An evaluation of the interactive video CAPTCHA method against automated attack

Kentaro Aburada\textsuperscript{1a)}, Shotaro Usuzaki\textsuperscript{1}, Hisaaki Yamaba\textsuperscript{1}, Tetsuro Katayama\textsuperscript{1}, Masayuki Mukunoki\textsuperscript{1}, Mirang Park\textsuperscript{2}, and Naonobu Okazaki\textsuperscript{1}

\textsuperscript{1} University of Miyazaki, Japan
\textsuperscript{2} Kanagawa Institute of Technology, Japan

\textsuperscript{a)} aburada@cs.miyazaki-u.ac.jp

Abstract: CAPTCHA is designed to detect automated programs (called bots) by requiring them to perform tasks that are easy for humans but difficult for automations. CAPTCHAs are vulnerable to relay attacks in which the challenges are relayed to remote human-solvers. In our previous paper, we proposed an interactive video type CAPTCHA that is strongly resistant to relay attacks. However, a quantitative evaluation of resistance to automated attacks still has not been carried out. Herein, we implement an automated attack for applying to our CAPTCHA and evaluate its resistance to automated attacks. Our results show the robustness of our proposed method against mean shift algorithm.

Keywords: CAPTCHA, bot, relay attack, human-solvers, automated attack

Classification: Multimedia Systems for Communications

References

[1] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, “CAPTCHA: Telling humans and computers apart automatically,” Advances in Cryptology, Eurocrypt’03, vol. 2656 of Lect. Notes Comput. Sci., pp. 294-311, 2003.

[2] Ryohei Tatsuda, Kentaro Aburada, Hisaaki Yamaba, Tetsuro Katayama, Masayuki Mukunoki, Mirang Park and Naonobu Okazaki, “An examination of the interactive video CAPTCHA method to resist relay attack,” IEICE Communications Express, vol. 7, no. 4, pp. 136-141, 2018. DOI:10.1587/comex.2017XBL0192

[3] Online CAPTCHA Solving and Image Recognition Services, https://2captcha.com/, accessed May. 24, 2019.

[4] M. Mohamed, N. Sachdeva, M. Georgescu, S. Gao, N. Saxena, C. Zhang, P. Kumaraguru P. C. van Oorschot and W. -B. Chen, “A three-way investigation of a game-CAPTCHA: automated attacks, relay attacks and usability,” Proceedings of the 9th ACM symposium on Information, computer and communications security. ACM, pp. 195-206, 2014. DOI:10.1145/2590296.2590298

[5] M. Mohamed, S. Gao, N. Saxena, and C. Zhang, “Dynamic cognitive game CAPTCHA usability and detection of streaming-based farming,”
1 Introduction

The “Completely Automated Public Turing test to tell Computers and Humans Apart” (CAPTCHA) is a challenge and response tool that works to prevent malicious programs from attempting to spam or penetrate individual Internet accounts[1]. Currently, the two most popular CAPTCHA types are: (1) text based CAPTCHAs that require the user to read distorted text and retype the correct string in the space provided and (2) image based CAPTCHAs that require the user to select the image that matches the described object. Although these methods are widely used, researchers point out that they can be defeated by bots, so it is desirable to design CAPTCHAs that are more resistant to automatic attacks.

To thwart automatic attacks, interactive video type CAPTCHAs that attempt to identify delays caused by communication relays have been proposed. However, these approaches remain insufficiently resistant to bots. In our previous paper, we proposed an interactive video type CAPTCHA that is strongly resistant to relay attacks[2]. A quantitative evaluation of resistance to automated attacks still has not been carried out. Therefore, we implement an automated attack for applying to our CAPTCHA and evaluate its resistance to automated attacks by comparing the differences in tracking success times between humans and automated object tracking techniques.

2 Relay attack and Related researches

Relay attacks attempt to defeat CAPTCHA tests by relaying the challenges to a remote human coconspirator who solves the problem and provides the answer back to the attacking bots. There are two types of remote human-solvers: (1) workers involved with answering-services such as 2captcha[3] and (2) unwitting victims tricked into providing answers to the CAPTCHA by Trojan programs.

One currently existing technique for defending against relay attacks is the
use of dynamic cognitive game CAPTCHA, called DCG-CAPTCHA\cite{4}\cite{5}. In DCG-CAPTCHA, users follow the instructions given on the challenge website to select particular objects from among multiple decoy images having the same contours, and then carry them to the answer areas via drag and drop operations. This CAPTCHA type can disrupt attacking bots by showing motion pictures on the background. Since this CAPTCHA has interactive and dynamic features that target the network latency inherent in relay attacks (such as relaying the CAPTCHA challenge, task solving by a remote human), it offers a higher level of resistance to those attacks. However, DCG-CAPTCHA has been shown to be insufficiently robust against automated attacks based on image processing\cite{4}.

3 Our proposed CAPTCHA

In our proposed scheme, the user recognizes a moving object (target object) from among a number of randomly appearing decoy objects and tracks it with his/her mouse cursor (Fig. 1). To pass the test, the user must track the target for a given amount of time. Since the target object moves quickly, the time delay makes it difficult for a remote human-solver to defeat the CAPTCHA during a relay attack. In contrast, a legitimate user only solves this CAPTCHA without time delay. In addition, it is also difficult for a bot to identify the target using image processing because it looks very similar to the decoy objects. To evaluate the advantages of our system, we tested its resistance to relay attack\cite{2}. Its ability to resist relay attacks was confirmed by identifying the differences in tracking success time between humans and remote human-solvers based on the assumption that delay time would inevitably result from relaying the test to overseas human operators.

3.1 Resistance to template matching

In order to make it difficult for object tracking techniques (such as template matching) to detect the target using visual features alone, we devised our CAPTCHA so that the target object and the decoy objects have the same shape, color, and size. However, the attacker may still identify the target using the differences of some frame images extracted from our video CAPTCHA. The two features of our proposed CAPTCHA scheme are particularly troublesome:
The decoy objects change positions randomly for each frame.

2. The actual target position changes between consecutive frames are small.

Utilizing these two features, it is possible for an attacker to exclude decoy objects by repeating the AND operation between consecutive binary images, and thus extract the target position. As a countermeasure to this issue, we improved the design of both objects so that the target is not easily captured by the difference between frame images. This countermeasure makes it possible to exclude common areas of a target object between consecutive frame images, thus making it difficult to specify the target position even if a logical operation is used.

4 Resistance to mean shift algorithm

The mean shift algorithm is an efficient approach to tracking objects whose appearance is defined by histograms[6]. Generally, it finds the position that a probability distribution is local maxima from the neighborhood of initial position[7]. In tracking with applying the mean shift, it is possible to find the most similar position to target object from the neighborhood of the detection position in the previous frame[8]. Real-time tracking is performed by applying this algorithm to each frame of the video. We implemented the mean shift using OpenCV[9].

This technique would be useless for our CAPTCHA because the moving target and decoy objects have the same color. However, since the positional changes of the target object between adjacent frames are small, they can be tracked if the target has been identified at least once within the search window and does not fully leave the search window in the next frame image. Fig. 2 shows an example video of applying mean shift to proposed CAPTCHA. To further consider the question of usability, we will adopt a countermeasure to increase the probability that decoy objects will enter the search window by increasing the number of such objects rather than increasing the target movement speed.

4.1 Experiment of mean shift algorithm

We implemented an object tracking program using the mean shift algorithm measured the $T_{\text{success}}$ rate. Human subjects are 20 students of University of
Miyazaki. We executed 100 attacks against each for four CAPTCHA patterns with 20, 30, 40 and 50 decoy objects, respectively.

Fig. 3 shows the result of $T_{\text{success}}$ by our bot program and 20 subjects. Here, we experimentally examined the threshold of the tracking success time in the case of 50 decoy objects while stipulating a FAR (false acceptance rate) no greater than 0.01% as the CAPTCHA security goal [10]. The threshold should be set about 6.2 sec to ensure that the FAR (in Fig. 3 the lower right) is 0.01%. From these results, we found that moving object tracking using the mean shift algorithm was effective against our proposed CAPTCHA scheme. As the number of decoy objects increased, the ability of the attacking bot to track moving objects decreased and the $T_{\text{success}}$ rates declined. However, there were few influences for human. Therefore, it is considered likely that our proposed CAPTCHA scheme can be successfully implemented by appropriately setting the parameters for the decoy objects.

5 Conclusion

In this paper, we evaluated its resistance to automated attacks by implementing mean shift algorithm. Although the object tracking technique using the mean shift algorithm was found to be effective against the original method, we confirmed that employing obfuscation with our proposed CAPTCHA could counter this attack type as well.

In our future work, it will be necessary to decide on appropriate parameters that provide an optimum level of usability and bot resistance.
6 Acknowledgements

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