AUTOMATED IMAGE BASED CAPTCHA SOLVER

CHOONG KAI BIN

UNIVERSITI TEKNOLOGI MALAYSIA

AUTOMATED IMAGE BASED CAPTCHA SOLVER

CHOONG KAI BIN

A project report submitted in partial fulfilment of the
requirements for the award of the degree of
Master of Engineering (Computer and Microelectronic Systems)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JANUARY 2018

I would like to dedicate my thesis to my beloved family

ACKNOWLEDGEMENT

I would like to thank the faculty of Electrical Engineering (FKE) in UTM for giving me an opportunity to enroll in the subject of master project. This has provided me a chance of researching on the latest technology and contributing to the society. Besides, I would like to express my gratefulness to Dr. Usman Ullah Sheikh for all his continous guide. He would always provide support when there is any obstacle. Without his guidance and support, I would not able to complete my master project successfully. Moreover, I also feel thankful to the panel of the examiners. The comments and suggestions provided by them are very useful in helping me to achieve the objectives of the project. Last but not the least, I would like to take the opportunity to thank my mother for his moral support. With those support, I am able to overcome all those challenges.

ABSTRACT

CAPTCHA is known as “Completely Automated Public Turing Test to tell Computers and Humans Apart”. Text-based CAPTCHA is the most common technique used across the internet to detect bot from attacking an online system. An image of distorted word is generated as computer program will have difficulty to read it. In fact, human can read the text in the image CAPTCHA easily. This will help to prevent websites from being attacked by automated scripts. Hence, CAPTCHA should be considered as a win-win strategy that is able to provide security for websites from bot attack but do not cause any disturbance to the user. On the other hand, due to the advancement of pattern recognition technology, current text based CAPTCHA may not be robust enough to defend the intelligence of bot. Thus, in this project, a CAPTCHA solving algorithm is developed to investigate on the strength of CAPTCHA in defeating the bot. Besides, it is also aimed to find out the gap of text based CAPTCHA which in turn helps to develop a more robust CAPTCHA. The project methodology can be broken down into pre-processing, segmentation and character recognition. In pre-processing stage, CAPTCHA image is converted to grey image. After that, lines and dots are removed in order to get back the original word in the image. Segmentation is carried out to crop out individual characters that exist in the image CAPTCHA for character recognition purpose. After the characters have been extracted, the characters are recognized by matching them with the database. If all the characters can be recognized, the text based CAPTCHA is broken. The CAPTCHA solving algorithm was developed with MATLAB, so that it can be trained against a custom dataset. It is able to break ASP.NET text-based CAPTCHA with accuracy of 96 % and 98.86 % in term of word and character recognition respectively.

ABSTRAK

CAPTCHA dikenali sebagai "Completely Automated Public Turing Test to tell Computers and Humans Apart". CAPTCHA merupakan teknik yang paling biasa digunakan di internet untuk mempertahankan sistem dalam talian daripada serangan robot. CAPTCHA direka bentuk dengan perkataan yang hurufnya herot kerana program komputer susah untuk membacanya. Malah, manusia dapat membaca perkataan dalam gambar CAPTCHA dengan mudah. Ini akan membantu laman web dalam membezakan manusia daripada robot untuk mengelakkan serangan cyber. Oleh itu, CAPTCHA adalah cara yang terbaik dalam mempertahankan laman-laman web daripada serangan robot dan tidak menimbulkan gangguan kepada pengguna. Sebaliknya, dengan kemajuan teknologi, CAPTCHA yang berasaskan perkataan mungkin tidak cukup mantap untuk mengatasi kecerdasan bot. Oleh itu, project ini dijalankan adalah untuk mengaji kekuatan CAPTCHA dalam membendung bot. Selain itu, ia juga bertujuan untuk membantu pakar-pakar dalam mereka bentuk CAPTCHA yang lebih mantap dengan mengaji kelemahan CAPTCHA yang sedia ada. Strategi-strategi yang digunakan dalam projek ini boleh dibahagikan kepada pra-pemprosesan, permisahan dan huruf pergenalan. Dalam tahap pra-pemprosesan, gambar CAPTCHA perlu ditukar kepada warna kelabu. Selepas itu, garisan dan titik perlu disingkirkan untuk mendapatkan perkataan asal dalam gambar. Proses pemisahan akan dijalankan untuk memisahkan huruf-huruf dalam perkataan bagi tujuan huruf pengenalan. Selepas dipisahkan, huruf-huruf akan dikenalkan melalui pembandingan dengan huruf dalam pangkalan kata. Jika semua huruf boleh dikenalkan, CAPTCHA yang berasaskan teks akan dipecahkan. Dalam projek ini, MATLAB digunakan unutk menyediakan program yang dapat mengaji CAPTCHA dengan pangkalan kata tersendiri. Dengan ini, ia dapat mencapai ketepatan sebanyak 96 % and 98.86 % dalam mengenalkan perkataan dan huruf dalam ASP.NET CAPTCHA yang berasaskan teks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	-	Artificial Intelligent
CAPTCHA	-	Completely Automated Public Turing Test to tell Computers and Humans Apart
CFS	-	Colour Filing Segmentation
CMU	-	Carnegie Mellon University
CNN	-	Convolution Neural Network
DFS	-	Depth First Search
FKE	-	Faculty of Electrical Engineering
GUI	-	Graphical User Interface
KNN	-	K-Nearest Neighbours
MATLAB	-	Matrix Laboratory
MIT	-	Masachusetts Institute of Technology
MRF	-	Markov Random Field
OCR	-	Object Character Recognition
SVM	-	Support Vector Machine
UTM	-	Universiti Teknologi Malaysia
VSIR	-	Vector Space Image Recognizer

LIST OF SYMBOLS

μ	-	Mean
σ	-	Standard deviation

CHAPTER 1

INTRODUCTION

1.1 Problem Background

In the month of November 1999, an online poll system was released to choose for the best school by the graduated students. However, Carnegie Mellon University (CMU)'s students found out a way to attack the program by voting for CMU for thousands of times. Hence, voting score for CMU was growing rapidly. The next day, Masachusetts Institute of Technology (MIT) also joined the voting "bots" contest. Thus, both universities ended up with each more than twenty-one thousand voting but the others have less than one thousand voting [24]. It raised the issue that the online system can be attacked by automated program if there is no proper internet security system.

In order to cope with the problem, Luis von Ahn, Manual Blum and John Langford introduced term "CAPTCHA" in the early of twentieth century [25]. As CAPTCHA acts as a security system for the online system, it should be designed to be able to tell the human and computer apart. It should be a hard Artificial Intelligent (AI) problem for the bot in order to meet the security purpose. Besides, in terms of the usability property, human can easily solve the challenge in a given duration. Besides, the CAPTCHA generator should be automatic and is not controlled by human in order to avoid any cheating. Moreover, it is needed to be open for all the websites that require any security protection from being attacked by automated script.

Due to its importance, CAPTCHA is used in a wide range of online systems. Beside the online polls and surveys system, it has been implemented in online registration systems such as MyEG in Malaysia as shown in Figure 1.1 [2]. Big companies such as Yahoo and Google have also employed CAPTCHA security system to avoid the attacking of bot by signing up thousands of emails in a few minutes

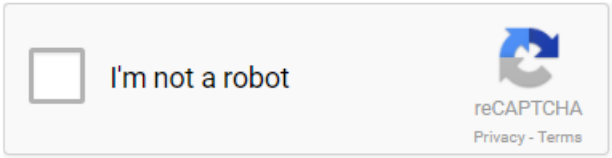

Figure 1.1: reCAPTCHA image in MyEG online registration website [2]

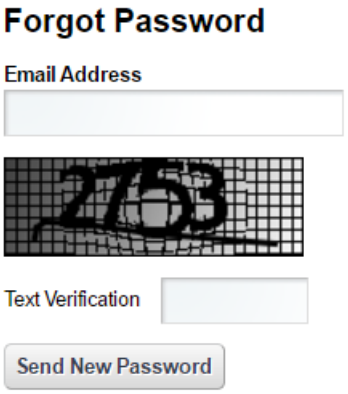

Figure 1.2: Text-based CAPTCHA in PTPTN password resetting system [3]

[26]. Malaysia local company, PTPTN has requested users to solve the text-based CAPTCHA when resetting password as show in Figure 1.2 [3]. Furthermore, it can act as spam blocking to make sure that the email is used on by human only. Last but not the least, CAPTCHA can be used to prevent game cheating and dictionary attack on keyword based system.

On the other hand, due to the advancement of machine learning algorithm, several CAPTCHA such as text-based, image-based, animation based and others have been successfully broken by CAPTCHA solving algorithms. Suphanne Sivakorn et al. managed to break the image reCAPTCHA and Facebook image CAPTCHA with an accuracy of 70.78% and 83.5% respectively [27]. Besides, Vu Duc Nguyen et al. claimed that their algorithm is able to solve all the six characters in the animated CAPTCHA with an accuracy range from 16% to 100% of the time [28].

In this project, research work will be focused on text-based CAPTCHA as currently, it is the most common CAPTCHA used in online systems. M. Korakis et al. [1] developed a CAPTCHA solving algorithm to break the CAPTCHA that produced by ASP.NET Security Image Generator. They claimed that their algorithm was capable of segmenting the characters of the text-based CAPTCHA with an accuracy of 89.39%. Moreover, they have also shown that Vector Space Image Recognizer (VSIR) was able to provide recognition accuracy of 62%, 48% and 18% for the number-based

CAPTCHA, letter-based CAPTCHA and both letter and number-based CAPTCHA respectively [1]. Due to the importance and usage of text-based CAPTCHA, research work will be done in investigating the robustness of the text-based cpatcha.

1.2 Problem Statement

Due to the importance and usage of text-based CAPTCHA, it is crucial to investigate on the robustness of the text-based CAPTCHA. As stated in the background problem section, M.Korakakis, E. Magkos and Ph. Mylonas developed a text-based CAPTCHA solving algorithm to find out the strength of CAPTCHA that is produced by the ASP.NET Security Image Generator [1]. However, they were not able to achieve high segmentation rate and recognition rate. The problems that encountered in their research work "Automated CAPTHCA Solving: An Empirical Comparison of Selected Techniques" are listed as below:

1. In order to remove noise, sum of non-white pixel in a row is computed or changed if it is less than or equal to certain threshold value. This technique requires to have a predetermined value that can be obtained through testing stage.
2. Histogram segmentation technique is used to separate the characters in the CAPTCHA image with a certain threshold value. Any connected characters having vertical sum of non-white pixel more than threshold value cannot be segmented.
3. The percentage of failure in segmenting CAPTCHA that is composed of both letters and numbers is more than 5% as shown in Table 1.1.
4. The accuracy of recognition engines (VSIR and Tesseract) are lower than 50% in recognizing the CAPTCHA that is encoded with both letters and numbers as shown in Table 1.2.

Table 1.1: Percentage of segmentation failure per CAPTCHA variation [1]

Number of images (per CAPTCHA variation)	Percentage of segmentation failure
574 (letters only)	4.87 %
420 (numbers only)	5.74 %
418 (letters and numbers)	7.85 %

Table 1.2: Percentage of recognition rate per CAPTCHA variation [1]

CAPTCHA variation	Accuracy of VSIR	Accuracy of Tesseract
Letters only	48.0 %	16.0 %
Numbers only	62.0 %	56.0 %
Letters and numbers	18.0 %	24.0 %

1.3 Objective

In this project, it is aimed to develop an automated CAPTCHA solver for security image that is able to solve text-based CAPTCHA. Hence, the objectives of the project are listed as following:

1. To investigate on the strength of text-based CAPTCHA in defeating the bot.
2. To improve the success rate in segmenting ASP.NET security image text-based CAPTCHA.
3. To enhance the accuracy in recognizing characters of the text-based CAPTCHA that is produced by ASP.NET security image generator.

1.4 Scope of Study

An automated CAPTCHA solver is to be developed and evaluated with a text-based CAPTCHA dataset. The dataset is produced by ASP.NET security image generator, a CAPTCHA generating free library [29]. It should consist of text-based CAPTCHA with numbers only, letters only and both numbers and letters. Each individual character in the text-based CAPTCHA has to be in equally-sized and standard font while the length of the characters can be arbitrary. Besides, background noise will be added by placing multiple random lines across the security image to prevent the bot from recognizing the CAPTCHA. CAPTCHA can be generated by running the CAPTCHA generator in Microsoft Visual Basic. After that, all the security images will be imported to MATLAB for CAPTCHA recognition purpose. If the recognition is success, the CAPTCHA security system is broken. Thus, these findings will help CAPTCHA designers to revise on security level of the text-based CAPTCHA.

1.5 Organization

In this thesis, there are four main chapters in total. In chapter one, the problem background, problem statement, objective and scope of project was introduced. The objective of carrying out this research is explained. In chapter two, the techniques of designing and breaking the text-based CAPTCHA are discussed. In order to let the readers to understand about the importance of CAPTCHA in protecting the online system from bot attack, the usage and application of CAPTCHA are covered. Besides, the works that are related to the designing and breaking of text-based CAPTCHA are described. The research methodology is covered in chapter three. It includes the information of design flow, system methodology and project scheduling. All of these information is related to the work that has been proposed for the development of automated CAPTCHA solver for security image. In chapter four, detailed descriptions about preliminary results and project outcomes are provided. As summarized in above, those are the four important chapters in the thesis report.

REFERENCES

1. M. Korakakis, E. Magkos, and P. Mylonas, "Automated CAPTCHA solving: An empirical comparison of selected techniques," *Proceedings - 9th International Workshop on Semantic and Social Media Adaptation and Personalization, SMAP 2014*, pp. 44–47, 2014.
2. Myeg, "MyEG Eservices account registration system." `https://www.myeg.com.my/register`, 2017.
3. PTPTN, "PTPTN password resetting system." `http://www.ptptn.gov.my/web/guest/anjung?p{_}p{_}id=58{&}p{_}p{_}lifecycle=0{&}p{_}p{_}state=maximized{&}p{_}p{_}mode=view{&}{_}58{_}struts{_}action={%}2Flogin{%}2Fforgot{_}password`, 2017.
4. Facebook, "An Explanation of Captchas." `https://www.facebook.com/notes/facebook-security/an-explanation-of-captchas/36280205765/`, 2017.
5. S. Pawar and B. M. M, "Captcha : a Security Measure Against Spam Attacks," *IJRET: International Journal of Research in Engineering and Technology*, vol. 02, no. 05, pp. 854–857, 2013.
6. D. Misra and K. Gaj, "Human Friendly CAPTCHAs: Simple Games," *ResearchGate*, 2009.
7. M. Bakry, M. Khamis, and S. Abdennadher, "AreCAPTCHA: Outsourcing Arabic Text Digitization to Native Speakers," *2014 11th IAPR International Workshop on Document Analysis Systems (DAS)*, pp. 304–308, 2014.
8. E. Bursztein, S. Bethard, C. Fabry, J. C. Mitchell, and D. Jurafsky, "How good are humans at solving {CAPTCHA}? A large scale evaluation," *Proc. of the 2010 IEEE Symposium on Security and Privacy*, pp. 399–413, 2010.
9. H. Gao, J. Yan, F. Cao, Z. Zhang, L. Lei, M. Tang, P. Zhang, X. Zhou, X. Wang, and J. Li, "A Simple Generic Attack on Text Captchas," *Proc. Network and Distributed System Security Symposium (NDSS)*, no. February, pp. 21–24,

2016.

10. A. E. Ahmad and W.-y. Ng, “CAPTCHA Design,” *IEEE Internet Computing*, 2012.

11. A. L. Coates, C. S. Division, H. S. Baird, R. J. Fatema, and C. S. Division, “Pessimal Print: A Reverse Turing Test,” *Proceedings of the International Conference on Document Analysis and Recognition (ICDAR ’01)*, pp. 1154–1159, 2001.

12. L. von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, “reCAPTCHA: Human-Based Character Recognition via Web Security Measures,” *Science*, vol. 321, no. 5895, pp. 1465–1468, 2008.

13. I. F. Ince, I. Yengin, Y. B. Salman, H. G. Cho, and T. C. Yang, “Designing CAPTCHA algorithm: Splitting and rotating the images against ocrs,” *Proceedings - 3rd International Conference on Convergence and Hybrid Information Technology, ICCIT 2008*, vol. 2, pp. 596–601, 2008.

14. M. Chew, H. S. Baird, S. Division, and U. C. Berkeley, “BaffleText : a Human Interactive Proof,” *Proceedings of SPIE-IS&T Electronic Imaging, Document Recognition and Retrieval X*, pp. 305–316, 2003.

15. A. Rusu and V. Govindaraju, “Handwritten CAPTCHA: Using the difference in the abilities of humans and machines in reading handwritten words,” *Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR*, pp. 226–231, 2004.

16. E. Bursztein, M. Martin, and J. C. Mitchell, “Text-based CAPTCHA strengths and weaknesses,” *Proceedings of the 18th ACM conference on Computer and communications security*, vol. 2011, pp. 125–138, 2011.

17. J. Yan and A. S. El Ahmad, “A low-cost attack on a Microsoft captcha,” *Proceedings of the 15th ACM conference on Computer and communications security - CCS ’08*, p. 543, 2008.

18. J. Yan and a. E. Ahmad, “Breaking Visual CAPTCHAs with Naive Pattern Recognition Algorithms,” *Twenty-Third Annual Computer Security Applications Conference (ACSAC 2007)*, pp. 279–291, 2007.

19. G. Mori and M. Jitendra, “Recognizing Objects in Adversarial Clutter : Breaking a Visual CAPTCHA,” *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2003.

20. G. Moy, N. Jones, C. Harkless, and R. Potter, “Distortion Estimation Techniques in Solving Visual CAPTCHAs,” *IEEE Conference on Computer*

Vision and Pattern Recognition (CVPR '04), vol. 2, pp. 23–28, 2004.

21. C. Cruz-perez, O. Starostenko, and F. Uceda-ponga, "Breaking reCAPTCHAs with Unpredictable Collapse: Heuristic Character Segmentation and Recognition," *Pattern Recognition*, pp. 155–165, 2012.

22. H. Gao, W. Wang, J. Qi, X. Wang, X. Liu, and J. Yan, "The Robustness of Hollow CAPTCHAs," *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 1075–1085, 2013.

23. E. Bursztein, J. Aigrain, A. Moscicki, and J. C. Mitchell, "The End is Nigh: Generic Solving of Text-based CAPTCHAs," *Usenix Woot*, p. 3, 2014.

24. A. K. B. Karunathilake, B. M. D. Balasuriya, and R. G. Ragel, "User friendly line CAPTCHAs," *ICIIS 2009 - 4th International Conference on Industrial and Information Systems 2009, Conference Proceedings*, pp. 210–215, 2009.

25. L. von Ahn, M. Blum, and J. Langford, "Telling humans and computers apart automatically," *Communications of the ACM*, vol. 47, no. 2, pp. 56–60, 2004.

26. Y. Rui and Z. Liu, "ARTiFACIAL: automated reverse turing test using FACIAL features," *Multimedia Systems*, pp. 8–11, 2004.

27. S. Sivakorn, I. Polakis, and A. D. Keromytis, "I am Robot: (Deep) learning to break semantic image CAPTCHAs," *Proceedings - 2016 IEEE European Symposium on Security and Privacy, EURO S and P 2016*, pp. 388–403, 2016.

28. Vu Duc Nguyen; Yang-Wai Chow; Willy Susilo, "Applied Cryptography and Network Security," in *Proceedings of the 10th international conference on Applied Cryptography and Network Security*, no. June, 2012.

29. Emir Tuzul, "Classic ASP (VBScript) and ASP.NET (VB.NET) Security Image Generator (CAPTCHA) Script." `http://www.tipstricks.org/`, 2017.

30. L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: Using Hard AI Problems for Security," *Proc. 22nd international conference on Theory and applications of cryptographic techniques (EUROCRYPT '03)*, pp. 294–311, 2003.

31. H. Gonzalez, M. A. Gosselin-lavigne, N. Stakhanova, and A. A. Ghorbani, "The Impact of Application Layer Denial of Service Attacks," *Case Studies in Secure Computing: Achievements and Trends*, p. 261, 2014.

32. M. Mehra, M. Agarwal, R. Pawar, and D. Shah, "Mitigating Denial of Service attack using CAPTCHA Mechanism," *International Conference and Workshop on Emerging Trends in Technology (ICWET 2011)*, no. Icwet,

pp. 284–287, 2011.

33. P. Golle and N. Ducheneaut, “Keeping bots out of online games,” *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology - ACE ’05*, vol. V, pp. 262–265, 2005.

34. Y.-W. C. W. S. H.-Y. Zhou, “CAPTCHA Challenges for Massively Multiplayer Online Games,” *2010 10th International Conference on Cyberworlds, CW 2010 (pp. 254-261)*, vol. 2010, pp. 254–261, 2010.

35. M. de Villa, “Two for the price of one,” *INSIGHT./ INDUSTRY HACKERS*, no. September, pp. 40–41, 2016.

36. V. D. Nguyen, “Contributions to Text-based CAPTCHA Security,” *University of Wollongong Thesis Collection*, p. 190, 2014.

37. A. A. Chandavale, A. M. Sapkal, and R. M. Jalnekar, “Algorithm to break visual CAPTCHA,” *2009 2nd International Conference on Emerging Trends in Engineering and Technology, ICETET 2009*, pp. 258–262, 2009.

38. D. Dileep, “A feature extraction technique based on character geometry for character recognition,” *CoRR*, vol. abs/1202.3884, no. 3884, pp. 1–4, 2012.

39. P. W. Frey, “Letter Recognition Using Holland-Style Adaptive Classifiers,” *Machine Learning*, vol. 6, pp. 161–182, 1991.