



Captcha Recognition and Robustness Measurement using Hybrid Approaches

Hina Parveen

Department of computer science
Kanpur Institute of Technology,
Kanpur, India.

Sudhir Singh

Department of computer science
Kanpur Institute of Technology,
Kanpur, India.

Abstract— Completely Automatic Public Turing test to tell Computers and Humans Apart (CAPTCHA) is a HIP (Human interactive Proof) system. CAPTCHAs are used to improve the security of Internet based applications in order to ensure that a web based application which is intended to be used by a human being is not maliciously used by Artificially Intelligent programs called bots. As the current CAPTCHA methods are striving to turn out to be difficult for bots, they are gradually becoming difficult and annoying for human users as well. This paper carries out a systematic study of various Text-based Captchas and proposes the application of Forepart based prediction and Row-wise mapping to break these captchas to evaluate their robustness. Captcha segmentation and recognition is based on Forepart prediction, necessity sufficiency matching and masking.

Keywords—CAPTCHA, Forepart Technique, Vertical Line, Row-wise map99ping, Recursive Function, Robustness.

I. INTRODUCTION

CAPTCHAs are short for Completely Automated Public Turing test to tell Computers and Humans Apart. The term "CAPTCHA" was coined in 2000 by Luis Von Ahn, Manuel Blum, Nicholas J. Hopper (all of Carnegie Mellon University, and John Langford (then of IBM). They are challenge-response tests to ensure that the users are indeed human. The purpose of a CAPTCHA is to block form submissions from spam bots – automated scripts that harvest email addresses from publicly available web forms. A common kind of CAPTCHA used on most websites requires the users to enter the string of characters that appear in a distorted form on the screen.

There are some properties defined in development of CAPTCHA [15].

- Automated: Computer programs should be able to generate and grade the tests.
- Open: The underlying database(s) and algorithm(s) used to generate and grade the tests should be public. This is in accordance with the Kerckhoffs's Principle [25].
- Usable: Humans should easily solve these tests in a reasonable amount of time. The effect of any user's language, physical location, education, and/or perceptual abilities should be minimal.
- Secure: The program generated tests should be difficult for machines to solve by using any algorithm.

There are two major issues should be considered while designing a successful CAPTCHA system: (1) robustness (difficult to break) and (2) usability (human friendly). In this paper, an idea of human recognition based CAPTCHA is presented considering above mentioned requirements. This is based on the concept that, humans can perceive the meaning of captcha recognition and robustness measurement. But there is no such algorithm that can be used to answer those in absolute accuracy. On the subsequent section of related works proposed schemes of CAPTCHA is discussed.

II. RELATED WORK

The first mention of ideas related to "Automated Turing Tests" seems to appear in an unpublished manuscript by Moni Naor [10]. This excellent manuscript contains some of the crucial notions and intuitions, but gives no proposal for an Automated Turing Test, nor a formal definition. The first practical example of an Automated Turing Test was the system developed by Altavista [8] to prevent bots from automatically registering web pages. Their system was based on the difficulty of reading slightly distorted characters and worked well in practice, but was only meant to defeat off-the-shelf Optical Character Recognition (OCR) technology. (Coates et al [5], inspired by our work, and Xu et al [14] developed similar systems and provided more concrete analyses.) In 2000 [1], we introduced the notion of a captcha as well as several practical proposals for Automated Turing Tests. Nowadays, CAPTCHAs are designed in the toughest possible way that prevents any algorithm to break them but still significant work has been done to break these. In 2003, Mori and Malik [4] proposed a shape matching algorithm to break EZ-Gimpy and Gimpy CAPTCHAs. They achieved a success rate of 92% in case of EZ-Gimpy and 33% in case of Gimpy. CAPTCHAs. Chellapillas' et al [5] attacked a number of early CAPTCHAs using machine learning algorithms, and they achieved 4.89% success on an early version of Google's CAPTCHA (around year 2004). In 2007 Ahmad Salah-El-Ahmad, Jeff Yan and Mohamad Tayara described different types of captcha & the need of captcha in the real time environment [3]. In 2011, Ahmad S, Jeff Yan and Tayara proposed a novel attack that is applicable to a whole family of text CAPTCHAs that build on top of the popular segmentation-resistant mechanism of "crowding character together" for security [1]. Jiqiang Song, Zuo Li, Michael R. Lyu Shijie Cai discussed about Recognition of Merged Characters in [2]. This method utilizes the information obtained from the forepart of merged characters to predict candidates for the leftmost character, and then applies character-adaptive masking and character recognition to verifying the prediction.

III. SYSTEM DESIGN

Steps Involved In the Design

Cropping: Cropping is the process by which image is cropped to remove the unwanted noise.

Binarization: Image binarization is usually performed in the preprocessing stage of different image processing related applications such as optical character recognition (OCR) and image retrieval. It converts a gray-scale image into a binary image and accordingly facilitates the ensuing tasks such as skew estimation and layout analysis. As more and more text documents are scanned, fast and accurate document image binarization is becoming increasingly important. Though image binarization has been studied for many years, the thresholding of degraded document images is still an unsolved problem. This can be explained by the fact that the modeling of the document foreground/background is very difficult due to various types of document degradation such as uneven illumination, image contrast variation, bleeding-through, and smear. We try to develop robust and efficient document image binarization techniques which are able to produce good results for badly degraded document images.

Method	F-Measure(%)	PSNR	NMR(*10 ⁻²)	MPM(*10 ⁻³)
Background Estimation	88.53	19.42	5.11	0.32
Local Maximum and Minimum	89.93	19.94	6.69	0.3

Recursive Function: A recursive function definition has one or more base cases, meaning input(s) for which the function produces a result trivially (without recurring), and one or more recursive cases, meaning input(s) for which the program recurs (calls itself). For example, the factorial function can be defined recursively by the equations $0! = 1$ and, for all $n > 0$, $n! = n(n - 1)!$. Neither equation by itself constitutes a complete definition; the first is the base case, and the second is the recursive case. Because the base case breaks the chain of recursion, it is sometimes also called the "terminating case". The job of the recursive cases can be seen as breaking down complex inputs into simpler ones. In a properly designed recursive function, with each recursive call, the input problem must be simplified in such a way that eventually the base case must be reached. (Functions that are not intended to terminate under normal circumstances)

Hybrid Algorithm: Recursive algorithms are often inefficient for small data, due to the overhead of repeated function calls and returns. For this reason efficient implementations of recursive algorithms often start with the recursive algorithm, but then switch to a different algorithm when the input becomes small. An important example is merge sort, which is often implemented by switching to the non-recursive insertion sort when the data is sufficiently small, as in the tilted merge sort. Hybrid recursive algorithms can often be further refined, as in Timsort, derived from a hybrid merge sort/insertion sort.

Forepart Technique: After horizontally scanning a document image, one can obtain the baseline position and the height of each text line, which is the vertical distance between the baseline and the top of the text line. Denoting the height of text line by H_l , the term —forepart H_l means the leftmost $H_l/4$ wide part of the input image of merged characters. For a prototype bitmap, it is nearly the left half

of the bitmap. The forepart prediction is based on three reliable forepart features, i.e., baseline-related feature, forepart height feature, and forepart boundary feature. The baseline-related feature indicates whether the forepart of a character has components under the baseline, which takes value —true| or —false.| The forepart height feature indicates whether the forepart occupies the full height above the baseline, which also takes value —true| or —false.| The forepart boundary feature is a little more complex, defined as follows: $B = \{B(i)|i \text{ is the row index of a bitmap and } \text{baseline} \leq i \leq \text{baseline} + H1\}$ B.

IV. TYPES OF CAPTCHA

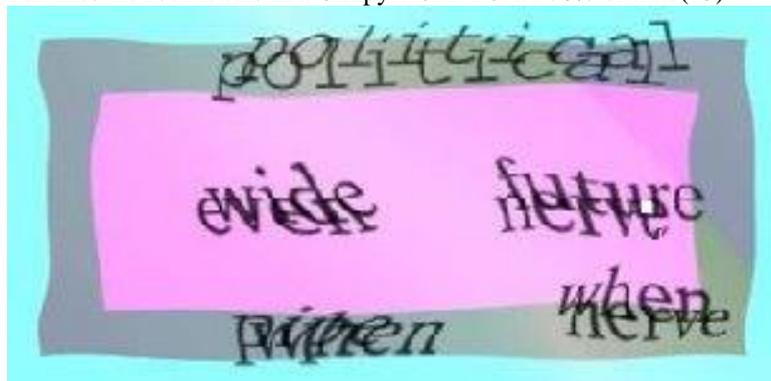
A. Text-based CAPTCHA

Among visual CAPTCHAs, Text-based CAPTCHA is one of the most popular types. It exploits the ability of people to read images of text more reliably than Optical Character Recognition (OCR) or other machine vision system. As these CAPTCHAs are becoming more difficult for genuine users, attackers are also getting better at breaking existing CAPCHAS[6].



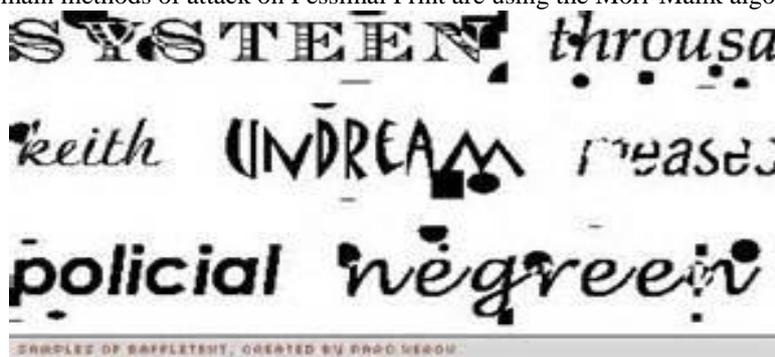
B. Gimpy CAPTCHA

As this method uses its word from a dictionary with 850 words, it can easily be broken in. A correlation algorithm was developed that correctly identified the word in EZ-Gimpy CAPTCHA 99% of the time and a direct distortion estimation algorithm that correctly identifies the four letters in a Gimpy-r CAPTCHA 78% of time(13).



C. Pessimial Print Method

This method tries to prevent the operations of destructive computer software by artificially lowering the quality of the printed letters [5]. Two main methods of attack on Pessimial Print are using the Mori-Malik algorithms and brute-force.



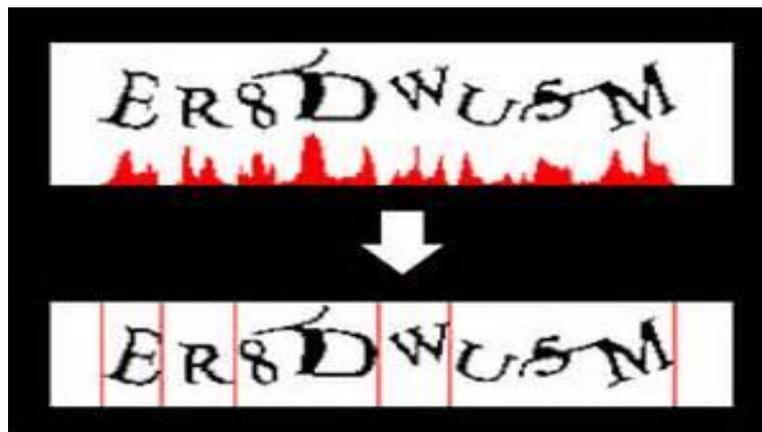
D. Baffletext Method

In the Baffletext method, words that are not provided in English dictionaries are produced, and then the picture of the word is changed with different degrees of ease or difficulty [14]. These text-based CAPTCHAs are prone to bot attacks.



E. MSN CAPTCHA

Microsoft uses a different CAPTCHA for services provided under MSN umbrella. These are popularly called MSN Passport CAPTCHAs. They use eight characters (upper case) and digits. Foreground is dark blue, and background is grey. Warping is used to distort the characters, to produce a ripple effect, which makes computer recognition very difficult.



V. PROPOSED WORK

OCR-based CAPTCHAs are mainly text-based CAPTCHAs in which the user is shown distorted images of letters and/or digits and the user is required to recognize them and type the answer. But, these CAPTCHAs have an inbuilt drawback. The strength of OCR-based CAPTCHAs extensively depends upon the degree of distortion in the displayed text and if increasing security is achieved by increasing text distortion, it may lead to failure of recognition by humans, thus making the CAPTCHA ineffective. In addition, OCR-based CAPTCHAs are problematic for mobile phones and devices like PDAs and palmtops, as the use of keyboard may be infeasible or difficult.

VI. USABILITY ISSUES OF TEXT BASED CAPTCHA

Are text CAPTCHAs like Gimp, user-friendly? Sometimes the text is distorted to such an extent, that even humans have difficulty in understanding it. **Distortion** becomes a problem when it is done in a very haphazard way. Some characters like 'd' can be confused for 'cl' or 'm' with 'rn'. It should also be easily understandable to those who are unfamiliar with the language. **Content** is an issue when the string length becomes too long or when the string is not a dictionary word. Care should be taken not to include offensive words. **Presentation** should be in such a way as to not confuse the users. The font and color chosen should be user friendly[4].

VII. IMPLEMENTATION AND EXPERIMENTAL RESULTS

The system can be implemented using MATLAB R2010a. Here the input to the system is a Captcha image and output is the characters recognized from the image. . Based on the accuracy of recognition robustness of Captcha is measured. A dataset of 345 images were used and a training set of all Captcha images were used for text based class of Captcha. Robustness is measured using the following formula:

Accuracy of Correctness = $\frac{\text{Correctly Recognized}}{\text{Total Number of Text Images}}$

Text Images/Total Number of Text Images

Accuracy of recognized = $\frac{\text{Correctly Recognized no. of Characters}}{\text{Total no. of Characters}}$

Correctness Percentage = $(\frac{\text{Total Number of Correct Recognized Text Images}}{\text{Total Number Of Text Images}}) * 100$

Input set of dataset	Output set of dataset	No. of Test Correct	Serial no. of dataset	% of Test Correct
Etapes	ETAPES	1	1	100
Facial	FACIAL	2	2	100
fasTED	@ @ @ @ @ @	2	3	66.667
Fazing	@ @ @ @ @ @	2	4	50
feisTs	FEISTS	3	5	60
fichEs	@ @ @ @ @ @	3	6	50
finiSh	FINISH	4	7	57.149

VIII. CONCLUSION AND FUTURE WORK

The algorithm addressed in paper successfully breaks above mentioned text based CAPTCHAs. In the experimental results, it was found that algorithm can uniformly improve the segmentation rate over the traditional algorithm. The proposed algorithm makes novel and useful contributions to the field of CAPTCHA analysis. The future work includes evaluation of more Text-based captchas in NON-OCR and to formulate guidelines & design principles for the generation of attack-resistant Captchas. The basic advantage of the developed system is that it can be used as a module in normal OCR, because a normal OCR program PTcould not recognize character in distorted background. Also it can be used to rate and improve the security of various websites

REFERENCES

- [1] L. von Ahn, M. Blum and J. Langford, "Telling Humans and Computer Apart Automatically," in *Communications of the ACM*, vol.47, no. 2, pp. 57-60, 2004
- [2] H.S. Baird and K. Papat, "Human Interactive Proofs and Document Image Analysis," in *Proc. of the 5th IAPR International Workshop on Document Analysis Systems*, Springer LNCS 2423, pp. 507-518, 2002.
- [3] A.L. Coates et al, "Pessimial Print: A Reverse Turing Test," in *Proc. of the 6th International Conference on Document Analysis and Recognition*, Seattle, WA, USA, pp. 1154-1158, 2001.
- [4] Ahmad El Ahmad, Jeff Yan, Wai-Yin Ng, "CAPTCHA Design: Color, Usability, and Security," *IEEE Internet Computing*, vol. 16, no. 2, pp. 44-51, 2012.
- [5] Jeff Yan, Ahmad Salah El Ahmad, "Captcha Robustness: A Security Engineering Perspective," *IEEE Computer*, vol. 44, no. 2, pp. 54-60, 2011.
- [6] www.captchas.net
- [7] www.bigcloudmedia.com
- [8] J. Yan and A. S. El Ahmad. Usability of CAPTCHAs or usability issues in CAPTCHA design. In SOUPS '08, pages 44–52, New York, NY, USA, 2008. ACM.
- [9] Technical Report:Ahmad.Salah-El-Ahmad, Jeff.Yan, Mohomad.Tayara, —The Robustness of Google CAPTCHAs Technical report, Newcastle University, UK, 2011.
- [10] Technical Report:Ahmad.Salah-El-Ahmad, Jeff.Yan, Mohomad.Tayara, —The Robustness of Google CAPTCHAs Technical report, Newcastle University, UK, 2011.