CHAMSIN: CaptcHA by camouflaged StrINg

Masayuki Mukunoki¹, Hisaaki Yamaba¹a), Shotaro Usuzaki¹, Kentaro Aburada¹, Tetsuro Katayama¹, Mirang Park², and Naonobu Okazaki¹

¹ University of Miyazaki
² Kanagawa Institute of Technology
a) yamaba@cs.miyazaki-u.ac.jp

Abstract: We propose a new CAPTCHA scheme that uses random dot patterns (RDPs) to prevent highly-developed bots attacks. Human beings can recognize a moving figure filled by a RDP from a background that is filled by another RDP; however, it is impossible to find such figures when they are stationary. Since image recognition by bots is usually carried out frame by frame, it is hard for bots to recognize such moving figures. The proposed CAPTCHA scheme exploits this characteristic. Several experiments were carried out to confirm that the proposed CAPTCHA scheme is usable enough and has enough resistance against bot attacks.

Keywords: CAPTCHA, bot, random dot pattern

Classification: Multimedia Systems for Communications

References

1 Introduction

In order to prevent malicious programs called bots, CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) is introduced in many websites [1]. CAPTCHA is a kind of challenge and response type tests, which asks a question that is easy to solve for human beings but difficult for computer programs and an answerer who can answer it correctly is accepted as a human user.

However, our society comes to need more sophisticated CAPTCHAs because the progress of computer technologies such as character reading or machine learning make it easy to solve text-based CAPTCHAs and image-based CAPTCHAs, which are used mainly now.

One of the approaches that makes CAPTCHAs sophisticated is introduction of high cognition ability of humans. It is expected that to model after the cognition ability is a difficult problem for malwares to resolve.

In this paper, we propose a new reading text CAPTCHA using random dot patterns, which requires human cognition ability to solve. A experimental system of the proposed CAPTCHA was developed and we evaluated the performance of it and its usability through experiments that were carried out using the developed program.

2 Related Work

2.1 Reading Text CAPTCHA
A reading text CAPTCHA is one of the most popular CAPTCHA scheme. CAPTCHAs of this type, such as Gimpy [2], EZ-Gimpy [2], r-Gimpy [3], display a distorted image of an alphanumeric text. A user supposes the original text and input it into a designated textbox. If the answer is correct, the user will be accepted as a human being.

Reading text CAPTCHAs have two advantages: their implementation is very easy to introduce and they have a high tolerance to brute-force attacks. On the other hand, reading text CAPTCHAs have a serious disadvantage that the progress of OCR techniques gives bots an ability to solve them.

2.2 Random Dot Pattern
A random dot pattern is created by scattering black dots at random onto a white background. Watanabe showed that human beings can track a figure filled by a random dot pattern that moves on a background filled by another random dot pattern; however, they lose sight of the figure in the background pattern when the figure stops [4].

Gestalt psychology explains this phenomenon. According to the principal of common fate, stimulus elements are likely to be perceived as a unit if they move together [5].

2.3 SNOW NOISE CAPTCHA
SNOW NOISE CAPTCHA [6] is a CAPTCHA scheme that uses random dot patterns. This CAPTCHA is a movie-based one and uses “meaning
Fig. 1. The proposed CAPTCHA using random dot patterns (video).

information generated from meaningless information (random dots).” As the meaning information will be changed while malware is detecting meaning information from movie CAPTCHA, malware hardly unlocks CAPTCHA. In this CAPTCHA scheme, an area of an answer dot pattern moves on a background dot pattern. A user finds the cluster of dots and input the coordinate of its position. This CAPTCHA will be unlocked if the answer is correct.

3 Proposed CAPTCHA

In this paper, we propose new CAPTCHA scheme that has tolerance to automated attacks by making good use of the characteristics of random dot patterns. The proposed CAPTCHA is a reading text CATCHA but a target strings is filled by a random dot pattern and the string moves on a background filled by another random dot pattern (Fig. 1). We can read such a string when it moves; however, it is impossible to read such a string when they are stationary. Since image recognition by bots is usually carried out frame by frame, it is hard for bots to recognize such moving figures. The answerer will be accepted as a human begin by showing that he can read the strings.

4 Countermeasure against Object Tracking methods

It is considered that there are some methods that may be effective to resolve the proposed CAPTCHA. One method is using differences between two successive movie frames, another method is Optical Flow, which expresses moves of an object as vectors and extracts the shapes of the characters of the string by following the moving text. Reading the text can be achieved by applying OCR technique on the extracted shapes.
To prevent these kinds of attacks, we adopted two countermeasures. First, we introduced a method that rolls the background (Fig. 1). If the background is stationary, it is easier to specify the shapes of the answer text.

Next, we display two planes, called obstacle planes, filled by semitransparent random dots in front of the answer text to prevent attacks based on optical flow. These planes make it difficult to grasp the shapes of the characters of the answer text.

Fig. 2 shows the results of detecting a moving text using an optical flow method. The upper figure is the detecting results without obstacle planes. Some shape of the answer text is detected and it is expected that this may be a hint to read the text. The lower one is the result of the case using obstacle planes and no shape was found. These results show that adding obstacle planes prevents text detection using optical flow.

5 Experimental Evaluation

Several experiments were carried out to confirm that human users can read a answer text of the proposed CAPTCHA. Also, usability evaluation was carried out to investigate that the proposed CAPTCHA is practical enough.

5.1 Implementation of Experimental System

The experimental environment for the experiments was developed using JavaScript. Random dot patterns were generated by selecting black \( (R, G, B) = (255, 255, 255) \) or white \( (R, G, B) = (0, 0, 0) \) for each pixel at random. Canvas element of HTML 5 is used to realize the proposed CAPTCHA in the experimental environment. We introduced four layers. The lowest layer is used to
display a background plane, the next layer is used to display an answer text and the rest two layers are used to display semitransparency obstacle planes. All random dot patterns are generated for each CAPTCHA trial. An answer text is also generated for each trial; four characters are selected from 26 upper case alphabets (for simplicity) at random in the experiments.

5.2 Conditions
30 students of University of Miyazaki participated as experimental subjects. The experiments were carried out as follows.

Each participant tried to solve the CAPTCHA ten times. The results, success or failure, and the elapsed time from the CAPTCHA was displayed to the user for each trial. After the trials, we asked the participants some questions to evaluate the usability of the proposed CAPTCHA. In order to obtain quantitative evaluation from the participants, the system usability scale (SUS) [7] was adopted. The average SUS score is 68. Score 80.3 or higher is needed to be included in the top 10% of scores, which means its usability is excellent.

The questions used in this study are shown below.

1. I think that I would like to use this CAPTCHA frequently.

2. I found the CAPTCHA unnecessarily complex.

3. I thought the CAPTCHA was easy to use.

4. I think that I would need the support of a technical person to be able to use this CAPTCHA.

5. I found the various contents and navigation in this CAPTCHA were well integrated.

6. I thought there was too much inconsistency in this CAPTCHA.

7. I would imagine that most people would learn to use this CAPTCHA very quickly.

8. I found the CAPTCHA very cumbersome to use.

9. I felt very confident using the CAPTCHA.

10. I needed to learn a lot of things before I could get going with this CAPTCHA.

5.3 Results and discussions
30 participants solved the proposed CAPTCHA ten times each and 300 results were obtained. The results are shown in Table I.

The success rate was very high (95.3%) and this means that the proposed CAPTCHA is easy enough to solve for human users. The average duration of the proposed CAPTCHA was 8.2 seconds and this is shorter than the duration of ordinary text reading CAPTCHA (about 12 seconds).
<table>
<thead>
<tr>
<th>success rate [%]</th>
<th>95.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>average elapsed time [sec]</td>
<td>8.2</td>
</tr>
<tr>
<td>maximum elapsed time [sec]</td>
<td>18.0</td>
</tr>
<tr>
<td>minimum elapsed time [sec]</td>
<td>3.7</td>
</tr>
<tr>
<td>average SUS score</td>
<td>84.7</td>
</tr>
</tbody>
</table>

Next, the results show that the proposed CAPTCHA is adequate for practical use. The average SUS score was 89 and this shows that the usability of the proposed CAPTCHA is very high so as to be included in top 10% scores.

6 Conclusion

In this paper, we proposed new CAPTCHA scheme that has tolerance against automated bot attacks based on the high cognition ability of human beings. We implemented the experimental system of the CAPTCHA using JavaScript and showed that it has enough resistance against bot attacks using representative image recognition methods. We also developed the computational system of the proposed CAPTCHA and carried out the experiments to evaluate the usability of the proposed CAPTCHA scheme. Both the success rate of human subjects and the SUS score that indicates the evaluation result of its usability were high. The results of experiments showed that the proposed CAPTCHA was useful.

In the future work, we plan to examine its tolerance against other attack schemes together with improvement of its usability.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Numbers JP17H01736, JP17K00139, JP17K00186, JP18K11268.