7-2-2010

Drag and Drop Image CAPTCHA

Nisar A. Shah
University of Kashmir, nassgr@yahoo.com

M. Tariq Banday
University of Kashmir, sgrmtb@yahoo.com

芽|Sprouts Working Papers on Information Systems ISSN 1535-6078

Drag and Drop Image CAPTCHA

Nisar A. Shah
University of Kashmir, India

M. Tariq Banday
University of Kashmir, India

Abstract
The massive and automated access to Web resources through robots has made it essential for Web service providers to make some conclusion about whether a "user" is human or robot. A Human Interaction Proof (HIP) like Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) offers a way to make such a distinction. CAPTCHA is essentially a modern implementation of Turing test, which carries out its job through a particular text based, image based or audio based challenge response system. In this paper we present a new image based CAPTCHA technique. Properties of the proposed technique offer all of the benefits of image based CAPTCHAs; grant an improved security control over the usual text based techniques and at the same time improve the user-friendliness of the Web page. Further, the paper briefly reviews various other existing CAPTCHA techniques.

Keywords: CAPTCHAs, HIP, Botnet, Image CAPTCHA

Permanent URL: http://sprouts.aisnet.org/8-46

Reference: Shah N.A., Banday M.T. (2008). "Drag and Drop Image CAPTCHA ," . *Sprouts: Working Papers on Information Systems*, 8(46). http://sprouts.aisnet.org/8-46

Drag and Drop Image CAPTCHA

Prof. N. A. Shah
Dept. of Electronics and Instrumentation Technology
University of Kashmir, Srinagar, India
Email: nassgr@yahoo.com

M. Tariq Banday, *Lifetime Member,* CSI
Dept. of Electronics and Instrumentation Technology
University of Kashmir, Srinagar, India
Email: sgrmtb@yahoo.com

Abstract

The massive and automated access to Web resources through robots has made it essential for Web service providers to make some conclusion about whether a "user" is human or robot. A Human Interaction Proof (HIP) like Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) offers a way to make such a distinction. CAPTCHA is essentially a modern implementation of Turing test, which carries out its job through a particular text based, image based or audio based challenge response system. In this paper we present a new image based CAPTCHA technique. Properties of the proposed technique offer all of the benefits of image based CAPTCHAs; grant an improved security control over the usual text based techniques and at the same time improve the user-friendliness of the Web page. Further, the paper briefly reviews various other existing CAPTCHA techniques.

Keywords

CAPTCHAs, HIP, Botnet, Image CAPTCHA

1. Introduction

Atomizing various Web activities by replacing human to human interactions on the Internet has been made indispensable due to its enormous growth. However, bots [1] also known as robots and Web-bots which have a malicious intend and pretending to be humans pose a severe threat to various services on the Internet that implicitly assume a human interaction. Accordingly Web service providers before allowing access to such services use various HIPs [2] to authenticate that the user is a human and not a bot. CAPTCHAs [3], a class of HIPs are tests which are easier for humans to qualify and tough for bots to simulate. Several Web services that include but not limited to free e-mail, online polls, chat rooms, search engines, blogs, password systems, etc. use CAPTCHAs as a defensive mechanism against automated Web bots.

The remaining text of this paper is organized as follows: Section 2 presents a short review of bots and existing CAPTCHA techniques. Section 3 illustrates the proposed Drag and Drop CAPTCHA technique. In section 4 security analysis of the proposed technique is discussed. Finally, we conclude and present future research directions.

2. Web bots and CAPTCHA

Web bots are scripts or applications designed to perform predefined functions repeatedly and automatically after being triggered intentionally or through a system infection. Although bots originated as a useful feature in carrying out repetitive and time consuming operations but its ability to imitate human behavior has been exploited for malicious intent. According to the level of complexity in the operations performed by a bot it may be classified as a 1st generation, 2nd generation or 3rd generation program [4]. 1st generation bots are used to download a predefined set of resources without interpreting their content. 2nd generation bots are capable to analyze downloaded content and perform other actions on the basis of the meanings derived from that content. 3rd generation bots are capable of fully interpreting client side script languages such as VBScript, Java Script, and Flash. Further, 3rd generation bots can derive meaning from the downloaded content with intelligence similar to that of a human user. Several anti-bot defense strategies that include HTTP client and server side strategies have been developed to protect Web applications from these bots. HTTP based strategies have been able to protect Web applications to a larger extend from 1st generation bots and to some extend from 2nd generation bots but cannot be used against 3rd generations of bots [5]. Human Interaction Proof (HIP) which are schemes that require some kind of interaction from a human in order to be recognized as a human or a member of a group [6] have been able to effectively prevent malicious programs from getting access to the Web services. CAPTCHAs is a class of HIP tests and are easier for humans to qualify and tough for bots to simulate. CAPTCHAs underlying hardness is based on Artificial Intelligence and the test may be text based, image based or audio based. A good CAPTCHA should be generated in a manner that they satisfy a set of desired properties [7] that include: a) the test must be generated automatically, b) the answer to the test should be quick and easy, c) the test should accept all human users and d) the test should resist attacks even if the algorithm is known. CAPTCHAs are used in diverse Web services for securing online polls, preventing spammers from getting free e-mail accounts, preventing Bot entry into the chat system, preventing online dictionary attacks in password systems, preventing unruly search engine bots from indexing private Web pages, preventing Web bots from adding advertisements to comment field in blogs, preventing download bots from downloading and archiving Web sites or FTP servers. Apart from offering advantages enlisted above, CAPTCHAs pose some disadvantages that include unfriendliness, requirement for larger image library, increased load on servers, delays in Web page download, accessibility problems and annoyance to genuine users. The following paragraphs summarize various existing CAPTCHA techniques.

Andrei Broder and his colleagues devised the CAPTCHA method in 1997 and in the same year Altavista Website used this method as a HIP. This method used a distorted English word that a user was asked to type. The distorted word was easier for users to understand but difficult for bots to recognize using OCR techniques. The Altavista Website CAPTCHA is shown in Figure 1 below.

Figure 1: Altavsta Website CAPTCHA

I. EZ-Gimpy and Gimpy CAPTCHA [8]

These techniques are based on OCR and were originally proposed by Blum and Von Ahn at Carnegic Melton University in Collaboration with Yahoo to protect chat rooms from spammers. Gimpy works by selecting several words from a dictionary and displays them corrupted and distorted in an image to gain entry to the service. These CAPTCHAs due to limited words in its dictionary (860 words) have been broken [9]. Gimpy and EZ-Gimpy CAPTCHA are shown in figures 2 and 3 respectively. Until year 2004 Yahoo had been

using this method for chat room protection. Yahoo is currently using another new method which is show in figure 4.

Figure 2: Gimpy CAPTCHA

Figure 3: EZ-Gimpy CAPTCHA

Figure 4: Yahoo Website CAPTCHA

Recently a new and a more secure type of text based HIP, called reCAPTCHA [10] shown in figure 5 has been proposed by the authors of EZ-Gimpy CAPTCHA.

Figure 5: reCAPTCHA

II. Baffle Text CAPTCHA [11]

This is the Xerox Pato Alto Research Center (PARC) version of Gimpy test. Baffle Text uses small pseudorandom pronounceable words to defeat dictionary attacks. It exploits gestalt psychology which posits that humans are very good at filling in missing portions of an image while computers are not. Sample Baffle Text CAPTCHA is shown in figure 6. Since this method can use words with a high degree of difficulty, the produced words are difficult for humans to distinguish. A CAPTCHA named Scatter Type is related to Baffle Text however it is different from this due to its legibility.

Figure 6: Baffle Text CAPTCHA

III. Handwritten Word based CAPTCHA [12]

In this method a database formed of handwritten names of American cities selected from letters posted by people, is used. A picture of the randomly selected city word is shown to the user who has to type the letters to gain access to the service. The words as shown in figure 7 are of poor quality to read for the human user.

Figure 7: Handwritten Word based CAPTCHA

IV. HVS Masking Characteristic CAPTCHA [13]

The Human Visual System masking Characteristic CAPTCHA sample of which is shown in figure 8 is composed of English alphabets that are picked randomly and written with a combination of texture and edges with added noise such as to deceive the bots by randomly choosing the visibility of characters.

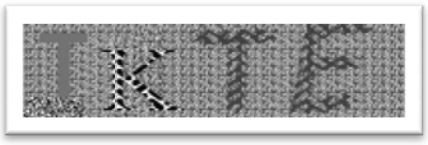


Figure 8: HVS Masking Characteristic CAPTCHA

V. PayPal CAPTCHA [14]

This CAPTCHA is used in PayPal Website which is an electronic money payment service. PayPal has not provided any detail on this method however due to a huge distance between the letters it may be quite possible to break this test using a good OCR program. A sample PayPal CAPTCHA is shown in figure 9.

Figure 9: PayPal Website CAPTCHA

VI. Hotmail CAPTCHA [15]

Hotmail Website CAPTCHA test uses a series of English letters selected randomly. The shapes of letters are modified and a word is formed which a user has to correctly type before availing the service on the Web site. The segmentation of letters is made complex by adding arcs so as to make it difficult for OCR programs to recognize the letters. This method poses some difficulty to the user in recognizing the letters due to complex segmentation of the letters. A sample of this CAPTCHA is shown in figure 10.

Figure 10: Hotmail Website CAPTCHA

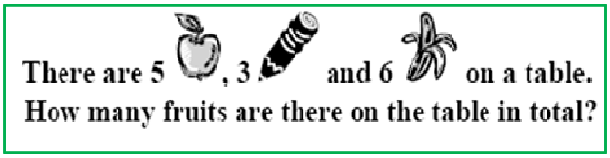
VII. Persian/Arabic CAPTCHA [16]

This method as shown in figure 11 suggest the use of Persian Arabic words in which connected letters, presence of dots and the right to left reading order makes programs to fail. The major drawback with this method is its limited domain of users.

Figure 11: Persian/Arabic CAPTCHA

VIII. Question Based CAPTCHA [17]

This test is an extension of simple question based test and proposes a simple mathematical problem according to a predefined pattern but instead of some object name, image of the object is placed. Figure 12 shows a sample of this CAPTCHA test.



IX. ESP-PIX CAPTCHA [3]

This CAPTCHA was initially proposed by Blum and Von Ahn and uses a larger database of photographs and animated images of everyday objects. The CAPTCHA system presents a user with a set of images all associated with the same objects or concept. The user must enter the object or concept to which all the images belong to e.g. the program might present pictures of Globe, Volleyball, Planet and baseball expecting the user to correctly associate all these pictures with the word ball. See the figure 13 for its sample representation.

Figure 13: ESP-PIX CAPTCHA

X. Bango CAPTCHA [18]

This CAPTCHA as can be seen from its sample in figure 14 uses a visual pattern recognition problem. It uses two sets of images, each set having some specific characteristics. One set might be boldface while the other is not. The system then presents a single image to the user who then must specify the set to which the image belongs. Since the number of possible solutions is small it is not robust to brute-force guessing [8].

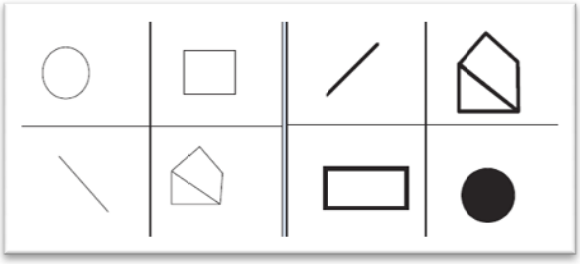


Figure 14: Bango CAPTCHA

XI. Microsoft Asirra CAPTCHA [19]

This is a CAPTCHA designed by Microsoft Corporation which uses animal species image recognition for restricting access and challenges the user to prove its humanity by selecting all images of specific specie among the set of pictures shown to the user. Asirra does not use any image transformation and its database is

not public. Another similar CAPTCHA is KittenAuth [20]. A sample run of Microsoft Asirra CAPTCHA method is shown in figure 15.

Figure 15: Microsoft Asirra CAPTCHA

XII. IMAGINATION CAPTCHA [21]

IMAge Generation for INternet AuthenticaTION uses a database of images of simple concepts and is composed by two subsequent tests, based on different Artificial Intelligence problems. This CAPTCHA appears quite effective but requires a human user to store two distinct tests.

XIII. Motion CAPTCHA [16]

In this method a movie describing some form of a moment is shown to the user. The user is next asked to select the option that best describes the movement. This method requires a very huge database to store sentences and also chances of brute-force guessing are very high. A sample of this CAPTCHA is shown in figure 16.

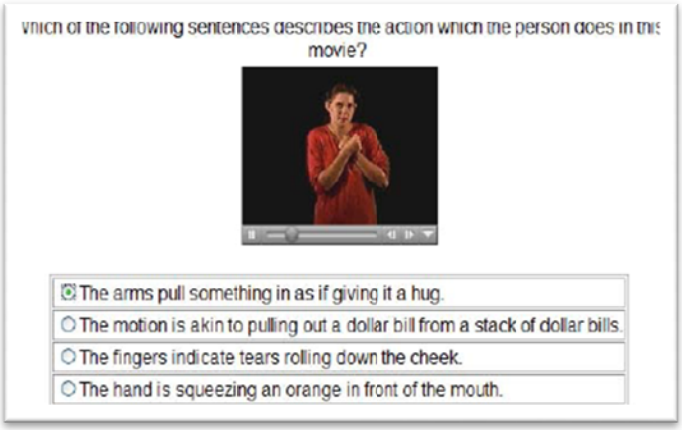


Figure 16: Motion CAPTCHA

XIV. Drawing CAPTCHA [23]

In this method a large number of dots with a few distinguishable ones with some noise added to it are shown to the user. A few distinguishable dots differ from other by making them to have small holes in center or making them rectangular. The user is asked to connect these distinguishable dots to gain access to the

service. This method is apparently consuming a larger space on the Web page and is also prone to be broken by bots using intermediate image processing algorithms. Figure 17 shows a sample of this technique.

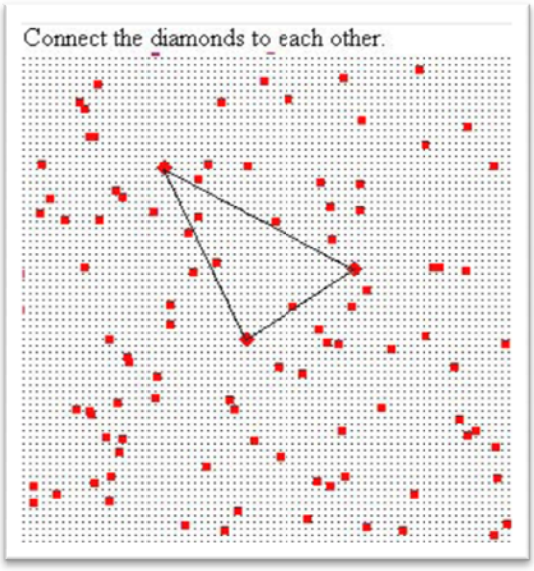


Figure 17: Drawing CAPTCHA

XV. Collage CAPTCHA [24]

In this CAPTCHA test a set of randomly chosen pictures selected from a picture database of objects like animals, persons, flags, etc. after applying a rotation is shown to the user at random places without overlapping within the CAPTCHA frame. The user is asked to select a particular picture as a Human Interaction Proof. A sample of this test is shown in figure 18.

An advanced version of this test known as Advanced College CAPTCHA [25] has been proposed by same authors, wherein the CAPTCHA method displays another set of pictures but with different shapes in a similar form. The user is required to choose the match object as the one chosen previously to prove human interaction. In another version of Collage CAPTCHA known as online college CAPTCHA [26], the picture database is dynamically formed by downloading pictures from the Internet. A sample image of this CATCHA test is shown in figure 19. Yet another variation of this technique named Multilingual CAPTCHA [27] uses a multilingual interface. Messages or names of the objects are shown in one of many languages.

Figure 18: Collage CAPTCHA

Figure 19: Advanced collage CAPTCHA

XVI. Image Block Exchange CAPTCHA [20]

In this CAPTCHA test as shown in figure 20 an image is randomly chosen from the image database. On this image two non-overlapping blocks of the same size are exchanged and is shown to the user. In order to pass the test, the user must click on the switched regions.

Figure 20: Image Block Exchange CAPTCHA

XVII. Face Recognition CAPTCHA [29]

This technique requires the user to recognize some image of a subject (face of a human being) with two distortions applied to it. As an extension to this technique, different photos of the same individual to which different distortions are applied respectively can be used. Figures 21 and 22 show sample images of this technique.

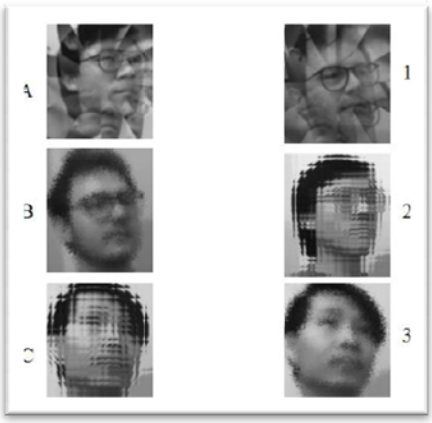


Figure 21: Face Recognition CAPTCHA (Same object with different distortions)

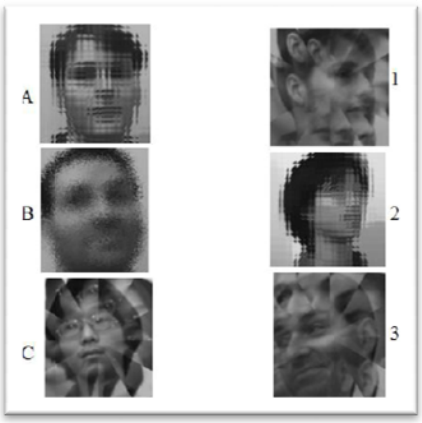


Figure 22: Face Recognition CAPTCHA (Different images of same object)

XVIII. Audio Based CAPTCHA [301]

Audio CAPTCHAs generally take a random sequence drawn from recordings of simple words or numbers, combine them and add some disturbance and noise to it. The CAPTCHA system then asks the user to enter the words and/or numbers in the recording. The first audio based CAPTCHA was implemented by Nancy Chan. Audio CAPTCHAs are more difficult to solve, hard to internationalize and more demanding in terms of time and efforts in comparison to text and image CAPTCHAs. However, audio base tests have become an alternative for visually impaired people.

3. Proposed Method

Although text based CAPTCHA are currently most widely used ones, but the advances in OCR techniques in terms of pattern recognition and computer vision have made them prone to more and more attacks [31,32,33]. Thus, it is reasonable to create new CAPTCHA challenges that are both unbreakable and usable. In this section we present a new CAPTCHA technique that aims to determine legitimate users and at the same time does not alienate them. The scheme works on current difficulty of image segmentation in presence of a complex background. The algorithm implementation and security analysis has been made using GIMP (GNU Image Manipulation Program) [36].

In the proposed technique a composite CAPTCHA image of a reasonable dimension and resolution is shown to the user. The user has to identify two small and simple embedded images (source image and target image) from the shown composite image as asked in the message appearing in the composite image itself. The user has to drag the source image and drop it over the target image to prove human interaction. This scheme is depicted in the figures 23 and 24. Both images have been developed using same algorithm but to image shown in figure 24 a background image with a desired transparency factor has been applied.

Figure 23: Drag and Drop CAPTCHA (Image 1)

Figure 24: Drag and Drop CAPTCHA (Image 2)

A subject-wise database of small sized, well known small real world objects is created. The images and the corresponding tags associated with each image are stored in separate places which are related with each other on an encoded key within a single database so that even if a hacker is able to break into the database, he is not able to get any meaningful content. An as option a dynamic image database could be created by downloading images from the Internet [34, 35]. A CAPTCHA image is created by placing selected images from this database on a larger image to which some complex background has been applied. Various transformations are applied to each selected image before placing it on the larger image. A text message describing which object a user needs to drag and drop over which other object is embedded in CAPTCHA image. Coordinates of the image to drag and that of the image over which to drop are preserved at the server. The CAPTCHA image thus created is shown to the user to prove his humanity. Once the user performs the correct drag and drop, he gains entry to the service otherwise after a few failed attempts a new CAPTCHA image is shown to the user.

The algorithm for creation of Drag and Drop image CAPTCHA is mentioned hereunder:

Step I: Create a composite CAPTCHA image C_{img} of size N X M pixels with a color gradient/RGB noise between randomly chosen colors from RGB.

Step II: Draw *Z* random colored and /or filled and random sized shapes of circles, arcs and lines at random places on the CAPTCHA image C_{img} created in step I. Choose Z as any value between Z_{min} and Z_{max} *depending* upon the required complexity of the CAPTCHA image.

Step III. Apply a required transparency factor to the CAPTCHA image C_{img}. Optionally choose an image from the image database $\boldsymbol{Img_{DB}}$ and place it on C_{img} with a desired transparency factor.

Step IV. Choose $\boldsymbol{P}$ images from the image database $\boldsymbol{Img_{DB}}$ and add them to the set of selected images $\boldsymbol{Sel_{imgs}}$. $\boldsymbol{P}$ *should satisfy the following relationship with the CAPTCHA image* $\boldsymbol{C_{img}}$.

$$P \leq \frac{Sizeof(\boldsymbol{C_{img}})}{MaxSizeof(\boldsymbol{Sel_{imgs}})}$$

Step V: Randomly choose two images from the selected set of images $\mathbf{Sel_{imgs}}$ as the source image $\boldsymbol{S_{img}}$ and the target image $\boldsymbol{T_{img}}$.

Step VI: For each image in $\boldsymbol{Sel_{imgs}}$ apply a random rotation factor $\mathbf{R_F}$, transparency factor $\mathbf{T_F}$ and scaling factor $\mathbf{S_F}$ such that:

$$RMin_F \leq R_F \leq RMax_F ,$$

$$TMin_F \leq T_F \leq TMax_F \text{ and}$$

$$SMin_F \leq S_F \leq SMax_F$$

Step VII: Place each image from the selected set of images $\boldsymbol{Sel_{imgs}}$ including the source image $\boldsymbol{S_{img}}$ and target image $\boldsymbol{T_{img}}$ on the CAPTCHA image $\boldsymbol{C_{img}}$ without overlapping at random positions. At the same time preserve the coordinates of the $\boldsymbol{S_{img}}$ source image and the target image $\boldsymbol{T_{img}}$ in a set $\boldsymbol{C}$.

Step VIII: Retrieve the tags corresponding to the source image $\boldsymbol{S_{img}}$ and target image $\boldsymbol{T_{img}}$ from the image database $\boldsymbol{Img_{DB}}$ and generate the text message.

Step IX: Place the text message string at desired location on the CAPTCHA image $\boldsymbol{C_{img}}$ after applying a transparency and rotation factor to it.

Step X: Return the CAPTCHA image $\boldsymbol{C_{img}}$ and the coordinate set $\boldsymbol{C}$ of source image $\boldsymbol{S_{img}}$ and target image $\boldsymbol{T_{img}}$.

4. Security Analysis

Security of a CAPTCHA technique can be analyzed in terms of time taken, resources involved and efficiency of a bot program that tries to breach the test. Any CAPTCHA technique is considered to be secure that is at least as expensive for a hacker as it would cost him using human operators. In this section we will present the security analysis performed on the proposed CAPTCHA technique.

Segmentation of an image in regions, identification of regions of interest and extraction of semantic content expressed by the image or part of it are the various steps involved in Content Based Image Retrieval (CBIR) methods. CBIR methods may use an edge detection technique for segmentation. Figures 25, 26, 27

and 28 show images after applying famous Edge Detection Techniques namely Difference of Gaussian and Laplace to the images produced with Drag and Drop CAPTCHA method shown in figures 23 and 24.

Figure 25: Edge Detection (Difference of Gaussians) Image 1

Figure 26: Edge Detection (Difference of Gaussians) Image 2

Figure 27: Edge Detection (Laplace) Image 1

Figure 28: Edge Detection (Laplace) Image 2

It is clear that no doubt the edge detection technique has extracted some features of each sub image; however, the presence of complex background does not allow CBIR methods to draw any semantics from it.

Another possibility of segmentation is by using thresholding which provides information about the statistical distribution of color values. An intensity Histogram can be used to look into this possibility. If the histogram produces two distinct peaks, it may be possible to separate the foreground from background using simple thresholding. Adaptive thresholding may be used to for thresholding an image whose intensity Histogram does not contain distinct peaks. Intensity Histogram for image produced in Drag and Drop CAPTCH is shown in figures 29 and 30 and images got after applying thresholding are shown in figures 31 and 32.

Figure 29: Image1 Histogram

Figure 30: Image2 Histogram

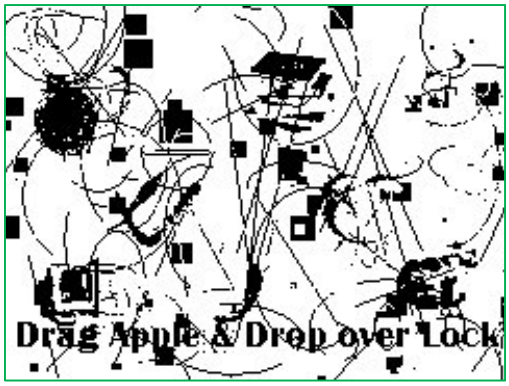


Figure 31: Thresholding on Image 1

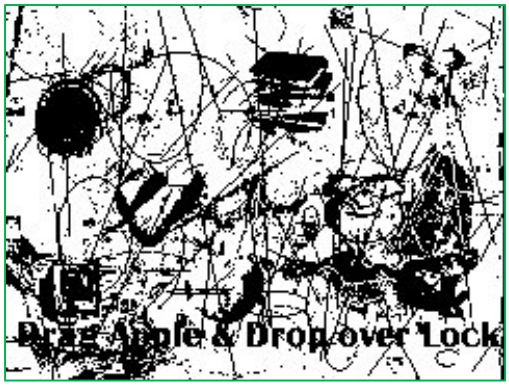


Figure 32: Thresholding on Image 2

As can be seen from the thresholding analysis neither adaptive nor simple thresholding is able to properly segment the image. Removal of some background information from the CAPTCHA image also removes some required information about sub images because of the uniform color distribution.

Shape matching is yet another technique employed in understanding images; however, our CAPTCHA technique increases difficulty in performing this on the produced CAPTCHA image. This is owing to reason that images of interest appear as sub images of different sizes and intensities. Further the source and destination sub image tags are embedded in the CAPTCHA image and sub images are not uniformly spaced which makes shape matching using CBIR methods to fail.

A simple possible attack to the proposed technique involves random guessing wherein an attacker may drag from one portion of the image to the other. In such a case the probability to produce the correct answer can be calculated by the follows formulas:

$$P\big(P(S_{img})and\ P(T_{img})\big) = P(S_{img})\ X\ P(T_{img}\ given\ S_{img})$$

$$P(S_{img}) = \frac{A_{S_{img}}}{A_{C_{img}}}$$

$$P(T_{img} given\ S_{img}) = \frac{A_{S_{img}}}{A_{C_{img}} - A_{S_{img}}}$$

Where P(P(S_{img}) and P(T_{img})) is the probability that source and destination images both have been correctly guessed, P(S_{img}) is the probability that source image has been correctly guessed and P(T_{img} given S_{img}) is the probability that target image is correctly guessed with given source image. $A_{C_{img}}$ and $A_{S_{img}}$ are respectievely the areas of CAPTCHA and sub images.

For our CAPTCHA image, the sub images (S_{img}) have been chosen with dimensions ranging from 45X45 to 55X55 giving an average area of 2500 pixels. The CAPTCHA image has 240X180 pixels dimension thus having an area of 43200 pixels. Using the formulas listed above the following results have been obtained:

$$P(S_{img}) = 0.0578$$

$$P(T_{img} given\ S_{img}) = 0.0614$$

$$P\big(P(S_{img})and\ P(T_{img})\big) = 0.0035$$

Thus the estimated probability that a brute force guessing is successful with the above given dimensions of CAPTCHA and the sub images is only 0.35%.

From the security analysis carried out in this section it is apparent that this CAPTCHA technique is secure from automated tools and can be applied successfully as a Human Interaction Proof.

Conclusion

We, in this paper besides enumerating various existing CAPTCHA methods presented a new visual CAPTCHA technique that not only is simple to generate but is also resistant to attacks from automated Web tools. Further, we carried out security analysis of the proposed technique in terms of segmentation, shape matching and brute force guessing. The results obtained have validated the efficiency of Drag and Drop CAPTCHA algorithm. We are currently working to improve further the usability of the presented technique and apply other image segmentation algorithms to verify its robustness. We are constructing an AJAX based test Web site to gather user statistics and comments for its improvement.

References

[1] M. Rajab, J. Zarfoss, F. Monrose, and A. Terzis. A multifaceted approach to understanding the botnet phenomenon. In Proceedings of the 6th ACM SIGCOMM on Internet Measurement (IMC), pages 41–52, 2006.

[2] H. Baird and K. Popat, Human interactive proofs and document image analysis, Proc. IAPR 2002 Workshop on Document Analysis Systems, August 2002.

[3] M. Blum et al, The captcha project, Department of Computer Sciences, Carnegie-Mellon University, homepage: http://www.captcha.net, accessed 25 August 2008.

[4] G. Ollmann, Stopping automated attack tools, Whitepaper – NGS Software Insight Security Research, http://www.ngssoftware.com/papers/StoppingAutomatedAttackTools.pdf, 2005, accessed 25 August 2008.

[5] M. D. Vivo, G. O. D. Vivo, R. Koeneke, G. Isern, Internet vulnerabilities related to tcp/ip and t/tcp, SIGCOMM Comput. Commun. Rev. 29 (1) 81–85, 1999.

[6] N. J. Hopper, M. Blum, Secure human identification protocols, in: ASIACRYPT, vol. 224 of Lecture Notes in Computer Science, Springer, 2001.

[7] M. Blum, L. A. von Ahn, and J. Langford,, The CAPTCHA Project, "Completely Automatic Public Turing Test to Tell Computers and Humans Apart," www.captcha.net, Dept. of Computer Science, Carnegie-Mellon Univ., November, 2000.

[8] L. von Ahn, M. Blum, and J. Langford, Telling Humans and Computers Apart Automatically, Communications of the ACM, vol. 47, no. 2, pp. 57-60, February 2004.

[9] G. Mori, J. Malik, Recognizing objects in adversarial clutter: Breaking a visual CAPTCHA, in: Proc. Conf. Computer Vision and Pattern Recognition, Madison, USA, 2003.

[10] reCAPTCHA: Stop spam, read books., Dept. of Computer Science, Carnegie Mellon University, http://www.recaptcha.net/, 2007, accessed 25 August 2008.

[11] M. Chew and H.S. Baird "BaffleText", a Human Interaction Proof, proc. 10th SPIE/IS&T Document Recognition and Retrieval Conference (DRR2003), Santa Clara, CA, USA, pp 305-316, 2003.

[12] A. Rusu, V. Govindaraju, Handwritten CAPTCHA: using the difference in the abilities of humans and machines in reading handwritten words, proc. of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR- 9 2004), 2004.

[13] R. Ferzli, R. Bazzi, L. J. Karam, A captcha based on the human visual system masking characterists, ACME, 2006.

[14] PayPal, PayPal Registeration, https://www.paypal.com/, accessed 25, Aug 2008

[15] Microsoft, Microsoft Hotmail, http:/www.hotmail.com/, accessed 25, Aug 2008

[16] M.H. Shirali-Shahreza and M. Shirali-Shahreza, Persian/Arabic Baffletext CAPTCHA, Journal of Universal Computer Science (J.UCS), vol. 12, no. 12, pp. 1783-1796, December 2006.
[17] M.H. Shirali-Shahreza and M. Shirali-Shahreza, Question-Based CAPTCHA, proc. of Int'l conference on computational Intelligence and Multimedia Applications, 2007.
[18] C. Pope and K. Kaur, Is It Human or Computer? Defending E-Commerce with Captchas, IEEE IT Pro, May 2005.
[19] Microsoft, Microsoft asirra, http://research.microsoft.com/asirra/, accessed 25 Aug 2008.
[20] Kittenauth, http://www.thepcspy.com/kittenauth, accessed 25 Aug 2008.
[21] R. Datta, J. Li, J. Z. Wang, Imagination: a robust image-based captcha generation system, in: Proceedings of the 13th annual ACM international conference on Multimedia (MULTIMEDIA '05), ACM Press, New York, NY, USA, 2005.
[22] M. Shirali-Shahreza, and S. Shirali-Shahreza, HIS 2008, Krakow, Poland, May, 2008.
[23] M. Shirali-Shahreza, and S. Shirali-Shahreza, "Drawing CAPTCHA," Proceedings of the 28th International Conference Information Technology Interfaces, Dubrovnik, Croatia, June 19-22, pp. 475-480, 2006.
[24] M. Shirali-Shahreza and S. Shirali-Shahreza, Collage CAPTCHA, in Proceedings of the 20th IEEE International Symposium Signal Processing and Application (ISSPA 2007), Sharjah, United Arab Emirates (UAE), February 2007.
[25] M. Shirali-Shahreza and S. Shirali-Shahreza, Advanced College CAPTCHA, 5th Int'l conference on Information Technology: New Generation, 2008.
[26] M. Shirali-Shahreza and S. Shirali-Shahreza, online college CAPTCHA, 8th Int'l Workshop on Image Analysis for Multimedia Interactive Services(WIAMIS'07), 2007.
[27] M. Shirali-Shahreza and S. Shirali-Shahreza, Multilingual CAPTCHA, 5th IEEE International Conference on Computational Cybernetics ICCC 2007, Oct 2007.
[28] Wen-Hung Liao, A CAPTCHA mechanism based on exchanging image blocks, proc. of the 18th int'l conference on pattern recognition (ICPR'06), 2006.
[29] D. Misra and K. Gaj, Face Recognition CAPTCHAs, Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT/ICIW 2006), 2006.
[30] C. Nancy, Sound oriented captcha, in: Proceedings of the First Workshop on Human Interactive Proofs, Xerox Palo Alto Research Center, CA, 2002.
[31] G. Moy, N. Jones, C. Harkless, R. Potter, Distortion estimation techniques in solving visual captchas, in: CVPR (2), 2004.
[32] K. Chellapilla, P. Y. Simard, Using Machine Learning to Break Visual Human Interaction Proofs (HIPs), MIT Press, Cambridge, MA, pp. 265–272, 2005.
[33] K. Chellapilla, P. Simard, M. Czerwinski, Computers beat humans at single character recognition in reading-based human interaction proofs (hips), in: In Proceedings of the Second Conference on Email and Anti-Spam (CEAS), Palo Alto, CA, 2005.
[34] K. Yanai, M. Shindo, and K. Noshita, A fast image gathering system from the World-Wide Web using a PC cluster, Image and Vision Computing, vol. 22, Issue 1, pp. 59-71 January 2004.
[35] M. Chew, J. D. Tygar, Image recognition CAPTCHAs, in: Proc. of the 7th International Information Security Conference (ISC 2004) Springer, 2004.
[36] GIMP, GNU Image Manipulation Program, http://www.gimp.org/ accessed 25 Aug 2008.

芽|Sprouts

Working Papers on Information Systems | ISSN 1535-6078

Office:
Sprouts
University of Amsterdam
Roetersstraat 11, Room E 2.74
1018 WB Amsterdam, Netherlands
Email: admin@sprouts.aisnet.org

芽|Sprouts