**Chapter 1**

**Introduction**

Today, the computer and the Internet constitute the cornerstone [15] of any modern society supporting the daily human activities and enabling new intelligent methods of fast and easy communication. The growing role of the Internet as a communication medium is highlighted, beyond the obvious living experience, by the high volume of research on practical directions like the development of tools and algorithms that can handle processes and messages generated from a wide number of heterogeneous platforms and technologies, as well as on the more theoretical ones which are ranging from concepts and designs that aim to increase the familiarity and usability of the multi-modal interfaces, to the investigation of the effects of the digital divide in specific societal contexts.

Similarly, the World Wide Web (WWW or Web) has become a platform for deployment of complex applications of increased interactivity, as it takes the form of a medium used for multifaceted and important tasks like commercial and governmental transactions, social intercommunication, collaborative work, learning and information retrieval. Within this realm, adaptation and personalization of interactive systems is considered a promising research direction as it is sufficient that the consequences of a system that will not offer the expected user experience, hinder the interactions of users, or create false expectations and not effective use, will generate frustration and decrease its acceptability. In this context, one of the most important and challenging issues is to support users, engaged on tasks related to their everyday processes, through adaptive and personalized human-computer interface designs and applications.

Adaptation and personalization, which could be considered as a central practice of Human-computer Interaction (HCI), is therefore pronounced as the main ingredient of future online services and applications which are expected to offer a rich set of computing and communication services to users in a broader context, representing unprecedented opportunities to access, manipulate, and share information as well as to accomplish tasks through heterogeneous devices and contexts of use. In this regards, adapting the functionality and the content, of an interactive system, into an assemblage that specific users are able to understand and use intuitively in order to perform specific tasks is a challenging endeavor. It entails understanding and modeling of human behavior for diverse user groups and stakeholders, with regard to structural and functional user requirements. These, in turn, need to be translated into usable human-computer interaction designs and workflows that aim to minimize user cognitive loads, perceptual and learning efforts and erroneous interactions.

Taken into consideration that users of the World Wide Web do not necessarily share common conventions, cultural, and cognitive backgrounds and contexts in which specific decisions are required to be taken we propose in this book that adaptive and personalized systems and user interfaces could provide a viable alternative in order to ensure simplicity and ease of use offering equal chances for participation by all. Adaptive user interfaces provide an alternative to the “one-size-fits-all” approach of static user interfaces by adapting the interactive system’s structure, navigation, terminology, functionalities and presentation of content to users’ perceptions and level of knowledge with regard to any type of information or service, aiming to increase the usability of the interface and provide a positive user experience.

This studies the important theoretical dimensions and technologies that govern the relationship between human-computer communication, from the adaptation and personalization angle, suggesting methods and techniques that can promote the robust interaction among the two entities in various application domains. Throughout the report, we have the standpoint that the application of adaptation and personalization should always have the human (user) in the center. This expands the focus of current approaches that might consider traditional user characteristics (as for example needs and/or requirements), and gives emphasis on how intrinsic human individual characteristics could be explicitly considered with respect to users’ interactions with any kind of interface. Henceforth, we have coined the term ‘Human-centered Adaptation and Personalization’ since we tackle the subject from a more holistic and interdisciplinary perspective bringing human factors in the centrum and discussing methods for creating inclusive user models, intelligent algorithms and techniques for specialized use cases that increase user experience and usability.

In addition, our main concern, amongst others, is to present in the best possible way the co-existence of the two worlds, of theory and practice, in the adaptation and personalization area, and how researchers and practitioners can benefit depending on their area of application. Therefore, we move beyond standalone and isolated attempts, investigating methods and techniques of how a high level understanding (or the extraction) of specific human factors can be put into practice and provide the desired adaptive and personalized outcome. This could be achieved by (a) defining accurate human-centered models based on users’ perceptions, behaviors, abilities, experiences, etc.; and (b) creating adaptation and personalization algorithms, interaction

principles and smart interfaces that can handle the increasing complexity of data structures and the high volume of information.

In chapter 2 we have described the CAPTCHA.

In chapter 3 we have described the MAPTCHA.

In chapter 4 we have discussed some well-known research papers dealing with MAPTCHA and have compared them.

In chapter 5 we have described our proposed methodology in which the algorithms used are described.

In chapter 6 we have described the implementation of our proposed algorithm.

In chapter 7 we have discussed the results that we have observed after implementing our algorithm and analyzed the results.

In chapter 8 we have described the future work that we will do to improve our proposed MAPTCHA algorithm.

In chapter 9 we have described the conclusion.

In chapter 10 we have described the references that we have used throughout this report.

**Chapter 2**

**CAPTCHA**

CAPTCHA (Completely Automatic Public Turing Test to Tell Computer and Human Apart) methods are one of the important branches of HIP (Human Interactive Proof) systems which are used to distinguish between human users and computer programs automatically. A good CAPTCHA should satisfy two main requirements: robustness and usability. The robustness aspect is its strength to defend against adversarial attacks; the usability aspect is the ease with which humans pass its challenges. The advancement in Information Technology (IT) services implied that many services are offered on the Internet for the users. These services may range from banking to blogging and from e-mails to e-commerce. These services and their associated resources are largely offered free of cost (e.g.: email) or for a nominal fee (e.g.: banking services). But, unfortunately, malicious users can gain access to these resources by various means thereby denying these resources to genuine users. Malicious users such as hackers have developed automated programs (called bots) that act as simulated human beings to gain access to a variety of services. For example, bots are being used to create e-mail accounts, send spam messages, participate in online surveys and hence skew the results. The pipeline of attacking CAPTHCHA by computers normally consists of three parts: preprocessing, segmenting the crowding characters, recognizing signal characters.

**2.1 What is CAPTCHA?**

In artificial intelligence (AI), a test known as the Turing test is proposed for proving the intelligence of a computer. In this test, a human person and a computer are put in two different rooms and a human interrogator in a third room asks them questions. If the interrogator cannot recognize which room the computer is in and which one the human, it is said that the computer has passed the Turing test. A similar method to the Turing test can be used to distinguish human users from computer programs with the difference that the human interrogator is replaced with a computer. The computer interrogator asks questions from the applicant to distinguish between the human user and the computer program. These methods are known as CAPTCHA (Completely Automated Public Turing test to tell Computers and Human Apart).

**2.2 Why do we use CAPTCHA?**

The technology is used mostly to block spammers and bots that try to automatically harvest email addresses or try to automatically sign up for or make use of Web sites, blogs or forums. CAPTCHA, whose users include Yahoo and Google, blocks automated systems, which can't read the distorted letters in the graphic.

**2.3 Brief history of CAPTCHAs**

Moni Naor (Naor, 1996) is the first person that mentioned [8] some theoretical methods for telling apart computers from humans remotely. He did it looking for ways to prevent the abuse of web services. The first known use of a CAPTCHA test was done in the Alta-Vista web-search engine. Early in 1997 they faced the problem of the automatic submission of URLs to their search engine. This free ‘‘add-URL’’ service broadens its search coverage, but some users were abusing it by automating the submission of large numbers of URLS, in an effort to manipulate AltaVista’s importance ranking algorithms (that same kind of attacks have been known against other search engines, like Google). Andrei Broder at AltaVista and his colleagues developed a filter. Their method is to generate an image of printed text randomly so that machine Optical Character Recognition systems (OCR) cannot read it but humans still can. Five years later, Broder claimed that the system had been in use for ‘‘over a year’’ and had reduced the number of ‘‘spam add-URL’’ by ‘‘over 95%’’. A U.S. patent was issued in April 2001 (US patent no).In 2000, Udi Manber of Yahoo! described their ‘‘chat room problem’’ to researchers at Carnegie-Mellon University: ‘bots’ (automatic computer scripts – a kind of program) were joining on-line chat rooms and pointing the chat-room users to advertising sites. They wanted to find a way to prevent ‘bots’ from joining the chat rooms. Professors Manual Blum, Luis A. von Ahn, and John Langford, from Carnegie-Mellon University, described some desirable properties of such a test:

1. The test’s challenges can be automatically generated and graded, that is, a computer program should be able to generate the challenges and mark the answers as qualifying or not.

2. The test can be taken quickly and easily by human users.

3. The test will accept virtually all human users with high reliability.

4. Low false negatives (‘‘insults’’): the test will reject very few human users.

5. Low false positives (‘‘frauds’’): the test will reject virtually all machine users.

6. The test should resist automatic attack for many years even as technology advances.

The Carnegie-Mellon University team developed a ‘hard’ GIMPY CAPTCHA which picked English words at random and rendered them as images of printed text under a wide variety of shape deformations and image distortions, including the images of different words overlapping. The user was asked to transcribe some minimum number of those words correctly. A simplified version of GIMPY, using only one word-image at a time, was installed by Yahoo!. The term CAPTCHA (for Completely Automated Turing Test to Tell Computers and Humans Apart) was coined in 2000 by this CMU team (Ahn et al., 2003).

**2.4 Relation with AI**

While used mostly for security reasons, CAPTCHAs [21] also serve as a benchmark task for artificial intelligence technologies. According to an article by Ahn, Blum and Langford, “Any program that passes the tests generated by a CAPTCHA can be used to solve a hard unsolved AI problem.”

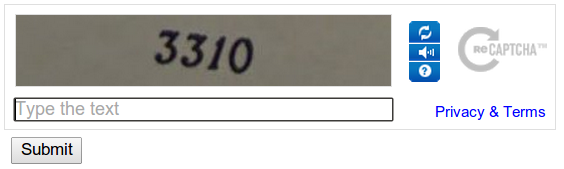
They argue that the advantages of using hard AI problems as a means for security are twofold. Either the problem goes unsolved and there remains a reliable method for distinguishing humans from computers, or the problem is solved and a difficult AI problem is resolved along with it. In the case of image and text based CAPTCHAs, if an AI were capable of accurately completing the task without exploiting flaws in a particular CAPTCHA design, then it would have solved the problem of developing an AI that is capable of complex object recognition in scenes.

**2.5 Different types of CAPTCHA**

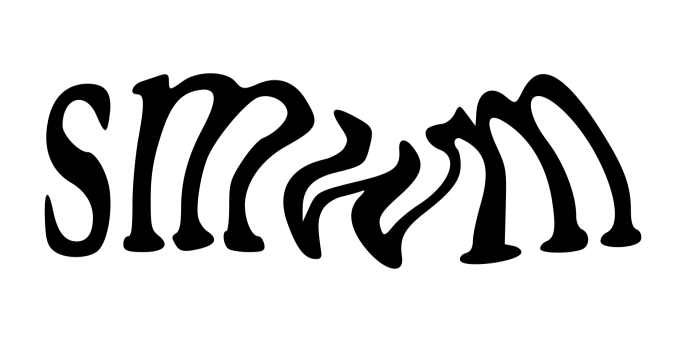
CAPTCHA has been developed with images and sounds since text-based CAPTCHA is no longer safe. However, any kinds of them have problems or difficulties. Several kinds of methods for CAPTCHA are introduced.

A.Text-based CAPTCHA

The most popular type of CAPTCHA is text-based CAPTCHA. An example of text-based CAPTCHA in Yahoo is shown in Fig 1 (a). This kind of CAPTCHA can be easily analyzed with character recognition algorithms such as those [5][9] used in OCR (Optical Character Reader). As another example, e.g.-gimpy which is a simplified version of gimpy is shown in Fig 1 (b). The CAPTCHA is also analyzable with the rate of 92 percent as is mentioned in a paper by Mori et al. Moreover, CAPTCHA analysis projects such as PWNtcha and development of OCR technology have pushed up the performance of analysis of CAPTCHA. Therefore, the readability of a text-based CAPTCHA has decreased as a countermeasure against that analysis. However, the decrease of the readability has a problem since extremely transformed characters such as shown in Fig 1 (c) are hard to read for not only bots (abbreviation for robots) but also human being.



(a)

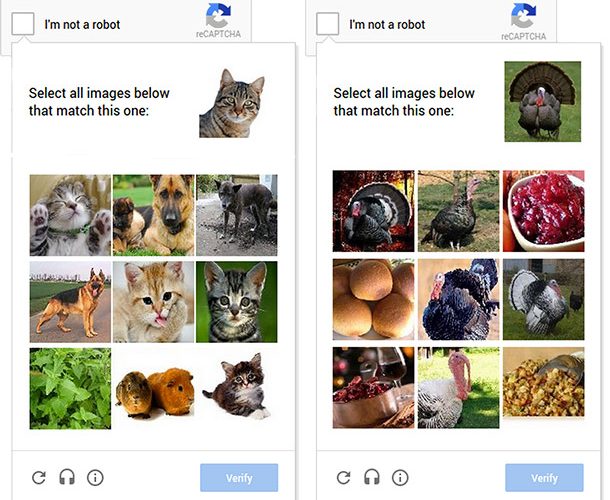


(b) (c)

Fig 1(a), (b), (c): Text Based CAPTCHA

B.Image CAPTCHA

Image CAPTCHA was proposed as a solution of the problem of text-based CAPTCHA. Images which are well matched with a theme are selected in this kind of CAPTCHA. That is, contents in the images must be recognized. As an example, reCAPTCHA by Google is shown in Fig 2. In reCAPTCHA, only suitable images [19] for a theme must be chosen among images on a screen. It seems difficult for bots to read reCAPTCHA since complex algorithms are required to recognize contents in images. However, image CAPTCHA can be attackable by using database. A provider of image CAPTCHA has to classify images manually since the images must be difficult to be recognized by computers in order to prevent bots from recognizing contents of the images. That is, the number of images which can be stored in a database depends on human resources and not unlimited. The fact indicated that it is also possible for attackers to construct the same database by collecting images from the original database.

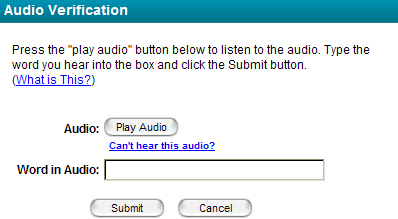


(a) (b)

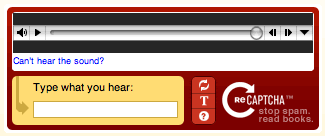
Fig 2 (a), (b): Image CAPTCHA

C.Audio CAPTCHA

Audio CAPTCHA is also a kind of CAPTCHA methods although no vision such as texts or images is required. Human beings and bots [19] are distinguished with their ability of listening by audio CAPTCHA. However, pronunciation of characters or words is the same in all questions so that it can be easily matched with enrolled patterns which can be collected from the same CAPTCHA. If the volume of noise enlarged or plural voices are wrapped in order to prevent bots from analyzing the CAPTCHA, the usability of human beings decreases as same as it decreases in text-based CAPTCHA.



(a)



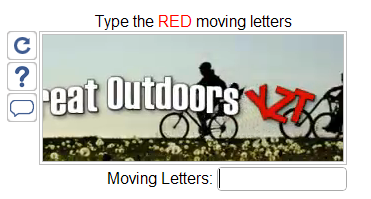
(b)

Fig 3 (a), (b): Audio CAPTCHA

D.Video CAPTCHA

Video CAPTCHA is one of subspecies of image CAPTCHA. Plural images are sequentially displayed in this CAPTCHA. The plural images [19] make machinery analysis more difficult. An example of this CAPTCHA is NuCAPTCHA by Leap Marketing Technologies which is a Canadian software company. In NuCAPTCHA, alphabetical or numerical characters are randomly generated with plural fonts. Human beings can easily recognize characters when each character moves to different direction since the characters are grouped together in the motion. Furthermore, in NuCAPTCHA, difficulty of recognition changes according to the prediction whether an examinee is a correct user or an attacker so that both security and usability are kept in high level. However, NuCAPTCHA is also possible to be analyzed since a video can be separated into images which are readable with OCR technology. **

(a)

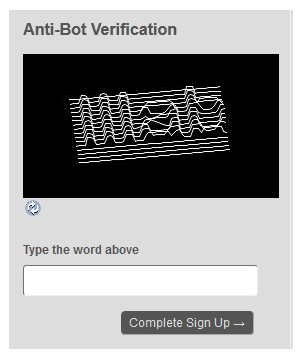
**

(b)

Fig 4 (a), (b): Video CAPTCHA

E.3D CAPTCHA

CAPTCHAs adopting segmentation-resistant mechanism enhance the robustness, but the overlapping of neighboring characters makes [14] human users hard to recognize these characters. To overcome the limitation, 3D text CAPTCHAs have been proposed. The assumption is that human can recognize 3D images of text characters better than computer vision system presents characters on individual faces of a 3D cube, and build a text-based CAPTCHA from stereoscopic 3D images. To create 3D visual effect, CAPTCHA mentioned in appears on a grid in 3D space, and CAPTCHA in uses shadows. Some have even proposed that the 3D text-based CAPTCHA is the next generation in CAPTCHA design. The research on the generation of 3D CAPTCHAs is less than that of 2D CAPTCHAs, and the research on their breaking and security is much less. Research is the only one describing a method of breaking a 3D-based CAPTCHA to our knowledge and the target CAPTCHA is implemented by rediff.com (see Fig. 5(a), (b) for examples).



(a)

******

(b)

Fig 5 (a), (b): 3D CAPTCHA

Also there are several other types of CAPTCHA s which include Puzzle CAPTCHA and also sliding CAPTCHA.

**2.6CAPTCHA with Amodal Completion**

CAPTCHA method with amodal completion is proposed by Mori et al [19]. This CAPTCHA is a kind of video CAPTCHA methods but analysis by OCR is prevented by amodal completion which is one of the completion ability of vision for human beings. Of course, amodal completion is also emulated by computers with neural network technology as is mentioned Fukushima et al. Amodal completion is used not for preventing bots from recognition but for helping human beings from recognition in the method by Mori et al. In their method, human beings can recognize characters at four moments within ten seconds in which three hundred images are included. That is, bots must analyze three hundred images to find the four images in which characters can be cognized with amodal completion. The cost for analysis of one image is not so high, but the cost for three hundred images cannot be ignored for bots. On the other hand, human beings can easily recognize a character at each moment with amodal completion. Therefore, their CAPTCHA can be easily recognized [19] by human beings while it cannot be recognized bots which are working hacked someone’s computers since usual computers are not powerful enough to calculate their CAPTCHA. However, their CAPTCHA has a problem. Cognition of characters is easy for human beings when they can choose a correct image in the video. On the other hand, the choice requires quite high concentration. That is, for human beings, recognition is easy but segmentation is difficult in their method. Furthermore, in the research by Sawada et al.

**2.7Important properties of a CAPTCHA system**

The four important properties that every CAPTCHA system must possess:

**1. Secure:** The program generated tests must be difficult for the machines to solve using any algorithms.

**2. Automated:** The programs (computer programs) must be capable of generating and grading the test.

**3. Open:** In accordance with the Kerckhoff’s principle, the underlying Algorithms and databases must be made public.

**4. Usable:** Humans as users should be able to solve these tests in a reasonable time.

**2.8****Attacks on CAPTCHA**

**2.8.1 A SEGMENTATION ATTACK:**

We have developed a low-cost attack that can effectively and efficiently segment challenges generated by the MSN scheme [12]. Specifically, our attack achieves the following:

• Identify and remove random arcs

• Identify all character locations in the right order; in other words, divide each challenge into 8 ordered segments, each containing a single character.

This attack involves 6 consecutive steps, each of which is detailed in the following sections:

**Pre-processing:** We first convert a rich-color challenge [12] [14] to a black-white image using a threshold method: pixels with intensity higher than a threshold value are converted to white, and those with a lower intensity to black.

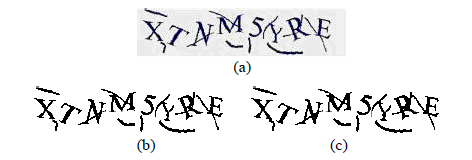
******

Fig 6. Pre-processing. (a) original image, (b) binarized image, (c) after fixing broken characters.

**Vertical Segmentation:** A vertical segmentation method is applied to segment a challenge vertically into several chunks, each of which might contain one or more characters. The process of vertical segmentation starts by mapping the image to a histogram that represents the number of foreground pixels per column in the image. Then, vertical segmentation lines separate the image into chunks by cutting through columns that have no foreground pixels at all.

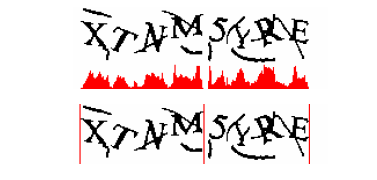
******

Fig 7 : Vertical Segmentation

**Color filling segmentation:** In this step, a “color filling segmentation (CFS)” [12] [13] algorithm is applied to eachchunk segmented in the previous step. The basic idea of this algorithm is to detect every connected component, which we call an objec*t*, in a chunk. An object can be an arc, character, connected arcs, or connected characters. The algorithm works as follows. First, detect a foreground pixel, and then trace all its foreground neighbors until all pixels in this connected component are traversed – that is, an object is detected. Next, the algorithm locates a foreground pixel outside of the area of the detected object(s), and starts another traversal process to identify a next object. This process continues until all objects in the chunk are located.

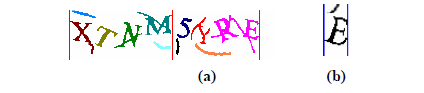


Fig 8 (a),(b): Color filling segmentation

**Thick arc removal:** Thick arcs, if any, will be detected and removed after the above color filling process. Thick arcs have the following characteristics, which make it possible to identify and remove them automatically.

• **Pixel count**. Often, a thick arc has a relatively small pixel count (i.e., the number of foreground pixels in the arc).

• **Location**. Thick arcs are located close to or even intersect with the image border, something which rarely occurs with valid characters unless they are connected to the thick arc.

• **Shape**. Thick arcs do not contain circles. Characters such as A, B, D, P, Q, 4, 6, 8 and 9 all contain one or more circles.

• **Interplay between shape and location**. The position of thick arcs and their geometric shapes are somehow correlated. Thick arcs can be removed using some algorithms they are:

**Circle detection,** which detects if an object contains a circle. If an object contains a circle, we know it is definitely not an arc, and all other arc removal methods can be skipped.

**Scan all objects that contain no circles for discriminative features**(other objects are safely ignored), Such discrimination is largely about pixel count checking. If an object has a pixel count smaller than or equal to 50, it is removed as an arc.

**Relative position checking***.* This step examines the relative position of objects in a chunk, and is applied to all chunks that contain more than one object (note that connected characters are considered as a single object). The basic idea behind this step is that the relative positions of objects can tell arcs and real characters apart.

**Detection of remaining arcs**. The above steps do not necessarily identify all the arcs in an image. What is done in this step is as follows. First, count the number of remaining objects in the image (identified arcs are already removed and thus not counted). If this number is larger than 8, then there is at least one undetected arc in the image.

**2.8.2Dictionary attack: -**

There are a lot of tools that can automate the login process by submitting authentication credentials to a web application. With a bit of effort, anyone can create such a script. How does this affect you and the security of your web applications? In the hands of a malicious person, this can be a dangerous weapon. With a list of user-ids and passwords, this person can try to guess a valid user’s credentials and login. Once logged in, the application will assume that this person is the valid user that the credentials belong to and will allow access to resources (links with the application, documents, reports, etc.).

**2.9Advantages and Disadvantages of CAPTCHA**

**2.9.1 Advantages**

The advantages of CAPTCHA are as follows:

1)Distinguishes between a human and a machine.

2)Make Online polls more legitimate.

3)Reduces spam and viruses.

4)Makes Online shopping safer.

5)Diminishes abuse of free email account services.

**2.9.2 Disadvantages**

The disadvantages of CAPTCHA are as follows:

1)Sometimes very difficult to read.

2)Are not compatiable with users with disabilities.

3)Time-consuming to decipher.

4)Technical difficulties with certain internet browsers.

5)May greatly enhance Artificial Intelligence.

**2.10 The Usable Security Case**

Nowadays, a high number of security mechanisms exist in which users play an important role in the security process. For example, users setup and make use of passwords for accessing a system, while they have [15] to comply with security

certificates and share information that highly affect the security of a system. Furthermore, security mechanisms are deployed on the World Wide Web in which many users are required to interact as secondary tasks in order to proceed with their primary one. For example, they are required to solve a CAPTCHA challenge to prove that they are humans in order to proceed with their main task. Such practices have shown to significantly decrease the experience of users while interacting with the system. The two areas that receive significant attention from the research community (and that we also focus in this chapter) are those of user authentication and CAPTCHA challenges. Both security mechanisms are currently widely deployed in online services and are of critical importance for the security of today’s interactive systems. User authentication on the one hand aims to verify that the identity of a user is genuine, whereas CAPTCHA challenges [15] aim to prove that the entity interacting with a service is human and not malicious software. These tasks are currently performed by millions of users as part of their daily activities, thus having a usability flaw in such human-computer interaction cycles could eventually decrease the overall user experience and user acceptance of an interactive system.

**2.10.1 User Authentication**

User authentication is the process of verifying the physical identity of a person and is a vital component of any security infrastructure of today’s interactive systems. During an authentication task, users are required to provide specific secret information in order to prove their identity. Depending on the factor used for authentication, researchers and practitioners promote different mechanisms; knowledge-based authentication mechanisms[15]that require from users to either memorize and provide a sequence of characters (e.g., password, personal identify- cation number (PIN)) or a sequence of images; token-based authentication mechanismsthat require a specific object from users such as a credit card; or biometric-based authentication mechanisms that require biometric information from users such as fingerprint information. In this chapter we primarily focus on knowledge-based authentication mechanisms (that principally require users’ cognitive processing) with the aim to assist users during such cognitive tasks by providing personalized user authentication tasks bootstrapped on their preferred cognitive processing styles and abilities.

**2.10.2 Human Interaction Proofs (CAPTCHA)**

Human Interaction Proofs (HIP) are security defense mechanisms aiming to prove that the entity interacting with a system is a human being and not malicious software. A Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA – von Ahn et al. 2004) is an example HIP mechanism which is widely used today by service providers to protect their systems against automated software attacks (e.g., denial of service attacks, [15] password dictionary attacks, etc.). CAPTCHA challenges typically require from legitimate users to solve visual cognitive-based challenges before performing the primary task of interaction in a system. For example, systems require from users to recognize distorted alphanumeric characters or solve image puzzle problems before commenting on a Web-site. Main aim of this process is to prevent a possible automated software attack that could automatically generate and send thousands of comments in the system that would decrease the quality of services. These challenges are based on the assumption that they can be easily solved by humans but present significant difficulty for computing systems (e.g., Optical Character Recognition (OCR) or other image-recognition systems).

**2.11 Some commonly used CAPTCHAs**

**Table 1: Some commonly used CAPTCHAs**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Web-Site | Subject | Challenge | Challenge | Challenge |
| MySpace | Social | C:\Users\SAIKAT-\Downloads\Figure-2-Example-of-an-inaccessible-sign-up-Captcha-from-Myspacecom-Other-major.png | C:\Users\SAIKAT-\Downloads\myspace-captcha.png | C:\Users\SAIKAT-\Downloads\myspace-captcha2111.png |
| Facebook | Social | C:\Users\SAIKAT-\Downloads\8nRnl.jpg | C:\Users\SAIKAT-\Downloads\http_%2F%2Fmashable.com%2Fwp-content%2Fuploads%2F2009%2F09%2Ffacebook_captcha_example_2.jpg | C:\Users\SAIKAT-\Downloads\fbcaptcha-1.jpg |
| Microsoft live/Passport | email social | C:\Users\SAIKAT-\Downloads\microsoft-account-recover-password-access-2.png | C:\Users\SAIKAT-\Downloads\passport_captcha.png | C:\Users\SAIKAT-\Downloads\MSN+Passport+service+CAPTCHAs_.jpg |
| eBay | Thrift | C:\Users\SAIKAT-\Downloads\captcha-lead.png | C:\Users\SAIKAT-\Downloads\anti-spam-secure-form-example-03.gif | C:\Users\SAIKAT-\Downloads\decaptcha-7-638.jpg |
| megaupload | File exchange | C:\Users\SAIKAT-\Downloads\mic20120200445 (1)1.gif | C:\Users\SAIKAT-\Downloads\mic20120200445 (1)2.gif | C:\Users\SAIKAT-\Downloads\mic20120200445.gif |
| AOL | email  ISP | C:\Users\SAIKAT-\Downloads\AOL___Sign_Up_cropped.png | C:\Users\SAIKAT-\Downloads\download.jpg | C:\Users\SAIKAT-\Downloads\Captcha-protection-example-in-AOL.png |
| Hulu | Video | C:\Users\SAIKAT-\Downloads\635531658741248269-Screen-Shot-2014-12-02-at-10.13.59-PM.jpg | C:\Users\SAIKAT-\Downloads\1zco8lt.png | C:\Users\SAIKAT-\Downloads\http_%2F%2F.jpg |
| youtube | Video | C:\Users\SAIKAT-\Downloads\HhH80.png | C:\Users\SAIKAT-\Downloads\HTML5-Signup-Form-with-Dynamic-Captcha.jpg | C:\Users\SAIKAT-\Downloads\get-rid-or-orkut-captcha (3).png |
| Orkut | Social | C:\Users\SAIKAT-\Downloads\get-rid-or-orkut-captcha (1).png | C:\Users\SAIKAT-\Downloads\get-rid-or-orkut-captcha (2).png | C:\Users\SAIKAT-\Downloads\get-rid-or-orkut-captcha.png |
| fotolog | Picture share | C:\Users\SAIKAT-\Downloads\formvalidationsample.png | C:\Users\SAIKAT-\Downloads\captchas.jpg | C:\Users\SAIKAT-\Downloads\symfony-captcha-form-validation-example.png |
| Wikipedia | Culture | C:\Users\SAIKAT-\Downloads\Captcha_example.png | C:\Users\SAIKAT-\Downloads\220px-It-wikibooks-antispam.png | C:\Users\SAIKAT-\Downloads\Wikipedia-The_Missing_Manual_0304.png |
| blogger | Blogs | C:\Users\SAIKAT-\Downloads\chaptchas-5232021.jpg | C:\Users\SAIKAT-\Downloads\Figure-1-Samples-of-OCR-Based-CAPTCHA-Techniques.png | C:\Users\SAIKAT-\Downloads\figure42.png |
| Yahoo! | email  games | C:\Users\SAIKAT-\Downloads\yahoo-mail-password-recovery.jpg | C:\Users\SAIKAT-\Downloads\rcjVicarious.jpg | C:\Users\SAIKAT-\Downloads\captcha-3.jpg |
| Globo | email  news | C:\Users\SAIKAT-\Downloads\laravel-captcha-formvalidationexample_52.png | C:\Users\SAIKAT-\Downloads\laravel-captcha-formvalidationexample_51.png | C:\Users\SAIKAT-\Downloads\laravel-captcha-basicexample_52.png |

**2.12 Applications of CAPTCHA**

CAPTCHAs have several applications for practical security. Some of the applications are as follows:

i)Preventing Comment Spam in Blogs: Most bloggers are familiar with programs that submit bogus comments, usually for the purpose of raising search engine ranks of some website. This is called comment spam. By using a CAPTCHA, only humans can enter comments on a blog. There is no need to make users sign up before they enter a comment, and no legitimate comments are ever lost.

ii) Protecting Website Registration: Several companies (Yahoo!, Microsoft, etc.) offer free email services. Up until a few years ago, most of these services suffered from a specific type of attack: "bots" that would sign up for thousands of email accounts every minute. The solution to this problem was to use CAPTCHAs to ensure that only humans obtain free accounts

iii) Protecting Email Addresses From Scrapers: Spammers crawl the Web in search of email addresses posted in clear text. CAPTCHAs provide an effective mechanism to hide your email address from Web scrapers. The idea is to require users to solve a CAPTCHA before showing your email address. A free and secure implementation that uses CAPTCHAs to obfuscate an email address can be found at reCAPTCHA Mail Hide. Online Polls: Online polls are attacked by bots and are susceptible to ballot stuffing. This gives unfair mileage to those that benefit from it.

iv)Preventing Dictionary Attacks: CAPTCHAs can also be used to prevent dictionary attacks in password systems. The idea is simple: prevent a computer from being able to iterate through the entire space of passwords by requiring it to solve a CAPTCHA after a certain number of unsuccessful logins. This is better than the classic approach of locking an account after a sequence of unsuccessful logins, since doing so allows an attacker to lock accounts at will.

v) Search Engine Bots: It is sometimes desirable to keep webpage’s unindexed to prevent others from finding them easily. There is an html tag to prevent search engine bots from reading web pages. The tag, however, doesn't guarantee that bots won't read a web page; it only serves to say "no bots, please." Search engine bots, since they usually belong to large companies, respect web pages that don't want to allow them in. However, in order to truly guarantee that bots won't enter a web site, CAPTCHAs are needed.

vi) Preventing Unauthorized Access: The CAPTCHA mechanism prevents a hacker who tries to crack a password using Brute force method or any other password cracking method. Worms and Spam: CAPTCHAs also offer a plausible solution against email worms and spam. Spammers register themselves with free email accounts such as those provided by Gmail or Hotmail and use their bots to send unsolicited mails to other users of that email service.

**Chapter 3**

**MAPTCHA**

The last decade has seen increasing interest in abusing some of the services provided by the Internet, mainly for economical reasons. There has been misuse of on-line services: e-mail account creation (for spam sending and phishing), automatic anonymous posting (Wikipedia, blogs comments, news sites, job listing sites, etc.) for adding links for commercial promotion, or for harassment or vandalism, abuse of remote voting mechanisms (Hernandez-Castro et al., 2002), automatic site wandering for resource consumption, automatic anonymous abuse of on-line games or chat rooms for commercial promotion, etc. There are plenty of sound economical reasons to abuse services provided through the Internet. The main trend to prevent this automatic abuse has been to develop the ability to tell humans and computers apart remotely and through an untrustworthy channel. Many tests generically called CAPTCHAs1 or HIPs2 – developed with that aim. These tests rely on capacities inherent to the human mind but supposedly difficult to mimic for computers. That is, problems traditionally hard to solve with computers (as problems that still remain wide open for Artificial Intelligence researchers).

**3.1 What is MAPTCHA?**

**MAPTCHA** is a 100% effective and easy to use **CAPTCHA** for Word Press that seamlessly integrates into login, registration, lost password, comments, bbPress and Contact Form.

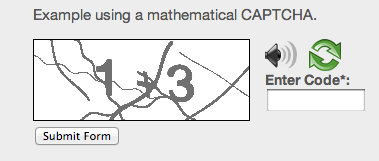
****

Fig 9: Example of MAPTCHA

**3.2 Why do we use MAPTCHA?**

The technology is used mostly to block spammers and bots that try to automatically harvest email addresses or try to automatically sign up for or make use of Web sites, blogs or forums. MAPTCHA is a part of CAPTCHA and we know that CAPTCHA, whose users include Yahoo and Google, blocks automated systems, which can't read the distorted letters in the graphic. Moreover, [6] the number of different mathematical challenges is very few and the answer for them is in single digits. Since there is no restriction on the number of retries allowed, a single-digit value will eventually pass the challenge with trial and error. It is our belief that this mathematical CAPTCHA is too complicated for the general public.

**3.3 Brief history of MAPTCHAs**

Moni Naor (Naor, 1996) is the first person that mentioned [8] some theoretical methods for telling apart computers from humans remotely. He did it looking for ways to prevent the abuse of web services. The first known use of a CAPTCHA test was done in the Alta-Vista web-search engine. Early in 1997 they faced the problem of the automatic submission of URLs to their search engine. This free ‘‘add-URL’’ service broadens its search coverage, but some users were abusing it by automating the submission of large numbers of URLS, in an effort to manipulate AltaVista’s importance ranking algorithms (that same kind of attacks have been known against other search engines, like Google). Then many people research in MATH CAPTCHA.

1) Carlos Javier Hernandez-Castro, Arturo Ribagorda made a MATH CAPTCHA [8] 11 March 2009, “Pitfalls in CAPTCHA design and implementation”.

2) Ramanpreet Kaur, Pooja made a MATH CAPTCHA in 2016, “NON OCR Approach for MATH CAPTCHA design based on Boolean algebra” [2].



Fig 10: Boolean algebra (Digital) CAPTCHA

3) A mathematical-problem-based CAPTCHA design and implementation [Hernandez-Castro and Ribagorda, 2010]. [6] This CAPTCHA challenge provides users with a complex mathematical problem with symbols, letters, and numbers as shown in Fig 11.

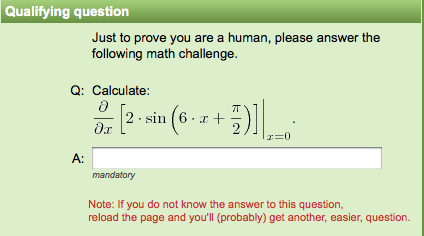


Fig 11: MAPTCHA 1

4) Kanwaldeep Kaur Kanwal, Anupama Gupta, Vivek Aggarwal, Amandeep Kaur, “Puzzle Based CAPTCHA Implementation for Noisy Environment”, in 2016 [1].



Fig 12: MAPTCHA 2

**3.4 Different Types of MAPTCHA**

**3.4.1 Puzzle based MAPTCHA [7]**

In this test, a small mathematical problem is generated according to some predefined rules. The answer to the question is then cross checked with the database answer. Solving of this problem requires an ability of understanding text of question, only a human user can answer this question.

****

Fig 13: Puzzle based MAPTCHA

Also, the shortcoming lies in the limit of the number of such questions that have a unique answer and that too which are easily recallable.

**3.4.2 QRBGS MAPTCHA [6]**

It was described the pitfalls on a mathematical-problem-based CAPTCHA design and implementation [Hernandez-Castro and Ribagorda, 2010]. This CAPTCHA challenge provides users with a complex mathematical problem with symbols, letters, and numbers as shown in Figure 3.4.

****

Fig 14: QRBGS MAPTCHA

This is an innovative idea not done in the past. One advantage of using mathematic symbols is that no commercial OCR software can recognize them while OCR software is often used in breaking CAPTCHA. This particular CAPTCHA challenge uses characters with resolution that are too low for OCR software to recognize, since the OCR software requires a resolution around 300 DPI. However, there are several design flaws in this CAPTCHA implementation. The size of each character is fixed and not distorted; therefore, allowing shape-recognition algorithm and pixel pattern matching methods to be used. Moreover, the number of different mathematical challenges is very few and the answer for them is in single digits. Since there is no restriction on the number of retries allowed, a single-digit value will eventually pass the challenge with trial and error. It is our belief that this mathematical CAPTCHA is too complicated for the general public. However, it contributes to our understanding of the importance of making CAPTCHA challenge questions and the answers vary as much as possible.

**3.4.3 CHINESE MAPTCHA [3]**

CAPTCHA of Chinese characters based on mathematical expressions, and use Delphi to achieve. Randomly generated string based on the number of Chinese characters. Using the random function can generate random numbers within a certain range, then change the random number to uppercase Chinese characters. This CAPTCHA supports addition, subtraction, multiplication, division and negative numbers and decimal, the difficulty can be set to not affect human user mental arithmetic for premise. In order to describe more intuitive algorithm, this paper does not consider the decimal and negative.

C:\Users\SAIKAT-\Pictures\Screenshots\Screenshot (1644).png

Fig 15: CHINESE MAPTCHA

* **INNOVATIONS AND ADVANTAGES OF THIS CHINESE CAPTCHA [3]**

The CAPTCHA based on English character and numeric characters currently in use is most prevalent. This CAPTCHA program automatically recognizes the high rate of security cannot be guaranteed. In this paper, the mathematical expressions used in the CAPTCHA, increasing the difficulty of the program of automatic identification, Programs need to identify the character is no longer a simple string, but two Chinese characters string by deformation and added noise. In addition, programs need to identify add, subtract, multiply, and divide operations, and convert the result of the operation to uppercase characters of Chinese. Greatly increases complexity of the program. Practice has proved that: the CAPTCHA based on the mathematical expression of Chinese characters with greater security than ordinary CAPTCHA.

**3.4.4 MATH CAPTCHA design Based on digital circuit [2]**

In Digital CAPTCHA scheme, to enhance usability and security rate as compared to current MATH CAPTCHAs, turn OCR based MATH CAPTCHA into non OCR CAPTCHA by use digital gates based on Boolean algebra rules. Boolean algebra is a technique of mathematics based on logic reasoning has some set of laws or rules like associative, negation, distributive and identity, commutative law that used to define and minimize Boolean expressions. In Boolean algebra, an expression given like A.B, A+B that transform into diagram by use logical AND, OR, NOT, NOR, NAND, etc. Boolean algebra CAPTCHA called digital CAPTCHA that construct by use digital circuits. In this, either one or more logic gate use that depict different shapes like (AND, OR, NOT, NOR, NAND) with different input symbols. Every gate executes a Boolean expression as shape indicates operation. For improve usability, built implicit CAPTCHA, users have not to type anything, by simple click on correct option from multiple option. The user select correct option relates to logic gate for pass the test. The digital CAPTCHA prove to enhance in both cases as human performances as well as automated attacks as compared to math CAPTCHAs and others.



Fig 16: Boolean algebra (Digital) CAPTCHA

**3.5 Attacks on MAPTCHA**

**3.5.1 OCR Attack [2]**

Character recognition is major problem in machine learning that easily recognize by OCR tool which aim to recognize the text. The different processing techniques are used by OCR tool which describe below:

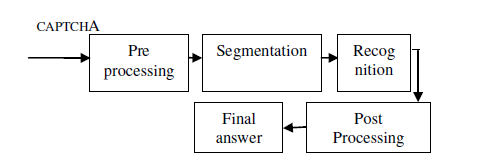


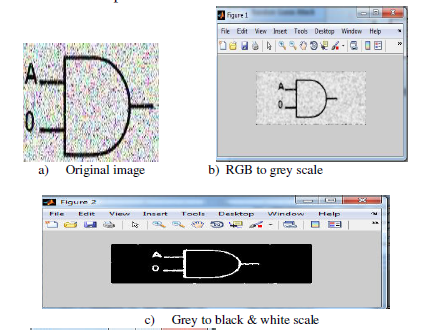
Fig 17: Process of OCR Attack

a) Pre processing:It is main process to abolish complex background and remove all additions, RGB image convert into grey scale and stored in a matrix of binary value*.*

b) Segmentation:count total components and apply segmentation on them for separation of characters.

c) Recognition:Identify same size of font or font faces, charset, distortion, blurring, tilting and waving.

d) Post processing:After recognize the characters convert them to plain text and use to break CAPTCHA test*.*



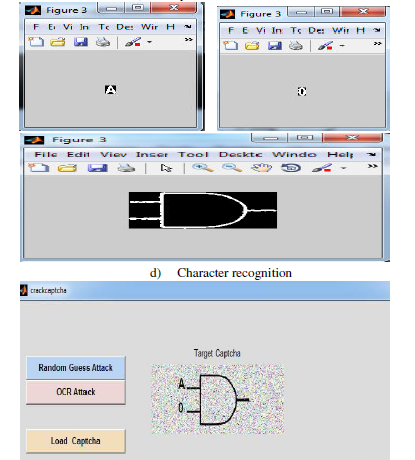


Fig 18: Example of OCR Attack on digital circuit

**3.6 Advantages and Disadvantages of MAPTCHA**

**3.6.1 Advantages**

The advantages of MAPTCHA are as follows:

**1)** In the math operation human performing an operation in the mathematics we denote add by multiply, multiply by add, divide by subtraction and subtraction by divide in the given expression. Thus the original operation is not visible to the front end and the BOT recognize it as the default operation as per the CAPTCHA image and not breakable by the BOT or OCR techniques as Fig 3.7.

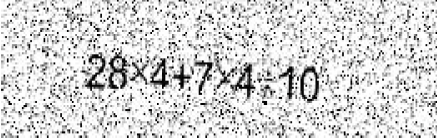
****

Fig 19: Cross operation MAPTCHA

**2)** In MAPTCHA mathematic symbols is that no commercial OCR software can recognize them while OCR software is often used in breaking CAPTCHA [6].

**3)** In MAPTCHA special features or characters (such as parentheses, brackets, braces and mathematical characters like Boolean characters) into the code is necessary to strengthen the CAPTCHA’s security Mechanism [5].

**4)** The use of mathematical CAPTCHA or skilled-test authentication alternatives has increased for a number of websites and blogging services. In comparison to conventional CAPTCHA images, mathematical require a level of comprehension that is easy for users, including older adults to understand and undertake. Computer-automated based attacks, however may not have the capability of what is being asked; thereby thwarting cyber-attacks [20].



Fig 20: MAPTCHA 3

**5)** Unlike OCR-based CAPTCHA methods, this method requires only typing a number as the answer. So it is easy to use, saves users time and more comfortable for them.

**6)** MAPTCHAs are easy to use and can be applied to or combined with a variety of other formats CAPTCHA [4].

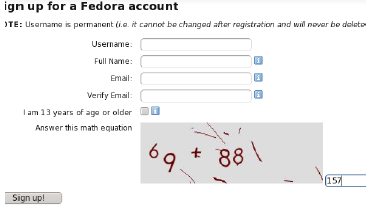


Fig 21: MAPTCHA 4

**3.6.2 Disadvantages**

The advantages of MAPTCHA are as follows:

1) In some MAPTCHAuse numeric characters (0 to 9) embed with operator symbols (+,-,\*, /, <,>, =) or also in the form of question or reasoning based on image. User has to identify to find the relation and give a correct answer. The problems seem that characters display in non trivial way and equations in form of questions are in simplest form. Due to simplicity of math CAPTCHA, it easily breaks CAPTCHA by modern online calculator API [2].

2) In place of trying hard to understand a garbled-looking word, you may be asked to find the solution to a simple mathematical problem such as “How much is 5 + 3?” Blind people as well as people who can see would be able to solve this problem. The main trick here is to incorporate problems that can be solved by all human beings irrespective of their education levels, while automated software bots would find it difficult to solve them [22].

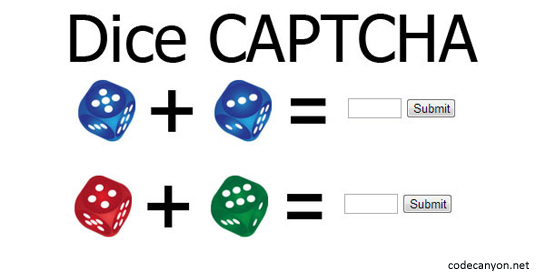


Fig 22: Dice CAPTCHA

**3.7 Applications of MAPTCHA**

**1) Protecting Registration Forms in Websites**

A number of websites such as Gmail, Hotmail, Facebook, Twitter, Yahoo, and so on offer free registration to the users. Therefore, to protect the registration process, these websites have incorporated CAPTCHAs. In fact, for any site that is offering free registration, it is best to set up a CAPTCHA, so that all the registrations are done by humans and not programs or bots.

**2) Search Engine Bots**

It is sometimes desirable to keep Web Pages unindexed to avoid them from being easily located. There is an html tag to prevent search engine bots from reading web pages. However, the tag does not guarantee that bots would not read a web page. Search engine bots, since they usually belong to large companies, respect web pages that don't want to allow them in. However, MAPTCHAs are needed in order to truly guarantee that bots won't enter a web site.

3) **Preventing Dictionary Attack**

MAPTCHAs can also be used to prevent dictionary attacks in password systems. The idea is simple: prevent a computer from being able to iterate through the entire space of passwords by requiring it to solve a MAPTCHA after a certain number of unsuccessful logins. This is better than the classic approach of locking an account after a sequence of unsuccessful logins, since doing so allows an attacker to lock accounts at will.

4) **Worms and Spam**

MAPTCHAs also offer a plausible solution against email worms and spam. A few companies are already marketing this idea of ensuring that a user only accepts emails if they are sure there is a human behind the computer.

**Chapter 4**

**Literature Survey**

Here we describe some well-known research papers dealing with MAPTCHA. From previous chapter (Chapter 3: MAPTCHA) we know that Moni Naor (Naor, 1996) is the first person who mentioned [8] some theoretical methods for telling apart computers from humans remotely. He did it looking for ways to prevent the abuse of web services. The first known use of a CAPTCHA test was done in the Alta-Vista web-search engine. Early in 1997 they faced the problem of the automatic submission of URLs to their search engine. This free ‘‘add-URL’’ service broadens its search coverage, but some users were abusing it by automating the submission of large numbers of URLS, in an effort to manipulate AltaVista’s importance ranking algorithms (that same kind of attacks have been known against other search engines, like Google). Then many people research in MATH CAPTCHA (MAPTCHA).

They are in below:

1) Puzzle Based Captcha Implementation for Noisy [1]. In this paper they did cross operation. In the cross operation the meaning of the operators are switched. For example: if human performing an operation in the mathematics we denote add by multiply, multiply by add, divide by subtraction and subtraction by divide in the given expression. Thus the original operation is not visible to the front end.



Fig 23: USER REGISTER WINDOW WITH cross operation MAPTCHA

2) A Study of CAPTCHAs for Securing Web Services [9]. In this paper, a small mathematical problem is generated according to some predefined rules. The answer to the question is then cross checked with the database answer. Solving of this problem requires an ability of understanding text of question, only a human user can answer this question. Also, the shortcoming lies in the limit of the number of such questions that have a unique answer and that too which are easily recallable.

****

Fig 24: CAPTCHAs for Securing Web Services

3) A NON OCR Approach for MATH CAPTCHA design based on Boolean algebra using digital gates to enhance web security [2]. In Digital CAPTCHA scheme, to enhance usability and security rate as compared to current MATH CAPTCHAs, turn OCR based MATH CAPTCHA into non OCR CAPTCHA by use digital gates based on Boolean algebra rules. Boolean algebra is a technique of mathematics based on logic reasoning has some set of laws or rules like associative, negation, distributive and identity, commutative law that used to define and minimize Boolean expressions. In Boolean algebra, an expression given like A.B, A+B that transform into diagram by use logical AND, OR, NOT, NOR, NAND etc. Boolean algebra CAPTCHA called digital CAPTCHA that construct by use digital circuits. In this, either one or more logic gate use that depict different shapes like (AND, OR, NOT, NOR, NAND) with different input symbols. Every gate executes a Boolean expression as shape indicates operation. For improve usability, built implicit CAPTCHA, users have not to type anything, by simple click on correct option from multiple option. The user select correct option relates to logic gate for pass the test. The digital CAPTCHA prove to enhance in both cases as human performances as well as automated attacks as compared to math CAPTCHAs and others.



Fig 25: Boolean algebra (Digital) CAPTCHA

4) A mathematical-problem-based CAPTCHA design and implementation. This CAPTCHA challenge provides users with symbols, letters and numbers as shown in Fig 26.



Fig 26: MAPTCHA 5

One advantage of using mathematical symbols is that no commercial OCR software can recognize them while OCR software is often used in breaking CAPTCHA.

5) Design and improvement of Chinese CAPTCHA based on mathematical expression[3]. In this CAPTCHA, Chinese characters based on mathematical expressions, and use Delphi to achieve. Randomly generated string based on the number of Chinese characters. Using the random function can generate random numbers within a certain range, then change the random number to uppercase Chinese characters. This CAPTCHA supports addition, subtraction, multiplication, division and negative numbers and decimal, the difficulty can be set to not affect human user mental arithmetic for premise. In order to describe more intuitive algorithm, this paper does not consider the decimal and negative. Add noise in the CAPTCHA In order to increase the difficulty of the program to automatically identify, add interference noise to the picture.

C:\Users\SAIKAT-\Pictures\Screenshots\Screenshot (1644).png

Fig 27: Chinese MAPTCHA

6)An Implementation of A Geometric and Arithmetic CAPTCHA without Database [4]. Geometric CAPTCHA is developed with HTML5 and Java Script with adobe Dreamweaver cs5 software that can run on virtually any type of web browser and can create random shapes, so do not use a database. These two advantages make this CAPTCHA easy to use. The user must enter the correct result into the designated slot. The result is a positive integer and computing sequence from left to right. The characteristic of geometric CAPTCHA is shown in Fig 28.



Fig 28: Characteristic of geometric CAPTCHA

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SL.  No | Paper Name | Disadvantage | Success Rate (In %) | Response Time (in sec) | Our Proposed  Work |
| 1 | Puzzle Based Captcha Implementation for Noisy Environment [1]. | 1) Very difficult to understand.  2) Takes a long time for solving. | 85 | 12 | We remove that difficulty and we use any one symbol (multiplication or addition) with numbers. |
| 2 | A Study of CAPTCHAs for Securing Web Services [9]. | 1)Very complicated.  2) Takes a long time for solving. | 84 | 15 | We remove that complicated problem, so that human can easily solve our MAPTCHA in a quick time. |
| 3 | A mathematical-problem-based CAPTCHA design and implementation [7]. | 1)Very complicated, only a mathematician can solve it.  2) Takes very long time. | 68 | 40 | We made our MAPTCHA easy for every human, but not for bots. |
| 4 | A NON OCR Approach for MATH CAPTCHA design based on Boolean algebra using digital gates to enhance web security [2]. | 1)Everyone doesn’t know about logic gates and its operation.  2) Also takes a long time for solving. | 75 | 21 | We cannot use any logic gate, we use only number, so that people of different ages can understand our MAPTCHA easily. |
| 5 | Design and improvement of Chinese CAPTCHA based on mathematical expression [3]. | 1)Everyone doesn’t know Chinese characters.  2) Complicated to understand. | 71 | 35 | We don’t use any Chinese characters; we use English characters and numeric number. |
| 6 | Implementation of A Geometric and Arithmetic CAPTCHA without Database [4]. | 1) Difficult to use in real applications. | 82 | 12 | We can use our MAPTCHA in real applications. |

**Table 2: Comparative study of existing MAPTCHA systems**

**Chapter 5**

**Proposed Methodology**

To overcome the disadvantages of different kinds of MAPTCHA as we have already discussed we have built our own MAPTCHA. It basically consists of texts rotated at different angles clockwise / anticlockwise which are shown as images to the human agent in a Registration form. This rotation of images at different angles is done by using random function. And for the clockwise / anticlockwise rotation also we have used again the random function.

* **Our Proposed MAPTCHA Algorithm MathSec:**

The algorithm that we have used for our own newly generated MAPTCHA is:

Step1: Take the left operand using the **texttoimage** algorithm.

Step2: Take the operator as Addition (+) or Multiplication(x) depending on the random value generator and convert it into an image.

Step3: Take the right operand using the **texttoimage** algorithm.

Step4: Merge the images together into a single image.

Step5: Crop the image in a proper dimension so that it can be used in the Registration form.

Step6: Use the cropped image in the Registration form.

Step7: User calculates the result of the image and enters it into the blank box and clicks on Submit.

Step8: If the entered calculation is correct then the user displayed a message showing successful registration otherwise it shows error that user calculated wrongly.

Step9: There is also an option of Refresh using which users can get another MAPTCHA if he/she has problem with the current one.

**texttoimage** algorithm goes like this:

Step1: Take some random text which can be numerical or alphabetical. Example- 7 or Seven.

Step2: Convert the text into its corresponding image.

Step3: Rotate the image clockwise or anticlockwise depending on the random value generator at a particular angle.

* **Flowchart of Our Proposed MAPTCHA Algorithm MathSec:**

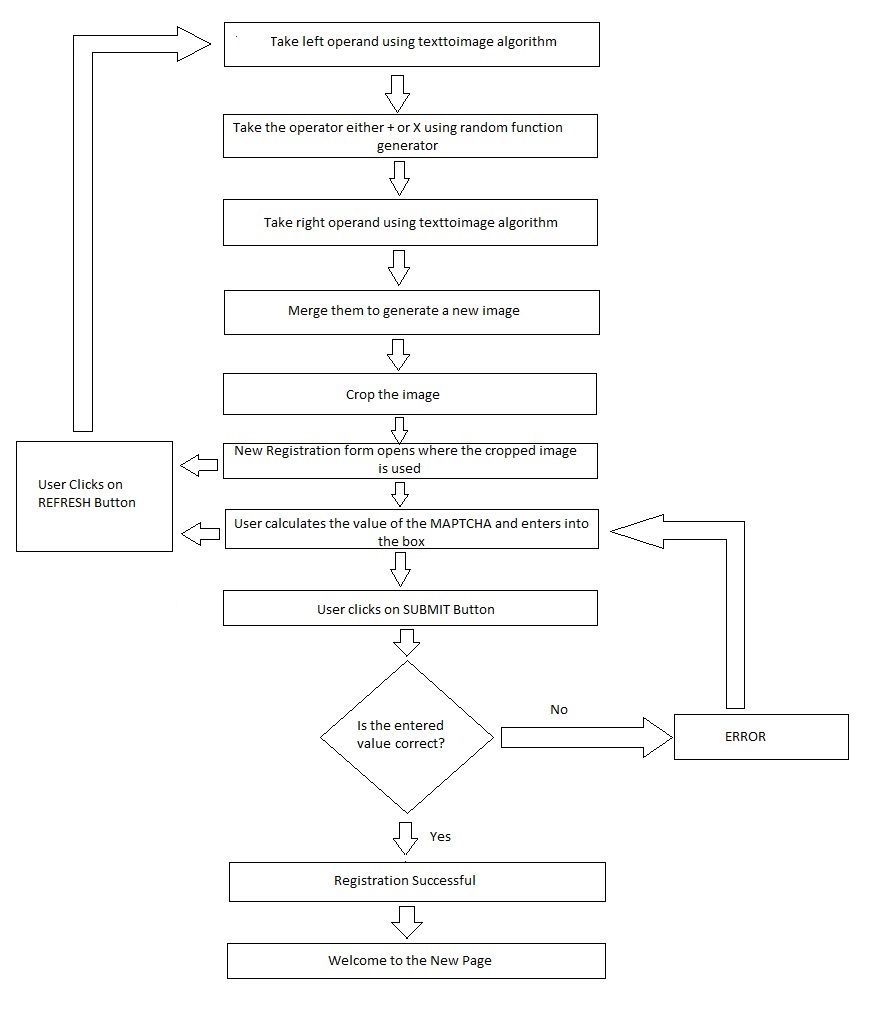


Fig 29: Flowchart of our proposed MAPTCHA Algorithm

**Chapter 6**

**Implementation**

In this chapter we will implement our proposed algorithm as discussed in the previous chapter i.e. Chapter 5. The software that we have used for implementation is MATLAB 2017a.

Step1: Let us take the left operand to be 3 (in numeric) which is generated with help of random value generator in MATLAB.

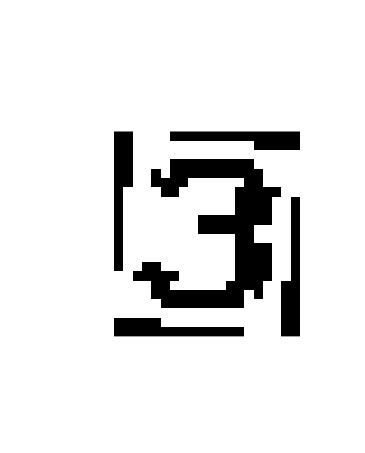


Fig 30: Step1

Step2: Then we get an operator to be x using random value generator in MATLAB.

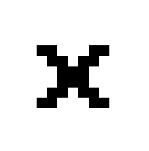


Fig 31: Step2

Step3: Then we take the right operand to be 7 (in alphabetic i.e. Seven) which is generated with help of random value generator in MATLAB.



Fig 32: Step3

Step4: Merge all the three images together to get a single image.

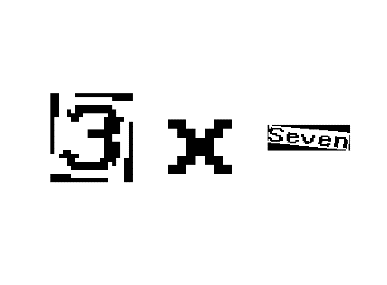


Fig 33: Step4

Step5: Crop the image to use in the Registration form.

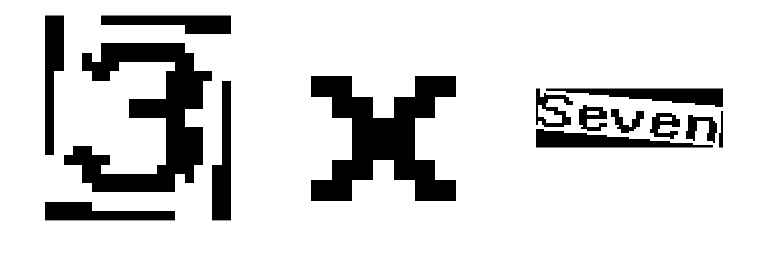


Fig 34: Step5

Step6: Use the cropped image in the Registration form.

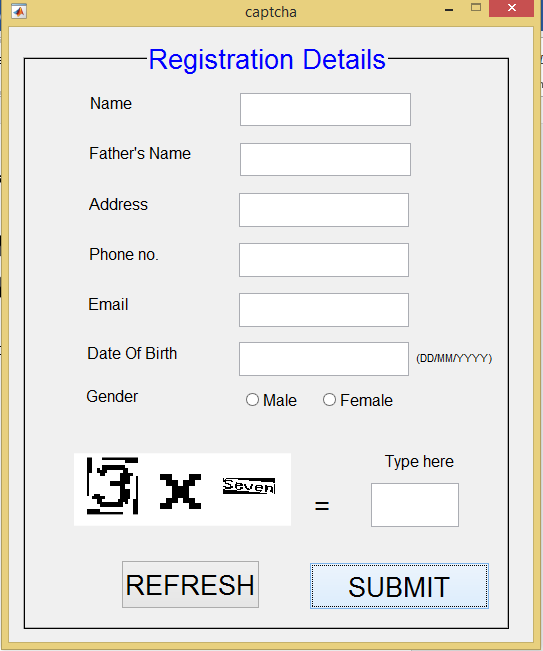


Fig 35: Step6

Step7: User calculates the result of the image and enters it into the blank box and clicks on Submit.

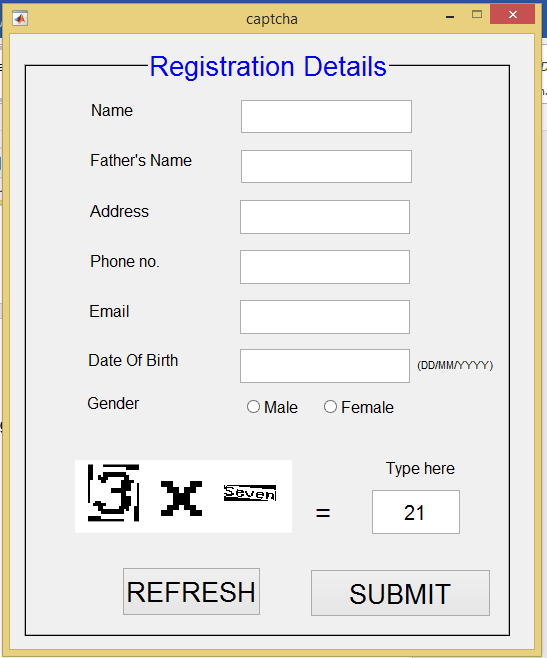


Fig 36: Step7

Step8: If the entered calculation is correct then the user displayed a message showing successful registration otherwise it shows error that user calculated wrongly.

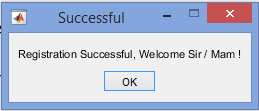


Fig 37: Step8

It shows error like this:

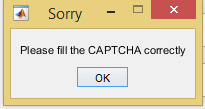


Fig 38: Error Message

For this case:



Fig 39: Entering wrong value

Step9: There is also an option of Refresh using which users can get another MAPTCHA if he/she has problem with the current one.



Fig 40: Step9

**Chapter 7**

**Result and Analysis**

We have used the software MATLAB 2017a for obtaining the results. We have rotated the operands at a particular angle which is 60 degree. We have taken 60 degrees because it gave us good results than 0 degree,15 degrees,30 degrees,45 degrees,75 degrees and 90 degrees. We have multiplied some values with these 60 degrees randomly which is generated using a random function inbuilt in MATLAB. This random function basically generates a random value in the range of 0 to 1 in which the values are distributed uniformly.

* **rand() function**

rand() function is a function which is inbuilt in the MATLAB itself. It basically generates some random value in the interval of 0 to 1 randomly. The probability that any number in between this interval can appear is equally likely. We have used rand() function in many places.

Choosing alphabetic or numeric is done by rand() function. For choosing operator between + or X also rand() function is used. Random values are generated and multiplied with 60 degree to get different angles. For clockwise or anticlockwise again rand() function is used.

* **Optimum angle**

We have rotated the images taking 60 as the optimum angle, because it gave us good results than 0 degree,15 degrees,30 degrees,45 degrees,75 degrees and 90 degrees. All the observations are summarized in Table 3 – 9.

* **Working Principle**

What we have done is basically we have taken 10 random values for k1 and k2. Where k1 and k2 are random variables. Then we have found the average of these 10 random values for each of k1 and k2. Then to find the angle in which image1 (left operand) should be rotated we have multiplied the particular angle with the k1 variable and in the same way to find the angle in which image2 (right operand) should be rotated we have multiplied the particular angle with the k2 variable

* **The observation tables are as follows:**

**Table 3: Observation table of 0 degree**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ANGLE(DEGREE) | K1(RANDOM VALUE) | K2(RANDOM VALUE) | AVG OF K1 | AVG OF K2 | IMG1 | IMG2 |
| 0 | 0.3502 | 0.6620 | 0.4121 | 0.5167 | 0 | 0 |
| 0.4468 | 0.3063 |
| 0.3786 | 0.8116 |
| 0.6225 | 0.5870 |
| 0.1948 | 0.2259 |
| 0.4302 | 0.1848 |
| 0.4087 | 0.5949 |
| 0.2967 | 0.3188 |
| 0.0292 | 0.9289 |
| 0.9631 | 0.5468 |

**Table 4: Observation table of 15 degrees**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ANGLE(DEGREE) | K1(RANDOM VALUE) | K2(RANDOM VALUE) | AVG OF K1 | AVG OF K2 | IMG1 | IMG2 |
| 15 | 0.8280 | 0.8441 | 0.4025 | 0.5799 | 6.0375 | 8.6985 |
| 0.6987 | 0.9281 |
| 0.3523 | 0.1172 |
| 0.0884 | 0.2059 |
| 0.1605 | 0.1237 |
| 0.1153 | 0.8060 |
| 0.6314 | 0.9674 |
| 0.4349 | 0.9808 |
| 0.2602 | 0.0857 |
| 0.4549 | 0.7401 |

**Table 5: Observation table of 30 degrees**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ANGLE(DEGREE) | K1(RANDOM VALUE) | K2(RANDOM VALUE) | AVG OF K1 | AVG OF K2 | IMG1 | IMG2 |
| 30 | 0.7425 | 0.3532 | 0.4717 | 0.6240 | 14.151 | 18.72 |
| 0.0682 | 0.3643 |
| 0.0526 | 0.5127 |
| 0.2830 | 0.9215 |
| 0.2587 | 0.9253 |
| 0.9942 | 0.4519 |
| 0.2630 | 0.9321 |
| 0.4273 | 0.7626 |
| 0.7171 | 0.1497 |
| 0.9107 | 0.8666 |

**Table 6: Observation table of 45 degrees**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ANGLE(DEGREE) | K1(RANDOM VALUE) | K2(RANDOM VALUE) | AVG OF K1 | AVG  OF K2 | IMG1 | IMG2 |
| 45 | 0.2057 | 0.4714 | 0.3790 | 0.4969 | 17.055 | 22.3605 |
| 0.5699 | 0.7806 |
| 0.4632 | 0.0178 |
| 0.1699 | 0.6743 |
| 0.8742 | 0.3126 |
| 0.1081 | 0.5399 |
| 0.0639 | 0.2201 |
| 0.4002 | 0.4723 |
| 0.0188 | 0.8807 |
| 0.9166 | 0.5990 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ANGLE(DEGREE) | K1(RANDOM VALUE) | K2(RANDOM VALUE) | AVG OF K1 | AVG OF K2 | IMG1 | IMG2 |
| 60 | 0.8480 | 0.1662 | 0.5083 | 0.5087 | 30.498 | 30.522 |
| 0.5247 | 0.9124 |
| 0.0837 | 0.3903 |
| 0.5401 | 0.3659 |
| 0.1592 | 0.9091 |
| 0.0968 | 0.1771 |
| 0.6745 | 0.1052 |
| 0.7510 | 0.2699 |
| 0.5651 | 0.9214 |
| 0.8399 | 0.8699 |

**Table 7: Observation table of 60 degrees**

**Table 8: Observation table of 75 degrees**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ANGLE(DEGREE) | K1(RANDOM VALUE) | K2(RANDOM VALUE) | AVG OF K1 | AVG OF K2 | IMG1 | IMG2 |
| 75 | 0.0975 | 0.2785 | 0.4412 | 0.4262 | 33.09 | 31.965 |
| 0.9572 | 0.4854 |
| 0.9595 | 0.6557 |
| 0.7431 | 0.3922 |
| 0.0462 | 0.0971 |
| 0.4387 | 0.3816 |
| 0.6463 | 0.7094 |
| 0.1190 | 0.4984 |
| 0.2551 | 0.5060 |
| 0.1493 | 0.2575 |

**Table 9: Observation table of 90 degrees**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ANGLE(DEGREE) | K1(RANDOM VALUE) | K2(RANDOM VALUE) | AVG OF K1 | AVG OF K2 | IMG1 | IMG2 |
| 90 | 0.3521 | 0.1966 | 0.6327 | 0.2815 | 56.943 | 25.335 |
| 0.5853 | 0.5497 |
| 0.5678 | 0.0759 |
| 0.5688 | 0.4694 |
| 0.5285 | 0.1656 |
| 0.4505 | 0.0838 |
| 0.9961 | 0.0782 |
| 0.8173 | 0.8687 |
| 0.9106 | 0.1818 |
| 0.5499 | 0.1450 |

In this way we have obtained all the angles in which image1(left operand) and image2(right operand) are rotated. After examining the results, we have come to the conclusion that 60 degrees should be the optimum angle. So, we have chosen 60 degrees as the optimum degree.

**Chapter 8**

**Future Work**

As we know that each and every CAPTCHA should possess two important properties, that they should be robust and usable. So, robust means that it should be very much secure to defend against the bots. Whereas, usable means that it should be easy for the human to solve them.

So, in future we will try to make our proposed MAPTCHA algorithm to be more robust and usable. To make it robust we will try to do different kind of attacks to break our MAPTCHA and improve the various aspects where it is requiring improvements to make it more secure. OCR (Optical Character Recognition) can be one of those attack techniques, where we will try to identify each character. And to make it more usable we will try to do our MAPTCHA as simple as we can do for human to read them and solve them more efficiently and quickly.

**Chapter 9**

**Conclusion**

CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) is a type of challenge-response test used in computing to determine whether or not the user is human. The term was coined in 2003 by Luis von Ahn, Manuel Blum, Nicholas J. Hopper, and John Langford.

In this paper we have proposed a Mathematical CAPTCHA or MAPTCHA algorithm. MAPTCHA is a type of CAPTCHA in which users have to calculate the value of the expressions and enter it in the given box. We have implemented our proposed MAPTCHA algorithm and we have found that it is secure and usable.

In future, our objective is to increase the robustness and usability of our proposed Algorithm.

**Chapter 10**

**References**

1) Kanwaldeep Kaur Kanwal, Anupama Gupta, Vivek Aggarwal, Amandeep Kaur, “*Puzzle Based Captcha Implementation for Noisy Environment*”, 142001 (Punjab), India, ISSN: [2395-6992] [Vol-2, Issue-11, November- 2016].

2) Ramanpreet Kaur, Pooja, “*A NON OCR Approach for Math CAPTCHA design based on Boolean algebra using digital gates to enhance web security*”, CT Group of Institutions Shahpur Jalandhar, India, 2016 IEEE.

3) HE Ding-hua, “*Design and improvement of Chinese CAPTCHA based on mathematical expression*”, Department of Computer Wuhan polytechnic Wuhan, China, 2012 International Conference on Computer Science and Service System.

4) Puchong Subpratatsavee, Preeyawal Kuha, Narongrit Janthong, Chanchira Chintho, “*An Implementation of A Geometric and Arithmetic CAPTCHA without Database*”, Department of Apply Mathematics Faculty of Science at Si Racha, Kasetsart University

Chonburi, Thailand, 2014 IEEE.

5) J.O. Okesola(School of Computing University of South Africa South Africa), Longe O. B.( Department of Computer Science & Mathematics Adeleke University, Ede, Osun State, Nigeria.), A.P. Obi, “*Towards the Development of a Time-Out Multiple C-R CAPTCHA Framework Using Integrated Mathematical Modeling*”, African Journal of Computing & ICT Reference Format, Vol 8. No. 2 June, 2015.

6) HIEU TRONG HO, “*Enhancing WWW Security Through CAPTCHA by Leveraging Cognitive Factors*”, Department of Computer Science in the Graduate School California State University San Marcos December 2011.

7) Adarsh Baluni, Sayali Gole, “*Two-Step CAPTCHA: Using a Simple Two Step Turing Test to Differentiate between Humans and Bots*”, University of Pune S.K.N.C.O.E- Dept. of I.T. Vadgaon (Bk.), Pune-41, India,Volume 81 – No 16, November 2013.

8) Carlos Javier Hernandez-Castro, Arturo Ribagorda, “*Pitfalls in CAPTCHA design and implementation: The Math CAPTCHA, a case study*”, Security Group, Department of Computer Science, Carlos III University, Avd Universidad 30, 28911 Leganes, Madrid, Spain, 11 March 2009.

9) M. Tariq Banday, N. A. Shah, "*A Study of CAPTCHAs for Securing Web Services*" International Journal of Secure Digital Information Age, Vol. 1. No. 2, December 2009.

10) Suliman A. Alsuhibany, “*Optimising CAPTCHA Generation*”, School of Computer Science Newcastle University Newcastle, UK, 2011 Sixth International Conference on Availability, Reliability and Security.

11) Elie Bursztein, Matthieu Martin, and John C. Mitchell, “*Text-based CAPTCHA Strengths and Weaknesses*”, Stanford University, Chicago, Illinois, USA, October 17–21, 2011.

12) Jeff Yan, Ahmad Salah El Ahmad “*A Low-cost Attack on a Microsoft CAPTCHA*”,

School of Computing Science, Newcastle University, UK*,* October 27–31, 2008.

13) Jeff Yan and Ahmad Salah El Ahmad “*Pixel-Count Attacks: CAPTCHA Security*,” Newcastle University, England*,* CO Published by the IEEE Computer and Reliability Societies, July/August 2009.

14) Qi Ye, Youbin Chen (Department of Electronic Engineering Tsinghua University Beijing, China), Bin Zhu (Multimedia Group Microsoft Research Asia Beijing, China), “*The Robustness of a New 3D CAPTHCHA*”, 978-1-4799-3243-6/14 © 2014 IEEE.

15) Panagiotis Germanakos, Marios Belk “*E-book: Human–Computer Interaction Series*” Jean Vanderdonckt, Université catholique de Louvain, Belgium, ISSN 1571-5035, ISBN 978-3-319-28050-9 (eBook), Springer International Publishing Switzerland 2016.

16) Tomoka Azakami, Ryuya Uda, “*Effective CAPTCHA with Amodal Completion and Aftereffects by Complementary Colors and Difference of Luminance*”, School of Computer Science Tokyo University of Technology Hachioji, Tokyo, Japan, 2016 IEEE.

17) Jeff Yan, Ahmad Salah El Ahmad “*Breaking Visual CAPTCHAs with Naïve Pattern Recognition Algorithms*”,School of Computing Science, NewcastleUniversity*,* UK*,* 2007 IEEE.

18) Misako Goto, Toru Shirato, Ryuya Uda, “*Text-Based CAPTCHA Using Phonemic Restoration Effect and Similar Sounds*”, School of Computer Science Tokyo University of Technology Tokyo, Japan, 2014 IEEE.

19) Tomoka Azakami, Ryuya Uda “Effective *CAPTCHA with Amodal Completion and Aftereffects by Complementary Colors and Difference of Luminance*”, School of Computer Science Tokyo University of Technology Hachioji, Tokyo, Japan, 2016 IEEE.

20) Marc Alexander Kowtko, “*Biometric Authentication for Older Adults*”, Seidenberg School of Computer Science and Information Systems Pace University 861 Bedford Road, Pleasantville, NY 10570, 2014 IEEE.

21) Retrieved from website: <https://en.wikipedia.org/wiki/CAPTCHA>.

22) Retrieved from website: <http://techvoi.com/4-benefits-of-captcha-that-make-it-a-must-have-for-every-website/>.