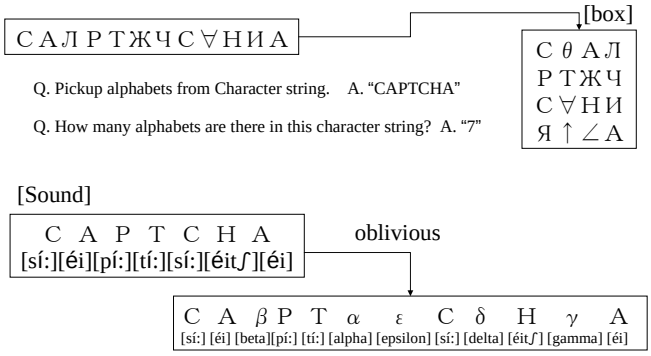

Figure 6: Application of Oblivious CAPTCHA.

Figure 6 shows other examples of applying Oblivious CAPTCHAs. An Oblivious CAPTCHA can be put into a rectangle (see Figure 6 (top)). In addition, if it is a question answered with a number, it is not necessary to specify the order in which the characters are read. Oblivious CAPTCHA can even be used for Sound (see Figure 6 (bottom)). The pronunciations of the names of the characters in the dummy string are read out loud.

3.2. Implementation

Figure 7: Prototype Implementation of Oblivious CAPTCHA. (Difficult Dummy)

Figure 7 is from a simple prototype implementation of Oblivious CAPTCHA. This prototype was implemented

with PHP 5.2.5, Apache 2.2.8, and related libraries.

3.3. Oblivious Transfer and Oblivious CAPTCHA

An oblivious transfer [27] protocol is a protocol by which a sender sends some information to the receiver, but remains oblivious as to what was received. An example of a k-out-of-n Oblivious Transfer would be a secure 2-party protocol where Bob has secrets m_1, m_2, ..., m_n out of n pieces, and Alice has secrets a_1, a_2, ..., a_k ($k < n$) out of of k pieces. After the protocol ends, Alice will have acquired m_{a1}, m_{a2}, ..., m_{ak}. For this example, there are two requirements:

- (1) Alice attaches nothing besides m_{a1}, m_{a2}, ..., m_{ak} and is not understood at all.
- (2) Bob does not reveal anything about a_1, a_2, ..., a_k.

Oblivious Transfer uses a function to conceal the true data. The dummy data is mixed into the true data to implement this function and cannot be distinguished from the valid data. The Oblivious CAPTCHA allows the character of the dummy data to exist together as valid characters and creates a state in which the dummy data cannot be recognized. The concept "It is made to do in the state that cannot be distinguished" is a common feature of Oblivious Transfer and Oblivious CAPTCHA.

4. Evaluation and Discussion

4.1. Question composition and examples

For our experiments we prepared ten kinds of Oblivious CAPTCHAs. The constraints were:

- The characters used for the correct answers are the 26 uppercase letters of English: "ABCDEFGHIJKLMNOPQRSTUVWXYZ".
- The easy dummy characters are these symbols: "◇◎△□☆".
- The difficult dummy characters are 26 letters from other languages and some special symbols: "Λ β θ δ ∈ Γ б Л ↑) ж ∠ Σ Π Θ ρ Φ Я $ Ψ Ц ∀ Ш Ж Ч И".

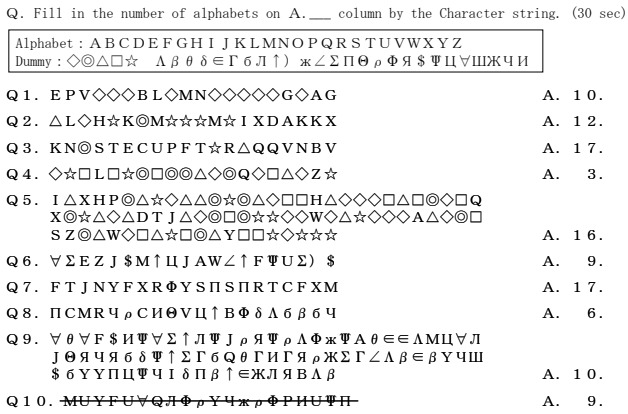

Figure 8: Example Questions.

Figure 8 shows the sample text strings as images. The tests gradually become more difficult.

Table 2 is a summary of the questions.

Table 2: Experimental Questions.

	Alphabet	Dummy	Number of Characters	Alphabet : Dummy
Q 1	A～Z	◇	20	about 5 : 5
Q 2	A～Z	◇◎△□☆	20	about 5 : 5
Q 3	A～Z	◇◎△□☆	20	about 8 : 2
Q 4	A～Z	◇◎△□☆	20	about 2 : 8
Q 5	A～Z	◇◎△□☆	80	about 2 : 8
Q 6	A～Z	Λ β θ～Ч И	20	about 5 : 5
Q 7	A～Z	Λ β θ～Ч И	20	about 8 : 2
Q 8	A～Z	Λ β θ～Ч И	20	about 2 : 8
Q 9	A～Z	Λ β θ～Ч И	80	about 2 : 8
Q 10	~~A～Z~~	~~Λ β θ～Ч И~~	20	about 5 : 5

4.2. Tests of human's reading

4.2.1. Experimental procedures

The experimental environment was:

- Human subjects: 18 people (16 men and 2 women, from 20 – 49 years old, 17 Japanese and 1 Belgian).
- Each subject was shown the ten test patterns of Q1-Q10. Each subject was shown each string for 30 seconds and asked to solve some problem involving the string.
- The number of questions that could be solved in 30 seconds for each of the Q1-Q10 images was recorded.

4.2.2. Experiment Results

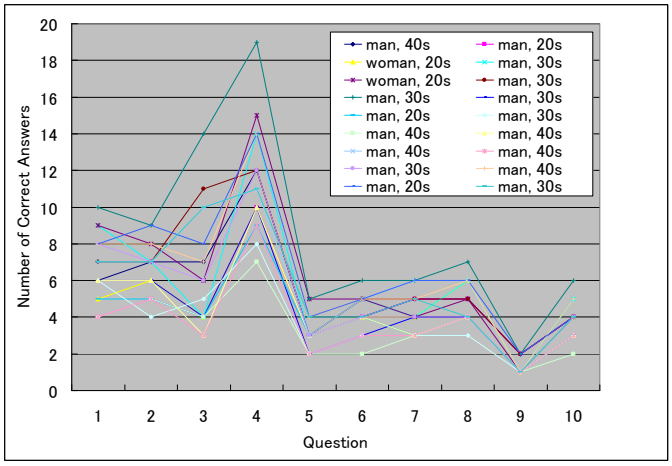

Figure 9: Results of Human Tests.

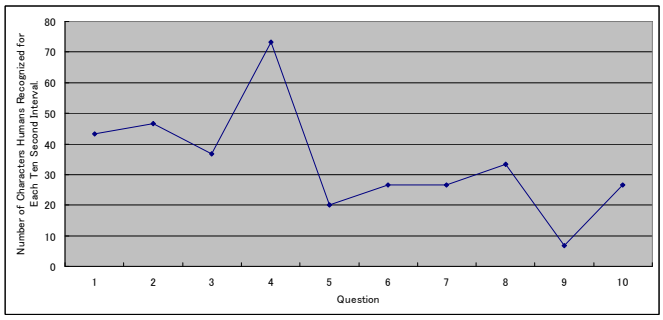

Figure 10: Number of Characters Humans Recognized for Each Ten Second Interval.

Figure 9 is a graph of the experimental results of the human testing. Figure 10 is a graph where the median of the experiment results of the test for each person is calculated, and the number of characters recognized for every period of ten seconds is shown.

Here are the conclusions that can be drawn from the graphs in Figures 9 and 10:

- Mixing in a complex dummy slows recognition speeds compared to mixing in a simple dummy. (From the results for Q1, Q2, and Q6.)
- The mixing ratio of the correct answer and the dummy increases and the recognition speed accelerates in the order of 2:8, 8:2, and 5:5. (From the results of Q2-Q4 and Q6-Q8.)
- When the correct answer exists with a complex dummy, the recognition speed barely changes even if a cancellation line is added. This shows that it is possible to use Oblivious CAPTCHA and Arc together. (From Q6 and Q10.)
- The recognition speed does not increase when there are more characters. (From Q1, Q5, Q6, and Q9.)
- Using the complex dummy characters does not prevent sufficiently accurate answers in sufficiently short response times, whether or not there is a cancellation line. (From Q1, Q2, Q6, and Q10.)

4.3. Test of reading by machine (OCR)

4.3.1. Experimental procedures

The experimental environment was:

- Each question was converted into an image, and then converted into a PDF image format with Acrobat.
- The OCR recognition processing was done by using the OCR function of Adobe Acrobat (Acrobat 8 Professional Japanese Edition), and Xero OCR (Yonde Koko 13.00), liading Japanese OCR Software.
- The correct recognition rate (Correct recognition rate for letters) and incorrect recognition rate (Rate at which dummy characters were falsely recognized as valid letters) were calculated as:
  - Correct recognition rate = (Number correctly recognized letters) / (Total number of letters)
  - Incorrect recognition rate = (Number of dummy letters falsely recognized as valid letters) / (Total number of letters)

4.3.2. Experiment Results

Figure 11 is a sample of the results of the OCR scanning tests. Figure 12 is a graph of the experiment results of the tests with an Acrobat OCR Scanner .Figure 13 is a graph of the experimental results with a Xero OCR scanner.

These results show:

- Mixing in complex dummy characters is effective. The mixing ratio of the correct answers and the dummy characters increases the recognition failures as shown for 2:8, 5:5, and 8:2.
- The higher the ratio of the dummy characters, the more recognition failures.
- For an analytical attack using OCR it would be more effective to focus on increasing the incorrect recognition rate rather than trying to lower the correct recognition rate.

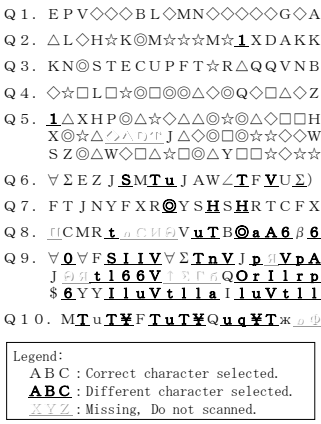

Figure 11: Sample of OCR Scanning Test Result.

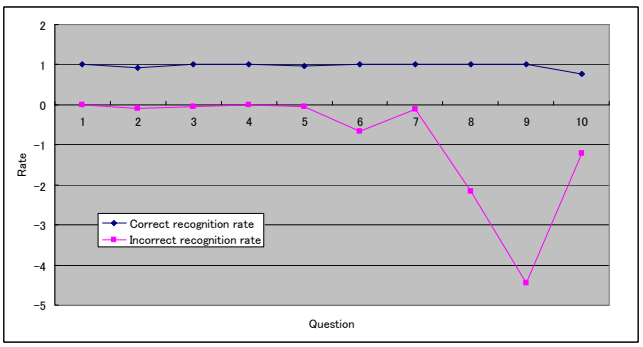

Figure 12: Result of OCR Scan Test (Correct/Wrong rate) (Acrobat OCR).

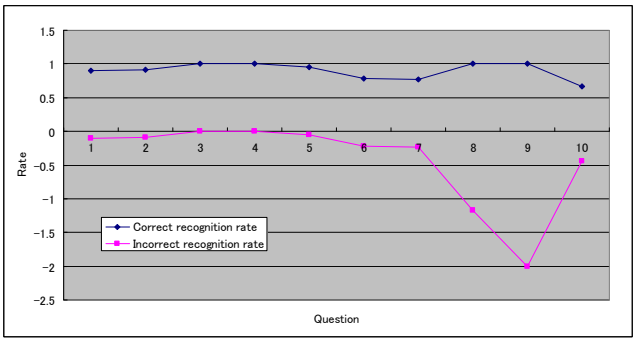

Figure 13: Result of OCR Scan Test (Correct/Wrong rate) (Xero OCR).

4.4. Combined method for Oblivious CAPTCHA

These results can be used together for a better system. The combination would focus on using more characters that humans can read easily, and more characters that machines (OCR) cannot read easily.

- Humans can quickly distinguish between the letters they can read and the dummy (non-alphabetic) characters. They can make these distinctions quickly even when the strings are longer.
- It is more effective to use Complex dummy characters such as "Λ β θ ～Ч И" than simple dummy symbols such as "◇◎△□☆".
- Combining a small number of letters and a lot of dummy characters is more effective.

- Using the Oblivious CAPTCHA technique and Arc together is effective. The reading speed of the humans is barely affected when they are used together.

5. Summary

In this paper, we proposed Oblivious CAPTCHA as a fifth-factor technology. Through experiments, we showed that our approach is easy for users, because humans can distinguish between letters of their alphabet and other characters at high speed and with high accuracy, and this task is difficult for computers, since OCR software misrecognises many characters. We want to prove the utility of this technique through a larger scale experiment in the future.

[References]

[1] Luis von Ahn, Manuel Blum, and John Langford: "Telling humans and computers apart automatically", *Communications of the ACM*, pages 56-60 (2004).

[2] Luis von Ahn, Manuel Blum, Nicholas J. Hopper, and John Langford: "CAPTCHA: Using Hard AI Problems For Security", *Eurocrypt 2003* (2003).

[3] H.S. Baird, M.A. Moll, and S.Y. Wang: "ScatterType: A Legible but Hard-to-Segment CAPTCHA", *Eighth International Conference on Document Analysis and Recognition*, pages 935-939 (Aug. 2005).

[4] M. Chew and H.S. Baird: "BaffleText: a human interactive proof", Proceedings of 10th IS&T/SPIE Document Recognition & Retrieval Conference 2003, San Jose; CA; USA (2003).

[5] A.L. Coates, H.S. Baird and R.J. Fateman: "PessimalPrint: A Reverse Turing Test", *Int'l. J. on Document Analysis & Recognition*, Vol. 5, pages 158-163 (2003).

[6] P. Simard, R. Szeliski, J. Benaloh, J. Couvreur, and I. Calinov: "Using Character Recognition and Segmentation to Tell Computers from Humans", *Int'l Conference on Document Analysis and Recognition* (ICDAR), (2003).

[7] Amalia Rusu and Venu Govindaraju: "Visual CAPTCHA with Handwritten Image Analysis", Lecture Notes in Computer Science, Springer Berlin, *Human Interactive Proofs*, Volume 3517/2005, pages 42-52 (2005).

[8] Henry S. Baird, Michael A. Moll, and Sui-Yu Wang: "A Highly Legible CAPTCHA That Resists Segmentation Attacks", Lecture Notes in Computer Science, Springer Berlin, Volume 3517/2005, *Human Interactive Proofs*, pages 27-41 (2005).

[9] H.S. Baird and J.L. Bentley: "Implicit CAPTCHAs", In *Proceedings of Document Recognition and Retrieval XII (IS&T/SPIE Electronic Imaging)*, volume 5676, pages 191-196, San Jose, CA (Jan. 2005).

[10] Jeremy Elson, John R. Douceur, Jon Howell, and Jared Saul: "Asirra: A CAPTCHA that Exploits Interest-Aligned Manual Image Categorization", Conference on Computer and Communications Security (CCS'07), pages 366-374, *Proceedings of the 14th ACM Conference on Computer and Communications Security* (Oct. 2007).

[11] Monica Chew and J.D. Tygar: "Image recognition CAPTCHAs", In *Proceedings of the 7th International Information Security Conference* (ISC 2004), pages 268-279. Springer (Sep. 2004).

[12] Ralph Gross, Jianbo Shi, and Jeff Cohn: "Quovadis Face Recognition?", Technical Report CMU-RI-TR-01-17, Carnegie Mellon University Robotics Institute (June 2001).

[13] Wen-Yi Zhao, Rama Chellappa, P.J. Phillips, and Azriel Rosenfeld: "Face recognition: A literature survey", *ACM Comput. Surv.*, 35(4):399-458 (2003).

[14] Jonathan Holman, Jonathan Lazar, Jinjuan Heidi Feng, John D'Arcy: "Developing usable CAPTCHAs for blind users", *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility*, ACM SIGACCESS Conference on Assistive Technologies, pages 245-246 (2007).

[15] Jeff Yan and Ahmad Salah El Ahmad: "Breaking Visual CAPTCHAs with Naive Pattern Recognition Algorithms", *Computer Security Applications Conference* (ACSAC 2007), pages 279-291 (Dec. 2007).

[16] Kumar Chellapilla, Kevin Larson, Patrice Simard, and Mary Czerwinski: "Designing human friendly human interaction proofs (HIPs)", In Proceedings of ACM CHI 2005 Conference on Human Factors in Computing Systems, Volume 1 of *Email and Security*, pages 711-720 (2005).

[17] Greg Mori and Jitendra Malik: "Recognizing objects in adversarial clutter: Breaking a visual CAPTCHA", In *Conference on Computer Vision and Pattern Recognition* (CVPR '03), pages 134-144. IEEE Computer Society (2003).

[18] K. Chellapilla, and P. Simard: "Using Machine Learning to Break Visual Human Interaction Proofs (HIPs)", *Advances in Neural Information Processing Systems 17, Neural Information Processing Systems* (NIPS), MIT Press (2004).

[19] Gabriel Moy, Nathan Jones, Curt Harkless, and Randall Potter: "Distortion Estimation Techniques in Solving Visual CAPTCHAs", *IEEE Conference on Computer Vision and Pattern Recognition* (CVPR'04), Vol. 2, pages 23-28 (June 2004).

[20] J. Yan: "Bot, Cyborg and Automated Turing Test", the *Fourteenth International Workshop on Security Protocols*, Cambridge, UK, (Mar 2006).

[21] Kolupaev, A. Ogijenko, J.: "CAPTCHAs: Humans vs. Bots", *Security & Privacy*, IEEE, Volume 6, Issue 1, pages 68-70 (Jan.-Feb. 2008).

[22] Daniel Lopresti: "Leveraging the CAPTCHA Problem", Lecture Notes in Computer Science, Springer Berlin, Volume 3517/2005, Human Interactive Proofs, pages 97-110 (2005).

[23] L. von Ahn, M. Blum, N.J. Hopper, and J. Langford: The CAPTCHA webpage. http://www.captcha.net

[24] Tim Converse: "CAPTCHA generation as a Web service", *Proc. of Second Int'l Workshop on Human Interactive Proofs* (HIP '05), ed. by HS Baird and DP Lopresti, Springer-Verlag. LNCS 3517, Bethlehem, PA, USA, 2005. pages 82-96 (2005).

[25] C. Pope and K. Kaur: "Is It Human or Computer? Defending E-Commerce with CAPTCHA", *IEEE IT Professional*, pages 43-49 (March 2005).

[26] S. Shirali-Shahreza and A. Movaghar: "A New Anti-Spam Protocol Using CAPTCHA", *2007 IEEE International Conference on Networking, Sensing and Control*, pages 234-238 (Apr. 2007).

[27] Moni Naor and Benny Pinksas: "Oblivious Transfer and Polynomial Evaluation", *Proceedings of the Thirty-first Annual ACM Symposium on the Theory of Computing*, ACM, pages 245-254 (1999).

Hiroaki OZEKI

Hiroaki Ozeki was born in Japan. He received B.E., M.E., and Ph.D. degrees from Tokyo University of Science in 2002, 2004, and 2007. He joined IBM Japan Ltd. in 2007. He is currently a researcher at the IBM Tokyo Research Laboratory. He is a member of the Information Processing Society of Japan (IPSJ).