Face Recognition CAPTCHAs

Deapesh Misra and Kris Gaj
Dept. of ECE, George Mason University, USA.
dmisra@gmu.edu, kgaj@gmu.edu

Abstract

Web-based services are becoming more and more ubiquitous and are replacing human-to-human interactions. Automated programs abuse the web services by pretending to be human. As a result, the need to authenticate that the other party on the web is a human and not a program, is on the rise. CAPTCHA is a test that can be used to reliably differentiate between human users and automated programs on the web. In this paper, we propose a new CAPTCHA scheme based on the problem of face recognition. This test takes advantage of the fact that recognizing human faces is considered to be a tough task for computers, but is relatively easy for humans. In the proposed test, human face photographs from a public database are distorted using two different image processing transformations. The user is asked to match distorted photographs of several different human subjects. The automatic generation and evaluation of tests is shown to be possible using the image processing open-source tool Gimp. The proposed CAPTCHAs have the desirable properties of being easy for humans while being difficult for programs to solve. Also the level of comfort in passing these tests is high, independently of the person's familiarity with the English language, when compared to other English text based CAPTCHAs.

1 Introduction

The web is increasingly replacing human to human interactions. Schemes which implicitly assume that the other party on the internet is a human, are susceptible to being deceived by smart programs called 'Web-Bots' which pretend to be humans. Such web bots generally have a malicious intent. Thus, the need to authenticate that the other party on the web is a human and not a malicious program is also on the increase rapidly.

Human Interactive Proofs (HIPs) are schemes which require some kind of interaction from a human user that is tough for a program to simulate. "Completely Automated Public Turing test to tell Computers and Humans Apart" (CAPTCHAs) are a class of HIPs which are tests that are so designed that humans can easily pass them while machines have a very tough time passing them [1]. Thus such HIPs try to prevent malicious programs while allowing humans to access the web services they are trying to secure.

The use of CAPTCHAs in web interfaces, it is hoped will keep such Bots from misusing the web service. Human users will be easily able to authenticate themselves to be human by passing the tests while machines will fail. Some practical examples of web services where CAPTCHAs are required are: online polls, preventing spammers from getting free mail ids, preventing chat bots from irritating people in chat rooms, preventing automated online dictionary attacks in password systems, preventing unruly search engine bots from indexing private web pages, preventing web bots from adding advertisements to comment fields in Blogs etc. As the web replaces human to human physical interaction such examples are bound to increase.

1.1 Introduction of a CAPTCHA

A CAPTCHA has been defined [1] as a program which generates a test which

- Most humans can pass
- Current computer programs can not pass

Additional requirements for a test to be called a CAPTCHA are as follows:

- Test generation code and data should be public
- The test should automatically be generated and graded by a machine

A more technical definition of CAPTCHA is provided in [13] as: "A CAPTCHA is a cryptographic protocol whose underlying hardness assumption is based on an AI problem."

This definition suggests that what could be classified as a CAPTCHA currently, would loose that distinction if a computer program could pass that test sometime in the future with the growth of Artificial Intelligence (AI). The authors

of [13] state that CAPTCHAs have a two way effect. On one hand they keep the malicious programs away and on the other hand they provide motivation to the growth of the field of AI.

Also to be noted is that the definition of the term "hardness" is not precise and is defined in terms of the consensus of a community: an AI problem is said to be hard if the people working on it agree that it's hard.

CAPTCHA tests should be such that an average computer user has no difficulty in passing it, and feels at ease while going through the test.

2 Some Existing Relevant CAPTCHAs

Most of the CAPTCHAs and particularly the ones currently in use are English word based CAPTCHAs [12, 5, 8, 6, 4]. Image based CAPTCHAs are few [1, 11, 3, 7]. Our scheme is an image based CAPTCHA. The scheme is similar to 'Artifacial' [11] the difference being that Artifacial is a face detection problem while ours is a face recognition problem.

The English word based CAPTCHAs irrespective of whether they use or do not use dictionary words, make the assumption that the test taker is familiar with English words. This might not be true for international web service providers, (e.g. Yahoo). These word based CAPTCHAs present a distorted image of a word composed of English letters to the user. The human user is able to apply error correction to the image to decipher the word while the machine is at a loss to know the word. The machine based OCR systems have not advanced so much as to reach the level of the error correction that a human can perform on distorted letters.

As stated earlier, image based CAPTCHAs are few and are not prevalent in use. The human face image based scheme 'Artifacial' [11], makes use of the fact that a human can quickly detect a human face in an image with a highly cluttered background. This problem is the human face detection problem.

'Implicit CAPTCHAs' [3] make use of images in a much more general way. The user is supposed to interact with the picture by clicking on some part of it and thus pass the test. The image in this scheme provides the background for the test, upon which an interaction based task is built.

In the scheme 'Image Recognition CAPTCHAs' [7] the hardness of the problem is provided by the one way transformation between words and pictures. For a machine, it is easy to get pictures corresponding to a particular chosen word, but tough, the other way around. Thus given pictures associated with a word, the human can easily find the word while the machine will fail. This scheme plays around with a few possibilities of this mapping between words and their associated pictures. This is a complicated scheme, the security analysis of which is tough.

In our scheme we move away from making any assumption about the language familiarity of the web service user. We use image based CAPTCHAs to make our tests universal and also to increase the comfort level of the user.

3 Proposed CAPTCHA Test

Our proposed scheme utilizes the fact that humans are better than computers at recognizing human faces. For a machine this task is still very tough [15]. Though a lot of work has been done in the area of face recognition by machines, as of now it is still a hard problem for machines. Moreover there is a good level of understanding as to how hard this problem is. These properties are well exploited to create a CAPTCHA.

The property that we exploit to create our CAPTCHA is that, given two distorted images of a human face, the human user can match these two images as being of the same person quickly, while for a computer program it is very tough to match these two distorted images.

The user is presented with two sets of distorted human face images. Each set has the distorted images of the same group of people. Each set could have four to five images to make random guessing attack less successful. The user is expected to match the same person's faces in these two sets to pass the tests.

The images are chosen from any one of the publicly available face databases. Image processing tools such as the Gimp can be easily automated to create the distortions and apply them to the photographs. The distortions applied to the faces are cleverly chosen so as to be able to defeat the face recognition algorithms.

4 Test Generation Scheme

The generation of the CAPTCHA requires a database of human face images and a way to distort these images all along with the precondition that the process of creation and evaluation can be automated.

4.1 Image Databases

Our scheme makes use of human face photograph databases that are public and have been available for a long time now. There is thus no need for the database to be secret.

Any face database can be chosen. For our experiments we chose the UMIST Face database [14]. Since there is some understanding that image recognition algorithms perhaps are better at recognizing female subjects better [9], we did not use female photographs. The frontal face shots of the people in the database were distorted to create the test.

The databases generally consist of photographs which are taken in constrained environments. Particularly, the lighting, expression and pose being very constrained [10]. The creation of an image database with CAPTCHA like tests in mind (with large variations in pose and lighting) will lead to images which are tougher to break by computer systems. The use of photographs from a database with extreme variations with lighting or facial expressions might even remove the need for distortions to be applied.

4.2 Image Processing Tools

The use of commonly available image processing tools was looked into. Successful results were obtained with the use of the Open Source tool 'Gimp 2.2' [2]. This tool which is available for the Linux environment and also for the Windows environment is particularly suitable for this task since it has a scripting language called 'Script-Fu', which allows automatic creation of new distortions and automatic creation of the CAPTCHAs.

The tool comes with built in image manipulation effects called 'Filters'. These basic built-in filters were used to create the distortion effects. The user can easily generate a large number of new basic filters aka image distortion effects. These image distortion effects can be easily extended to create new effects as and when the attackers are able to successfully attack a distortion scheme that is being currently used.

4.2.1 Distortions Used

For the distortions that we used in tests, there were a few already existing filters in Gimp which satisfied our requirements. These distortions need to be such that given the distorted image it should not be easily possible to recover the original image by applying something akin to an inverse transformation. Thus a few distortions were narrowed down to from the already available set in Gimp 2.2.

From the list of built-in filters that come shipped with the tool, we chose the following filters:

- Glass Tile filter
- Illusion filter under the category 'Map'
- Spread filter under the category 'Noise'

We do not claim that the distortions that were used for this experiment were the best in terms of defeating face recognition algorithms. This in fact is the strength of this scheme, that new better and attack resistant distortions can be easily generated when required.

For human faces, we can't use any random distortion since the output should be acceptable aesthetically. Extreme distortions to the human face would make the CAPTCHA disgusting. While on the other hand, when choosing the parameters for the distortions, we have to ensure that the distorted output is not too simple for an image recognition scheme applied by a machine. Acceptable parameter bounds for the distortions have to be decided by a human being. Once these bounds are set for the various distortions, at run time, random values for the parameters are chosen for the distortion. For the three distortions the parameters we chose were:

1. Glass tile: parameter: 16
2. Illusion: parameter: divisions 3, mode 1
3. Spread: parameter: 15

Since the image database is public, an attacker can access the same database and in spare time apply the distortions to all the images and store such images for comparison later during an attack. To prevent this, the distortion selected must be such that it is sensitive to the randomly run time chosen parameters of the distortion.

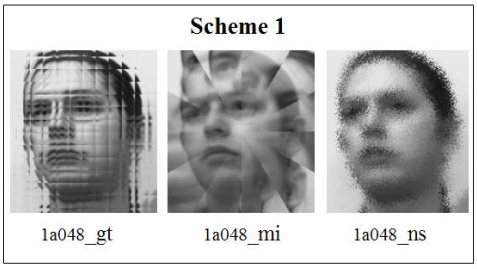

Figure 1. Example 1 for Scheme 1

4.3 Using human faces in image recognition - Scheme One

This CAPTCHA scheme requires the user to recognize the same image of a subject with two different distortions applied to it. Thus in effect, the human user is performing an image recognition task, the image being a human face.

As examples, in Figs.1 and 2, three distortion effects are applied to a randomly chosen human face image from the database.

The names of the images are a variant of the names given in the UMIST database and the name mapping between the distorted images and the applied filters being:

- Glass Tile filter: _gt
- Illusion filter under the category 'Map': _mi
- Spread filter under the category 'Noise': _ns

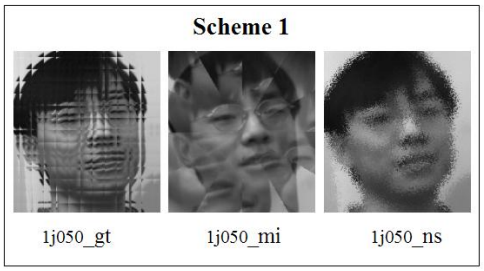

Figure 2. Example 2 for Scheme 1

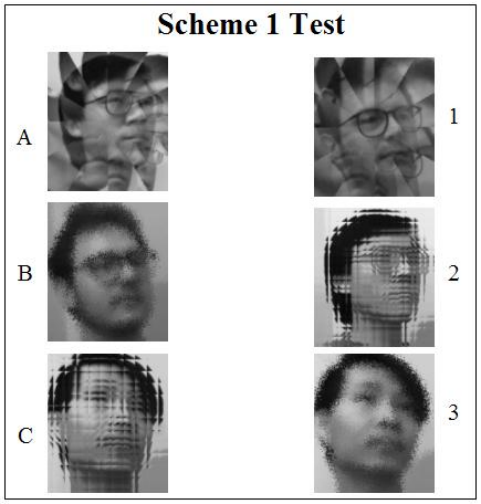

Figure 3. Example CAPTCHA test for Scheme 1

To make the scheme more robust, instead of having a constant set of two distortions, a set of many distortions are created. Randomly at run time, two distortions from this set are chosen and applied. These are applied to the two copies of the randomly chosen subject's image. A set of such pairs are created. The CAPTCHA test could display around five images to the user and ask the user to match the pairs. This is as illustrated in Fig. 3 wherein the number of subject images are three.

The two distortions can be so chosen that one distortion makes it tough for holistic feature matching face recognition schemes while the other makes it tough for feature matching face recognition schemes.

4.4 Recognizing human faces - Scheme Two

An extension to our basic idea is to use different photos of the same individual to which different distortions are applied respectively.Two different images, perhaps of the subject in two different poses are taken.

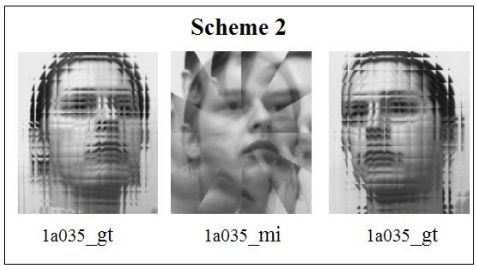

Figure 4. Example 1 for Scheme 2

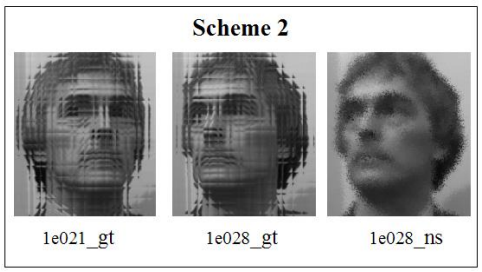

Figure 5. Example 2 for Scheme 2

In this scheme the distortions applied need not distort the image as much as the previous scheme. The human user has to recognize the subject given the two distorted different views of the subject. This is a true human face recognition scheme. The test combines the ideas that humans can recognize human faces better than computers and do it especially much better than machines, when the images are distorted. In Fig. 4 and Fig. 5 an example having three sets of images are shown. The photos of the different poses of the same subject are generally available in face recognition related image databases. To create the CAPTCHA, a subject is randomly chosen from the database and to different poses of this subject, randomly chosen distortions from the set of distortions are applied.

5 Discussion of our new scheme

Our new human face recognition scheme makes use of an area that is well researched and understood. Human face detection and recognition are still hard problems for machines to solve and this is made even harder by the application of distortions to the images. The distortions also serve to break the existing face recognition schemes. Easy extensibility of these distortions due to the use of the tool 'Gimp' ensures that as the face recognition schemes get better, newer distortions can be easily created, thus keeping this idea in vogue for a long time. The script that automatically generates the CAPTCHAs stores the answer for evaluation purposes.

A static constant database of human images with the use

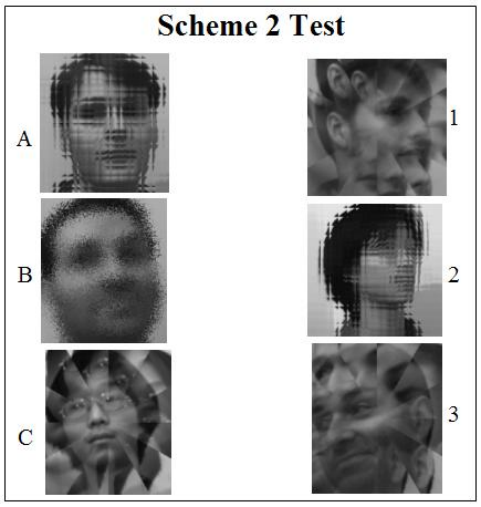

Figure 6. Example CAPTCHA test for Scheme 2

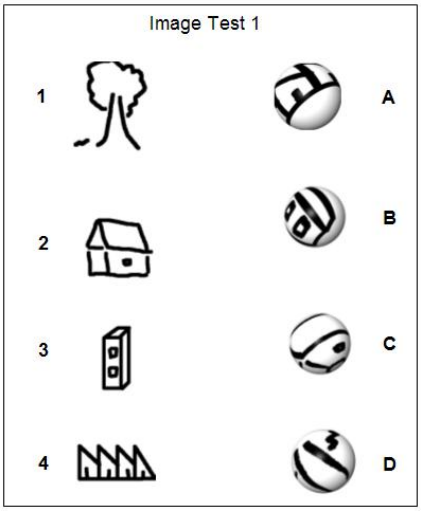

Figure 7. Example CAPTCHA test 1

of a changing set of new distortions can ensure the strength of this scheme for a long time.

The obvious disadvantage in such a 'multiple choice test' is that it is susceptible to guessing attacks. Word based CAPTCHAs have a much higher probable answer space, but at the same time are much more inconvenient for international users. Our scheme attempts to break away from the language barrier and also attempts to create tests that are user friendly rather than strict policy enforcers. With the ease of the user in mind we attempt to move towards the idea of "Human Friendly Human Interactive Proofs".

The most easiest attack would be to randomly guess the answers. The easiest mitigating measures that can be used are:

1. After every wrong attempt a new test is created
2. The identifying label for each image itself can be a text based CAPTCHA !

One easy way to increase the answer space is to have images in the second set be the answer for multiple images in the first set. This is possible if in the first set, there are more than two images of the same person. This removal of the constraint of being a one-to-one mapping between the first and second sets, increases the answer space.

We do not suggest the use of many rounds of tests as in [7]. Each CAPTCHA consists of just one round. In case the user fails the round then the user can try a different CAPTCHA.

Generating distortions which leave a distorted human face image still aesthetically acceptable is a task that needs to be done for the benefit of both the test takers and the people whose photos are used in the tests.

6 Extension

Face recognition is not a new science. Thus though we understand its current limitations and exploit them to create CAPTCHAs, there has been and will be progress in this area. To make the CAPTCHA tougher against human face recognition programs, this scheme could be extended to distortions of general images rather than only human face images. The advantage being that it is tougher to recognize general random images in comparison to recognizing human faces, since all human faces share some common features. The database can again be public in this case. The decision of what images would go into the database has to be made by a human. In this case the distortions can be from a very large set as there are no aesthetic consideration requirements. Thus, at two levels the scheme has larger independence i.e., the choice of images and the choice of distortions.

A few examples using the built in filters in Gimp, are depicted in the figures. In Fig. 7 a distortion is used to map the images onto surfaces. The user is asked to match these images. In Fig. 8, a 'Fractal' is created from the image. Many such distortions are possible. Again, a clever choice of distortions will ensure that the recognition schemes fail.

7 Conclusions

Web services have to ward off malicious programs from abusing their services. This growth of abuse of services is due to the fact that most protocols have not considered that the user might not be a human. Thus to ensure that it is always a human on the other end, CAPTCHA like tests will only increase their presence on the web. Since such tests

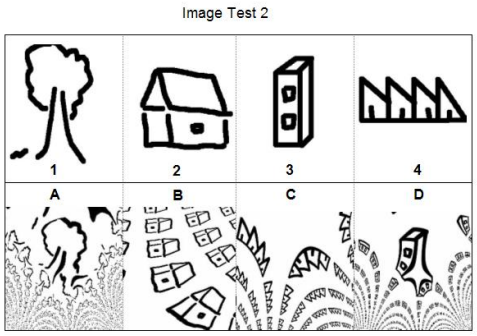

Figure 8. Example CAPTCHA test 2

will increase the discomfort of users using the web services, it is important to design user friendly CAPTCHAs. Our 'Human friendly Human Interactive Proofs' is an attempt in that direction.

The development of image distortion effects specifically to defeat human face recognition schemes, for instance Fischerfaces and Eigenfaces would be the way ahead. As new schemes are developed to recognize human faces, new image distortion effects will have to be developed. Also what needs to be looked into is a way to prevent guessing attacks. For non-human face based CAPTCHA schemes, a further investigation into recognition of general images by machines needs to be conducted. And the design of such a database has to be investigated further.

The new scheme that is proposed caters to all of the requirements of a CAPTCHA. It is also extensible and its defenses can be easily hardened as and when required. Thus, we not only return back to being compliant with the requirements of being a CAPTCHA as enumerated originally [1], we also ensure that this scheme can be easily extended as artificial intelligent techniques to recognize human faces get better in future. A shift towards human friendly designs is attempted by the use of image based CAPTCHAs.

References

[1] The captcha website. http://www.captcha.net.

[2] Gimp 2.2. http://www.gimp.org/.

[3] H. S. Baird and J. L. Bentley. Implicit captchas. In *Proceedings of SPIE/IS&T Conference on Document Recognition and Retrieval XII*, 2005.

[4] H. S. Baird and T. P. Riopka. ScatterType: a reading CAPTCHA resistant to segmentation attack. In *Vision Geometry XIII. Edited by Latecki, Longin J.; Mount, David M.; Wu, Angela Y. Proceedings of the SPIE, Volume 5676, pp. 197-207 (2004).*, pages 197–207, Dec. 2004.

[5] K. Chellapilla, K. Larson, P. Simard, and M. Czerwinski. Designing human friendly human interaction proofs (hips). In *CHI '05: Proceeding of the SIGCHI conference on Human factors in computing systems*, pages 711–720, New York, NY, USA, 2005. ACM Press.

[6] M. Chew and H. S. Baird. Baffletext: a human interactive proof. In *Proceddings of the 10th IS&T/SPIE Document Recognition & Retrieval Conference*, 2003.

[7] M. Chew and J. D. Tygar. Image recognition captchas. In *ISC*, pages 268–279, 2004.

[8] A. L. Coates. Pessimal print: A reverse turing test. In *ICDAR '01: Proceedings of the Sixth International Conference on Document Analysis and Recognition (ICDAR '01)*, page 1154, Washington, DC, USA, 2001. IEEE Computer Society.

[9] R. Gross, J. Shi, and J. Cohn. Quo vadis face recognition? In *Third Workshop on Empirical Evaluation Methods in Computer Vision*, December 2001.

[10] J. Howell. Introduction to face recognition. In L. Jain, H. U., H. I., L. S.B., and T. S, editors, *Intelligent biometric techniques in fingerprint and face recognition*. CRC Press, Inc., Boca Raton, FL, USA, 1999.

[11] Y. Rui and Z. Liu. Artifacial: automated reverse turing test using facial features. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, pages 295–298, New York, NY, USA, 2003. ACM Press.

[12] P. Y. Simard, R. Szeliski, J. Benaloh, J. Couvreur, and I. Calinov. Using character recognition and segmentation to tell computer from humans. In *ICDAR '03: Proceedings of the Seventh International Conference on Document Analysis and Recognition*, page 418, Washington, DC, USA, 2003. IEEE Computer Society.

[13] L. von Ahn, M. Blum, N. Hopper, and J. Langford. Captcha: Using hard ai problems for security. In *Proceedings of Eurocrypt*, pages 294–311, 2003.

[14] H. Wechsler, P. J. Phillips, V. Bruce, F. Fogelman-Soulie, and T. S. Huang, editors. *Characterizing Virtual Eigensignatures for General Purpose Face Recognition*, volume 163, 1998.

[15] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4):399–458, 2003.