Reading the Moving Text in Animated Text-Based CAPTCHAs

Syed Safdar Ali Shah[1], Riaz Ahmed Shaikh[2], Rafaqat Hussain Arain[3]
Department of Computer Science
Shah Abdul Latif University
Khairpur, Pakistan

***Abstract*—Having based on hard AI problems, CAPTCHA (Completely Automated Public Turing test to tell the Computers and Humans Apart) is a hot research topic in the field of computer vision and artificial intelligence. CAPTCHA is a challenge-response test conducted to single out humans and bots. It is ubiquitously implemented on the web since its introduction. As text-based CAPTCHAs are successfully broken by various researchers therefore several design variants have been proposed and implemented in order to further strengthen it. Animated Text-based CAPTCHAs are one of the design variant of it and are based on the difficulty of reading the moving text. They are based on zero knowledge per frame principle. Although it's still easy for humans to read animated text but it's a challenge for machines. As proposals for animated CAPTCHAs are on the rise so there is a strong need to scrutinize their strength against automated attacks. In this research, such CAPTCHAs are investigated to verify their robustness against automated attacks. The proposed methods proved that these CAPTCHAs are vulnerable and they do not guarantee the robustness against automated attacks. The proposed frame selection, noise removal, segmentation and recognition methods have successfully decoded these CAPTCHAs with an overall precision, segmentation accuracy and recognition rate of up to 53.8%, 92.9% and 93.5% respectively.**

Keywords—Bots; CAPTCHAs; ANNs; animations; image processing; HIPs; machine learning

I. Introduction

The acronym CAPTCHA stands for Completely Automated Public Turing test to tell Computers and Humans Apart. This term was coined by Ahn et al. in their pivotal publication in 2000 to thwart the web against bots [1]. CAPTCHAs are also known as HIPs (Human Interaction Proofs). According to one report by BBC, 61% of web traffic is generated by bots [2]. Therefore stopping the bots and securing the web traffic is indispensable. Now CAPTCHA is a standard security mechanism on the web to identify the human interaction. It is based on hard AI (Artificial Intelligence) problems. These are the problems which are supposed to be fairly easy for humans but extremely difficult for machines. As the CAPTCHAs are based on hard AI problems, they have emerged as a hot research topic in the fields of computer vision and artificial intelligence. Additionally as 'P' stands for public [3], means the underlying algorithm to create the test should be open to research community as stated by Ahn et al. This statement by the pioneers of the field motivates the researchers to work in this field.

Over the years, many design variants are introduced by CAPTCHA designers. On the other hand, attackers have successfully decoded these CAPTCHAs. As the existing CAPTCHA schemes are broken by the attackers, new design variants are introduced by designers. Therefore it is an ongoing war since the introduction of first CAPTCHA [4]. However in both cases it's a victory. If the CAPTCHA is successfully broken then a hard AI problem is solved which leads to one step forward in machine learning. On the other hand if a CAPTCHA is found resistant against automated attacks then a way to distinguish between human and machines is devised which provides a security mechanism on the web.

A large number of static text-based CAPTCHAs are successfully broken by various researchers [5-8]. Therefore a growing number of animated CAPTCHAs are proposed by many researchers. Animated text-based CAPTCHA are an alternative to static Text-based CAPTCHAs. In this type of CAPTCHA, the user is asked to read the moving text as shown in Figure 1. This type of test presents animated text typically in Graphics Interchange Format (GIF) images. However other formats like flash files and streaming videos can also be used to present animated CAPTCHAs. The information is usually, spread in multiple frames rather than presenting it in a single frame like 2D static CAPTHCAs. It is assumed that adding the time dimension in this type of CAPTCHAs makes it more secure than its counterparts, i.e. 2D static CAPTCHAs. As various attackers have successfully attacked the static text-based CAPTCHAs but very little research is carried out to verify the robustness of animated text-based CAPTCHAs. In this research, the robustness of these CAPTCHAs is verified. There is a dual benefit of this research, it not only helps to identify the design flaws in current animated CAPTCHAs but the proposed algorithms can also be used to read the moving text. By implementing the proposed frame selection, noise removal, segmentation and machine learning methods, targeted CAPTCHAs are successfully broken as discussed in section 3.

Rest of the paper is organized as follows; Section 2 presents literature review of the field. Section 3 presents the proposed frame selection, noise removal, and segmentation and recognition methods. Section 4 presents the results of the proposed methods, finally section 5 presents the conclusion and future work.

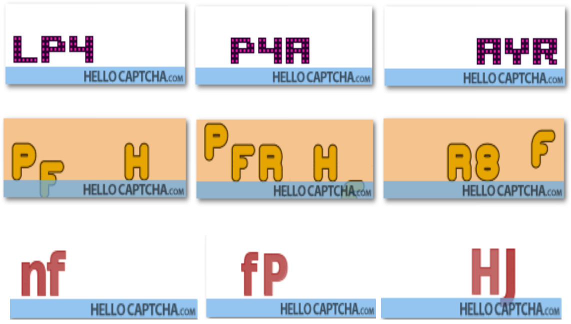

Fig. 1. Animated Text-Based Captchas.

II. Literature Review

Since its introduction, numerous design variants of CAPTCHAs are introduced by researchers. However text-based CAPTCHAs are still most prevalent due to simplicity and ease of use [9]. As text-based CAPTCHAs are successfully attacked by many researchers, therefore they have evolved over the years. Various extraction, segmentation and recognition resistance schemes have been introduced to improve their robustness against automated attacks. In spite of many resistance schemes, various popular CAPTCHAs offered by Yahoo, Google and MSN have been successfully broken [7][10][11]. It has led to the developers to introduce new design alternatives. Animated text-based CAPTCHA is an alternative to static text-based CAPTCHAs. In this CAPTCHA the user is asked to read the text which is spread in various frames.

Cui et al. termed it as Zero Knowledge Per Frame Principle [12]. They have proposed an animated CAPTCHA scheme containing moving letters on a noisy background. Fischer and Herfet proposed an idea of presenting the text on a deforming surface [13]. Chow and Susilo proposed an animated CAPTCHA scheme based on motion parallax [14]. In this scheme, segmentation resistance is used in moving text, humans can still identify the individual characters while they move However it is supposed as a difficult job for machines. Creo Group has introduced a HelloCAPTCHA scheme, which spreads the information in various frames [15]. Naumann et al. introduced a similar CAPTCHA scheme where letters and other sketches move on a noise background, they become observable while moving in different areas of an image [16].

Ince et al. introduced an interesting 3D CAPTCHA scheme where the users are presented with a randomly selected text [17]. The user is asked to type the characters/numbers in color input boxes read from each side of a multicolor 3D cube. Chaudhari et al. presented an idea of a 3D drag n drop CAPTCHA [18]. The user is presented with a randomly generated 3D text. Instead of typing the test, the user is asked to drag n drop the individual letters in boxes. Susilo et al. introduced a CAPTCHA which is built from stereoscopic 3D images [19]. They stated that the distorted and overlapped 3D text in stereoscopic images will increase the complexity against automated attacks. Imsamai and Suphakant proposed a 3D CAPTCHA scheme, where multiple factors like rotation, overlapping, distributed noise etc. were added in alphanumeric characters to improve its robustness against bots [20].

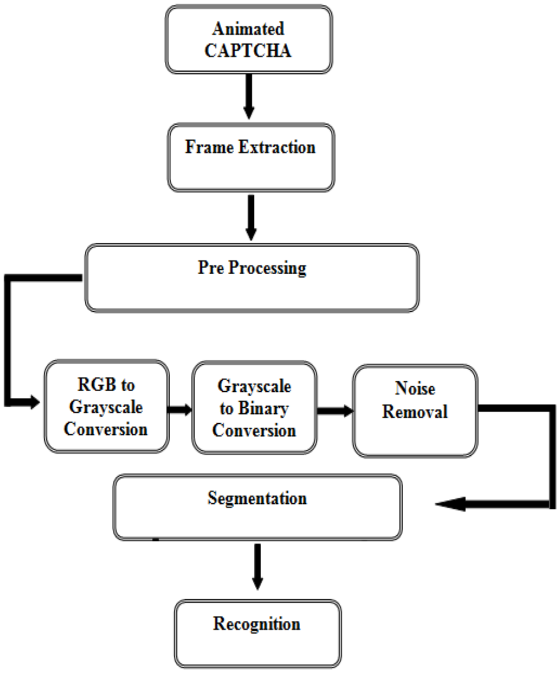

Fig. 2. Sytem Diagram of the Proposed Algortihm.

III. Proposed Method

In this research, HelloCAPTCHA[15] is selected as representative of animated CAPTCHA scheme because it offers a variety of design variants of animated CAPTCHAs. Three different schemes were selected, i.e. flitter, Popup and Smarties CAPTCHAs as shown in Figure 1. All the challenged images are GIF images containing a certain number of frames. These frames are displayed after a fixed duration of time. In this way, animation of characters is achieved. The system diagram of the proposed method is shown in Figure 2.

A. *Frame Selection*

As the animated CAPTCHAs present the challenge in multiple frames instead of presenting them in a single frame like static text based CAPTCHAs. Time dimension is therefore added as an extra layer of security. Therefore the selection of frames containing the challenged images of characters is a research problem. Once the required frames can be selected, the problem can be reduced to static CATPCHA. We have analyzed the attacked CAPTCHAs and serious design weaknesses were found in them. These design weaknesses were exploited to break these CAPTCHAs successfully with high precision as discussed in section 4.

All attacked types of CAPTCHAs are displayed in the size of 180x60 RGB images. Each challenge consists of 6 alphanumeric characters. Flitter CAPTCHAs consists of 173 to 177 frames. Characters appear after specified interval of time at fixed columns in the animation. The major weakness in this design is the fixation of columns and time intervals to display the individual characters. After a certain time period, the individual characters are displayed and we measured this time in flitter images as approx. 55 mil Seconds. Another major weakness is to display a complete character after every

20th frame. These weaknesses are exploited to extract characters from the animated CAPTCHAs. In Popup CAPTCHA scheme, the characters popup in the image at certain columns and then disappear. The animation is achieved by means of moving 80 to 85 frames at regular intervals of time, although the characters popup randomly but stay in the image for a certain period of time in order to catch the human's attention. The individual characters remain appeared in the fixed columns and hence extracted by calculating the amount of time. Smarties CAPTCHAs present each character after a certain interval of time. Every character is displayed at the screen at fixed locations. It remains appeared at that location for a certain period of time and then disappears. Once the 3rd character is appeared then the first is disappeared. The animation is achieved by means of moving 190 to 200 frames in a GIF file. Multiple such CAPTCHAs are analyzed and regular patterns in their appearances were found. These design weaknesses in attacked CAPTCHAs are exploited to successfully decode them. Once the required frames are extracted and labelled then further operations are performed on them, like preprocessing, segmentation and finally the recognition.

B. *Preprocessing*

The obtained frames contain the disconnected characters and the step of segmentation is fairly simple in the selected types of CAPTCHAs. These images are firstly converted to grayscale images as shown in Figure 3.

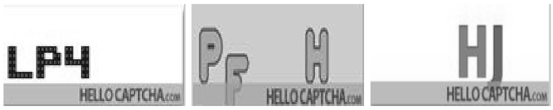

Fig. 3. RGB to Grayscale Conversion.

The obtained Grayscale images are converted to binary images using Equation No.1

Y = 0.2989*R + 0.5870*G + 0.1140*B (1)

Figure 4 shows the results of grayscale to binary conversion.

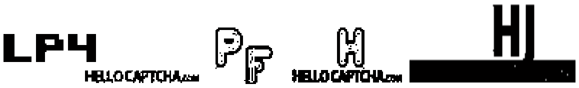

Fig. 4. Grayscale to Binary Conversion.

The obtained binary images contain noise as well footer of the designer's name. This noise can affect the results in the later steps of segmentation and recognition. The footer containing 'HelloCAPTCHA.com' always appears at a fixed location and it is easily removed by calculating its area. Rest of the noise is salt and pepper noise. A threshold value 't' is used to remove all the pixels having values smaller than 't'. Figure 5 shows the results after noise removal.

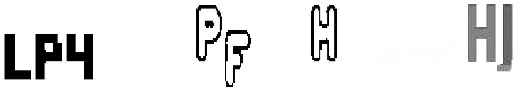

Fig. 5. Results after Noise Removal.

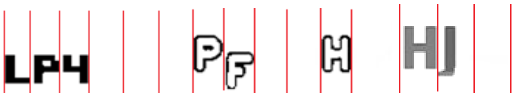

Fig. 6. Character Segments after Segmenation.

C. *Segmentation*

Segmentation aims to separate the individual characters. In the obtained binary images (after noise removal) there is no overlapping of characters. These images therefore can be easily segmented using the condition of blank columns. There are multiple blank columns (columns containing no black pixels) in the binary images and we can obtain the character segments as shown in Figure 6.

The character segments are labeled to store their positions in the string in order to reconstruct the string as output of the process.

D. *Recognition*

In the previous step of segmentation, 200 samples for each type of CAPTCHA were segmented. Therefore Approx. 1200 images of individual characters were obtained. These images of individual characters are used to train an Artificial Neural Network (ANN) using Matlab 9.2. The said ANN is trained using Scaled Conjugate Gradient Algorithm with backpropagation. In order to calculate the performance of the network cross entropy is used for the given targets and outputs. The data is randomly divided into training, validation and testing datasets as 70%, 15% and 15% respectively.

The segmented images of individual characters are normalized. ANN is constructed by calculating the feature vectors of the normalized images by calculating the local and global features of character skeleton [21].

IV. Results and Discussions

Overall success rate of the proposed algorithm depends on segmentation accuracy, recognition success rate of the classifier (ANN) and the number of characters in a challenged image. Therefore Equation 2 is used to calculate the overall precision [22].

OP (%) = SSR% (%) (2)

Where OP is the Overall Precision, SSR is the Segmentation Success Rate, SRR is the Success Recognition Rate. SSR depends on the numbers of characters correctly segmented in a dataset of images. For example if a segmentation algorithm can segment 500 characters in 100 images of Flitter CAPTCHAs (100*6, as there are 6 characters in each image) then the segmentation accuracy would be 500/600 = 0.833 or 83.3%. SRR depends on the accuracy of the classifier. Table 1 shows the results of the proposed algorithm.

TABLE I. Results of the Proposed Algorithm

CAPTCHA	**SSR (%)**	**SRR (%)**	**OP (%)**
Flitter	73.3	93.5	48.97
Popup	82.5	89.3	41.83
Smarties	92.9	91.3	53.8

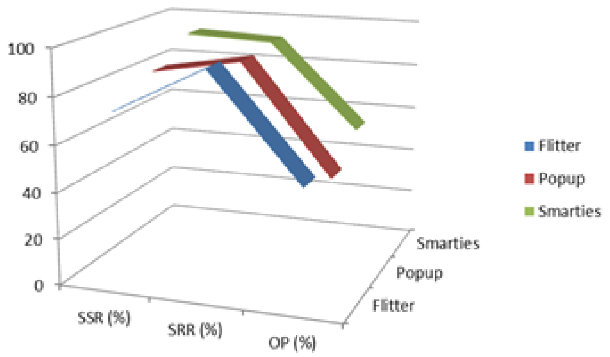

Fig. 7. Segmentation, Recognition and OP Rates.

As mentioned in [23], the CAPTCHA is assumed to be broken if an automated program can decode it with an accuracy of even 1%. Figure 7 displays the results in a graph where it can be observed that our segmentation, recognition and overall precision rates are clearly far beyond this imagination of an ideal CAPTCHA.

V. Conclusion and Future Work

In this paper, three popular animated CAPTCHA schemes offered by HelloCAPTCHA were successfully decoded. An overall precision, segmentation accuracy and recognition rate of up to 53.8%, 92.9% and 93.5% respectively were achieved. Simple but robust image processing techniques were applied to successfully decode them with high accuracy. Serious design weaknesses were exploited in the attacked CAPTCHA schemes. Regular patterns such as fixed columns for certain number of characters and regular time intervals for the appearances of certain characters makes these schemes vulnerable against automated attacks. Furthermore it was validated that the addition of time dimension does not guarantee the robustness against automated scripts.

In future, the robustness of other animated CAPTCHA schemes can be verified, which offer segmentation resistant principles along with addition of time dimensions in their proposed schemes.

References

[1] L. V. Ahn, M. Blum, J. Langford. Telling humans and computers apart automatically. Communications of the ACM, 2004, 47(2): 56-60.

[2] British Broadcasting Corporation. BBC News Technology. http://www.bbc.co.uk/news/technolgy-25346235, Dec.2013.

[3] K. A. Kluever and R. Zanibbi. Balancing usability and security in a video CAPTCHA. SOUPS, ACM international conference proceeding series. ACM. 2009.

[4] R. Hussain, H. Gao, R.A. Shaikh. Segmentation of connected characters in Text-based CAPTCHAs for intelligent character recognition. Multimedia Tools and Applications, 76(24), pp. 25547-25561, 2017.

[5] G. Mori, J. Malik. Recognizing objects in adversarial clutter: Breaking a visual CAPTCHA. International conference on Computer Vision and Pattern Recognition Proceedings, Washington, 2003, 134-141.

[6] K. Chellapilla, P. Y. Simard. Using machine learning to break visual human interaction proofs. 17th International Conference on Neural Information Processing Systems Proceedings, Columbia, 2004, 265-272

[7] A. S. El Ahmad, J. Yan, M. Tayara. The robustness of Google CAPTCHA's [EB/OL]. Computing Science Newcastle University, 2011.

[8] O. Starostenko, C. Cruz-Perez, F. Uceda-Ponga, et al. Breaking text-based CAPTCHAs with variable word and character orientation. Pattern Recognition, 2015, 48(4): 1101-1112.

[9] J. Yan, A. S. El Ahmad. Usability of captchas or usability issues in captcha design. 4th symposium on Usable privacy and security Proceedings, New York, 2008, 44–52.

[10] S. Y. Huang, Y. K. Lee, G. Bell, Z. H. Ou. An efficient segmentation algorithm for CAPTCHAs with line cluttering and character warping .Multimedia Tools and Applications, 2010, 48(2): 267-289.

[11] H. Gao, X. Wang, F. Cao, et al. Robustness of text-based completely automated public Turing test to tell computers and humans apart. IET Information Security, 2016, 10(1): 45-52.

[12] J. S. Cui, J. T. Mei, W. Z. Zhang, et al. A CAPTCHA implementation based on moving objects recognition problem. International conference on E-Business and E-Government Proceedings, 2010 Guangzhou, 2010, 1277-1280.

[13] I. Fischer, H. Thorsten. Visual CAPTCHAs for document authentication[C]. IEEE workshop on Multimedia Signal Processing Proceedings, Victoria, 2006, 471-474.

[14] Y. W. Chow, W. Susilo. AniCAP: An animated 3D CAPTCHA scheme based on motion parallax. International Conference on Cryptology and Network Security Proceedings, Sanya, 2011, 255-271.

[15] Crew Group. HelloCAPTCHA vs Spambots. http://www.hellocaptcha.com, 2018, last viewed on 10-Oct-2018.

[16] A. B. Naumann, F. T. Franke, B. Christian. Investigating CAPTCHAs Based on Visual Phenomena. IFIP Conference on Human-Computer Interaction Proceedings, Uppsala, 2009, 745-748.

[17] I. F. Ince, Y. B. Salman, M. E. Yildirim, et al. Execution time prediction for 3d interactive captcha by keystroke level model. Fourth International Conference on Computer Sciences and Convergence Information Technology Proceedings, Seoul, 2009, 1057-1061.

[18] S. K. Chaudhari, A. R. Deshpande, S. B. Bendale, et al. 3D drag-n-drop CAPTCHA enhanced security through CAPTCHA. International Conference & Workshop on Emerging Trends in Technology Proceedings, Mumbai, 2011, 598-601.

[19] W. Susilo, C. Yang-Wai, Z. Hua-Yu. Ste3d-cap: Stereoscopic 3d captcha. International Conference on Cryptology and Network Security, Kuala Lumpur Proceedings, 2010, 221-240.

[20] M. Imsamai, P. Suphakant. 3D CAPTCHA: A next generation of the CAPTCHA. International conference on Information Science and Applications, (ICISA) Proceedings, Ho Chi Minh, 2010, 1-8.

[21] D. D. Gaurav, & R. Ramesh. "A feature extraction technique based on character geometry for character recognition" arXiv preprint arXiv, 2012, 1202.3884.

[22] Starostenko, O., Cruz-Perez, C., Uceda-Ponga, F., & Alarcon-Aquino, V. (2015). "Breaking text-based CAPTCHAs with variable word and character orientation" *Pattern Recognition*, *48*(4): 1101-1112.

[23] Yan, J., & El Ahmad, A. S. (2008). A Low-cost Attack on a Microsoft CAPTCHA. In Proceedings of the 15th ACM conference on Computer and communications security (pp. 543-554) ACM.