Computer
& Robotics

IMPTCHA: A Creative Image CAPTCHA

Reza Shali

Faculty of Computer and Information Technology Engineering, Qazvin Branch, Inslamic Azad University, Qazvin, Iran

Received 21 January 2020; accepted 26 December 2020

Abstract

We present IMPTCHA, a new CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) as a security measure to recognize human users. The proposed system uses images instead of distorted text to label images as a valuable output. IMPTCHA is generated from images on the Web. For passing this CAPTCHA, users must type two words for description of two images. When users pass the challenge, the provided meaningful labels are used to determine the content of images. In addition, semantic graphs for labels and images are created and according of it we'll able to develop an image semantic search engine. Due to usage of images in this system, and its architecture, it is highly secure compared to its counterparts. In a user study involving 60 participants, IMPTCHA's word accuracy is measured to be 98.18% while 61.26% of users could pass the challenge.

Keywords:*CAPTCHA, Ontology, Image labeling, Security*

1.Introduction

In recent years, security has been an important aspect of the Web. Many operations such as banking transactions, registering, etc. are being performed on the Web. As a result, the Web administrators are often struggling to protect their websites against malicious attacks. One of the most popular and widespread security measures is a service that prevents bots and automated scripts from abusing websites. This service is known as CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart). Email provider stake advantage of CAPTCHAs in their registering forms; weblogs use it to prevent automated programs from spamming their website, etc.

There are a number of CAPTCHA systems; however, all of them face security problems. Currently reCAPTCHA [1] is the most resistant system against attacks; nevertheless, it has been cracked by Stanford University researchers [2].The proposed system, called IMPTCHA, aims to solve this problem by using images instead of distorted text. In addition to security, the labels provided by the users to pass the challenge can be used to determine the content of images grabbed from the web. This is similar to reCAPTCHA's approach which uses the provided words to digitize books.

The rest of this paper is organized as follows. Section 2 reviews related works. IMPTCHA is described in more detail in section 3. Experiments and their results are given in section 4, while section 5 concludes.

2.Related Work

A CAPTCHA is a program which is designed based on Automated Turing Test [3]. A Turing Alan Turing in 1950 [4] introduced it to test a machine's ability to exhibit intelligent behavior [4]. A machine will pass the test successfully if a person outside of a room cannot distinguish it from a human in the room only based on some textual interactions and their replies.

In the rest of this section some of the most effective and popular CAPTCHA systems and their features are discussed. By the end of this section, a comparison of IMPTCHA with them is presented.

2.1.Overview of CAPTCHA Projects

Based on the Turing test, CAPTCHAs create challenges which are not easily performed by computers. This usually involves an artificial intelligence problem, such as natural language processing, character

[*] Corresponding Author. Email: keyvanrezashali@gmail.com

recognition, speech recognition and image understanding. Due to this fact, there are a variety of different CAPTCHA systems.

In 1997, Moni Naor presented the first idea based on Turing test to distinguish a computer from human, which was not published [3]. This manuscript contains several crucial notions and intuitions which led to creation of CAPTCHA. The first practical example of Automated Turing test was a system developed by AltaVista which prevented bots from automatically registering web pages using images containing distorted texts [3].

CAPTCHA: In 2000, Luis Von Ahn, introduced CAPTCHA. Challenges of this system, which a user must solve it in order to pass a page, is composed of distorted and noisy random characters generated by a computer program [3]. Fig. 1(a) presents an example of it.

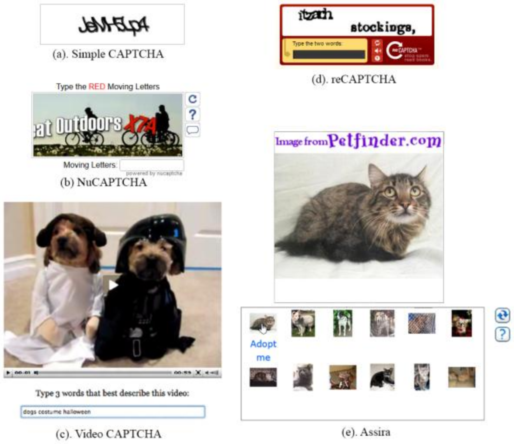
reCAPTCHA: Von Ahn presented reCAPTCHA[5] with slogan of “read book, stop spam” in 2007. As it is comprehensive from Fig. 1(d), challenges of reCAPTCHA contain two words. One of them is an intentionally distorted text, for which the answer is known, and the other one is photographically scanned from a physical book and the “Optical Character Recognition” (OCR) program cannot read it. The process of digitizing physical books helps making them searchable while decreasing the resources needed to store or transfer them. When a user attempts to solve the challenge, if the provided word for the known image is correct, the answer for the other image is believed to be correct. If several users enter the same value for the unknown word, the system will be confident enough to trust their answer. This has been proved to be effective with 99% accuracy.

Assira (Animal Species Image Recognition for Restricting Access): Jeremy Elson introduced Assira, an image CAPTCHA, in 2007. Assira displays twelve images from a database of more than three million photos and ask the user to identify all photos of cats or dogs [6]. A sample is provided in Fig. 1(e).

NuCAPTCHA: In 2008, NuCAPTCHA [7] was introduced as an animated CAPTCHA. Fig. 1(b) depicts a sample of it. In this system, characters are animated and in order to solve the challenge, the user must type these animated characters.

Video CAPTCHA: Kluever and Zanibbi [8], in 2008, introduced the video CAPTCHA. This CAPTCHA uses social video which is tagged by users on the web. In this system, a video will be displayed and to successfully pass page including Video CAPTCHA, user must describe it three words.

2.2.Features of CAPTCHA projects Vs IMPTCHA’s Features



(a). Simple CAPTCHA

(d). reCAPTCHA

(b) NuCAPTCHA

(c). Video CAPTCHA

(e). Assira

Fig. 1: Samples of different CAPTCHA – (a) Simple CAPTC [1], (b) NuCAPTCHA [4], (c) Video CAPTCHA [5], (d) re CAPT

All CAPTCHAs have a common goal which is providing a condition to prevent exploitation by bots and automated scripts in specific Web pages. However, each of them has different features and characteristics. In the rest of this section, the discussed CAPTCHAs are compared based on five important criteria: security, added benefit, ease of use, bandwidth usage and item count.

2.2.1 Security

A research team from the Stanford University has created a tool named DeCAPTCHA [2] to attack some CAPTCHAs of a few popular websites such as Wikipedia, eBay, CNN, etc. This tool has been tested on 15 web sites. Its success rate is shown in Table 1.

CAPTCHA’s security, this team has recommended using black and white characters. Additionally, taking advantage of complex backgrounds and lines over characters could help preventing bypassing the CAPTCHA system.

The DeCAPTCHA team from Stanford University has used this feature to successfully attack it [2,9].

Animated CAPTCHA is also cracked by DeCAPTCHA. The cracking algorithm for this CAPTCHA is presented in 5 phases. The first phase will extract frames from the animation. In the second phase the background is removed and the letters will be shown in white color with the black background. The third phase will merge captured frames to determine character locations. The characters related to the CAPTCHA will be extracted in the fourth phase and a machine learning algorithm will extract each of the characters in the fifth phase.

Table1
Success Rate of DeCAPTCHA on some CAPTCHAs

Success Rate	Web Site
1-10 %	Baidu, skyrock
10-24 %	CNN, Digg
25-49 %	eBay, Reddit, Slashdot, Wikipedia
50% or Greater	Authorize, Blizzard, Captcha.Net, MegaUpload, NIH

Short usage of Video CAPTCHAs may be related to their low performance and lack of security. In theory, it is Possible to bypass these CAPTCHAs using image matching algorithms such as SIFT (Scale Invariant Feature Transform) [10].

Finally, the common CAPTCHAs used in the web, share two important issues:

1. These CAPTCHAs have been cracked by many robots, thus the main goal of using these CAPTCHAs which is protecting web sites from robot attacks will not be satisfied.
2. While many benefits can be gained from solving these CAPTCHAs, they don't have any added values. Even in reCAPTCHA which uses this information to digitize books, it is possible to not participate in this act.

In general, security is an issues for all the CAPTCHAs discussed in this paper. Even though reCAPTCHA is considered one of the most secure CAPTCHAs, has been cracked. In other words, any text-based CAPTCHA, including reCAPTCHA can be attacked using optimized versions of currently available OCR algorithms. However, because in IMPTCHA two different images are presented to the user to label and because labeling images automatically is a difficult task, it could be considered the most secure approach in this regards.

2.2.2. Added Benefit and Useful output

None of the discussed CAPTCHAs have any added benefit except reCAPTCHA; however, even reCAPTCHA is facing the problem of not being able to have that benefit. Since the word extracted from the eBook is not very much distorted, it is usually easy to spot. In consequence, if a user knows about the reCAPTCHA's underlying structure, he/she can opt not to help the system and only provide the necessary label to solve the challenge; hence the added benefit of the system which is digitizing text is removed.

In IMPTCHA on the other hand, two different images will be presented to the users with no sign to differentiate them whatsoever; hence, the users will have to enter labels for both images to solve the IMPTCHA challenge. For this reason, IMPTCHA is the only CAPTCHA which can provide the added benefit, 100% of the time. This means, the labels provided by the users can be useful. For instance, search engines can search content of images using these labels.

2.2.3. Ease Of Use

Based on a survey in which we asked a few questions from 60 users in our tests for IMPTCHA, we compared currently available CAPTCHAs for their ease of use. In reCAPTCHA, in order to provide more security in the system, the distorted word which responsible for differentiating humans from computers, is becoming more complex every day. Most of the users participated in our survey were unhappy about the complexity of these images. They mentioned that there are a lot of times that these texts are very hard to read, even for humans.

Additionally, some other CAPTCHAs have a problem with their solving time. Picking a few images from a group of images in Assira [6], watching a video clip for labeling or solving an animated CAPTCHA is more time consuming than labeling a single image.

In IMPTCHA, none of these problems exist. Labeling two images is more pleasant than recognizing a distorted text while it doesn't take more time than conventional CAPTCHAs.

Table 2
A comparison between YAPPTCHA and other CAPTCHAs

Parameter/CAPTCHA	reCAPTCHA	NuCAPTCHA	Video CAPTCHA	Assira	YAPPTCHA
Security against DeCAPTCHA	Text: 0% Voice: 1%	90%	--	--	--
Added Value	Mostly	No	No	No	Always
Easily Recognizable Items	No	Yes	Average	Yes	Yes
Bandwidth Usage	~5KB	~50KB	~600KB	~130KB	~8KB
Items to Recognize	2	1	3	12	2

2.2.4. Bandwidth Usage

In today's web, the size of web pages matters. Because mobile data plans are still expensive, administrators are optimizing their web sites for mobile browsers. Even the number of the HTTP requests matter. Keeping this in mind, a video, or an animated CAPTCHA can cost a lot of money overtime for the users. Using 12 images in Assira has the same issue. In IMPTCHA, the size of image presented to the users is no more than 9 KB, even with colored images.

2.2.5. Item Count

One of the important factors in designing a CAPTCHA system is the number of items a user should solve in order to pass the challenge. The number of items which can affect user's experience and the system's security could introduce a tradeoff in the system. In Assira, twelve images should be processed by the user. NuCAPTCHA needs one word for three characters and the video CAPTCHA requires three words from users. In conventional CAPTCHAs processing one item would be enough while reCAPTCHA and IMPTCHA both require labels for two images.

The information provided in this section, points out the issues in other CAPTCHAs which initiated the intention of creating IMPTCHA.

Table 2 summarizes all the information presented in this chapter.

3. IMPTCHA

As mentioned in previous section, IMPTCHA was designed with the purpose of preventing spasm and labeling images without the issues with other kinds of CAPTCHA.

Fig. 2 depicts IMPTCHA in English 2(a) and Farsi 2(b). In this system, two images are used. One of them is labeled manually, and the other one is expected to be labeled by the users. In contrary to reCAPTCHA, in this system the labeled image is not distinguishable from the other one and the user is expected to label both images to get through IMPTCHA. In the rest of this section, first the architecture of IMPTCHA and its features are described and next the features in IMPTCHA which are not available in related projects are presented.

3.1 Architecture of IMPTCHA

IMPTCHA's architecture includes a few important aspects. In this section, each of these aspects is discussed in detail.

3.1.1 API

The proposed system exposes an API to enable any website to use it. Using IMPTCHA in a web page involves an 11-step process. The client/server model forth is process is shown in Fig. 3.

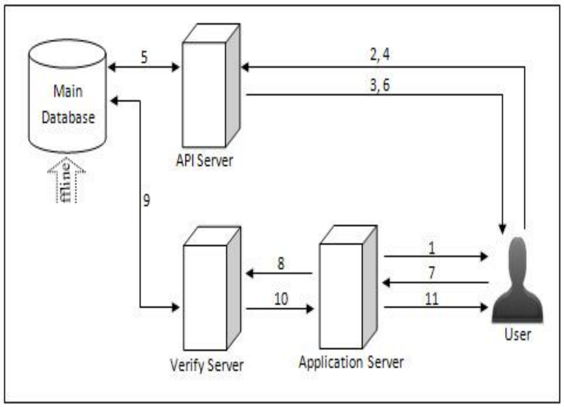


Fig. 2: Client/Server Model for YAPPTCHA

From this Fig.3, 11 procedures are comprehensible:

1) First, the web page in which an empty IMPTCHA form is presented is sent to the user along with a public key by the application server.
2) User's browser sends the public key to the API server and requests a IMPTCHA from the server through AJAX.
3) In the API server, all the information required for creating a IMPTCHA along with a unique code named IMPTCHA token which represents this information is generated, stored in the database and sent to the user.
4) The JavaScript code then writes the IMPTCHA token to certain attributes of necessary elements in DOM. Consequently, the user's browser sends a request for IMPTCHA's image by sending the token to the API server.
5) The API server looks up and then loads the information for the token in the database

6) Both labeled and non-labeled images are loaded in the API server and a single image gets created and sent as the IMPTCHA image to the user.
7) When the user solves the IMPTCHA he/she sends labels along with the public key and the IMPTCHA token to the application server by clicking the submit button of the form.
8) The application appends its own private key to this information and sends them to the verification server. The private key prevents any other party to send requests on behalf of the application server.
9) The verification server first verifies the public and private keys match. It then looks up, loads and then removes the information which the IMPTCHA token represents from the database. If the entered label was correct, it adds the label for un-labeled image.
10) The result of the entered label will be sent to the application server.
11) In case the user has entered the correct label, the application server will send the requested content by the user, otherwise another IMPTCHA will be sent for user to retry.

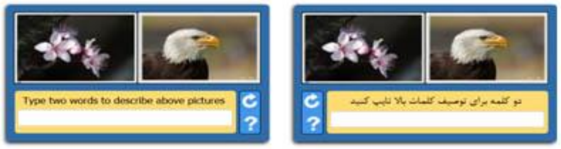

Type two words to describe above pictures
دو کلمه برای توصیف کلمات بالا تایپ کنید
?
?

Fig. 3: Sample YAPPTCHA in English and Farsi

3.1.2 Security

Regarding system structure as it is comprehensive from Fig. 3, IMPTCHA is highly secured against malicious attacks. There are three important factors in IMPTCHA's security:

- All the API, verification and database servers for IMPTCHA are separated.
- In case of a security breach, affected labeled images are recovered through the Shadow Database strategy mentioned before; hence, making bypassing the system extremely difficult even after a successful attack.

Using images instead of distorted text and the fact that currently recognizing an image's content via computer software is extremely difficult in

- contrast to using software such as OCR to bypass the system eliminates the possibility of using.

3.1.3 Labeling Images

Images are one of the useful information resources in web. Search engines with the ability to search image contents, would be tools which provide these information. One way to achieve this goal is labeling images. Labeling is in fact the process of recognizing all the objects in the images and explaining them in text. Currently there are a few techniques to label an image. One of the methods used today, is the one introduced by Luis Von Ahn in 2004 [11]. In this method, labels are acquired using a game named ESP in web. In the process of playing this game, users will label the images. Due to the fact that this game is not very popular and the users must opt in to play this game, this method is not very helpful to label a large amount of images found in the web. As described in the previous section, one of the two images presented to the user is an un-labeled image; hence, users will help labeling the images each time they solve a IMPTCHA. Each image is used for labeling in many times, so, every object in image will be labeled. For example, in image on Fig. 4(c), labels such as, sky, bird, eagle is provided by solving some IMPTCHA challenges.

3.1.4 Validation Process

As it is mentioned in section 2, in IMPTCHA, two images are shown to the user; one is labeled and the other will be labeled by the users. Current implementation is capable of summarizing the user-entered labels and if correct, finalizing their labels and adding them to the labeled images.

Suppose there are T images labeled by the users in the instance of analyzing. In this case, to evaluate a specific label such as l for an image such as i, we need a threshold acquired by the equation number 2. In addition, the total number of labels for all images, which is needed to calculate threshold is presented in equation 1:

$$C=\sum_{(i=0)}^{T}▒Ci \qquad (1)$$

$$\text{Threshold} = C/T \qquad (2)$$

In equation (1), the value for C is computed by adding the C_i which is total number of labels for image i, for first image through the Tth (last) image. Consequently, in equation (2), based on the total number of labels and total number of images, the value for threshold is computed.

At the end of each day, total number of labels for each image is calculated, and then based on this value, threshold is computed. If the frequency for each label is greater than the threshold for that image, this label will be finalized and its image will be added to the labeled images. For example, suppose 300 un-labeled images are in the database and at the end of the day there are 9270 labels for those images. If the image Fig. 4(c) was one of those images with four labels “anima”, “bird”, “eagle” and “sky” with 40, 35, 32 and 10 frequencies respectively, we would have:

$C_{Animal}=40, C_{Bird}=35, C_{Eagle}=32, C_{Sky}=10$

$\text{Threshold}=9270/300=30.9$

Therefore, since the frequency for three labels “animal”, “bird” and “eagle” is more than threshold, this labels will be finalized for this image.

3.1.5 Forbidden Word

Based on our user study, which its data is provided in section 4, people are more likely to enter general words for IMPTCHAs rather than entering specific words. For instance, the chance of a user entering the word “bird” is higher than him/her entering the word “eagle”. For our system to be able to capture more labels for a single picture, a method of using the “taboo word” concept is introduced. In this approach the user is forbidden by the system from entering the labels with frequencies more than a threshold specified by the system admin. This means if a picture has been labeled “bird” more than a number of times, the system automatically triggers a feature which asks the user not to enter “bird” in the related text box.

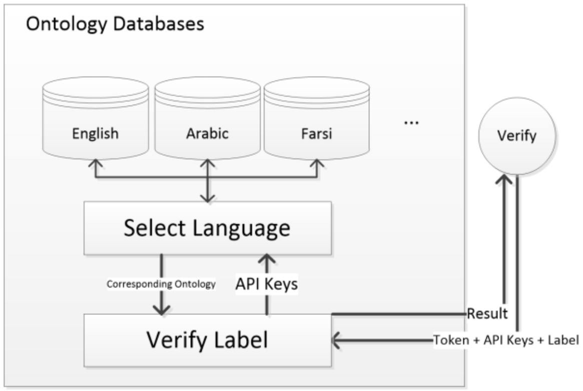


Fig.6: Multilanguage support in YAPTCHA

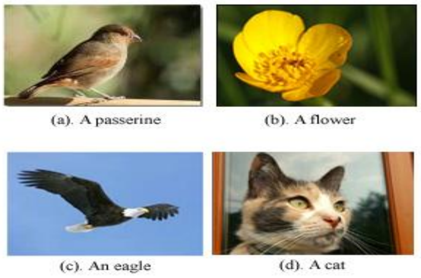

(a). A passerine
(b). A flower
(c). An eagle
(d). A cat

Fig. 4: Four example image in YAPPTCHA

The basic idea to achieve this, each time a IMPTCHA is being generated, the labels for the non-labeled image is processed and if one or more of them have the frequency greater than threshold, they will be printed in IMPTCHA form to notify the user of these forbidden words. However this will help a clever user to distinguish the labeled image from the non-labeled one and taking advantage of it by just providing the label for the labeled image which would hurt our system’s labeling value rather than improving it.

To overcome this problem, each labeled image in our database is classified by a field containing one or more categories. When a forbidden word is triggered for a non-labeled image, a labeled image matching the category for that word will be selected randomly by the system and sent to the user along with the forbidden word for both of images. This way, the user won't have a chance to identify the labeled image.

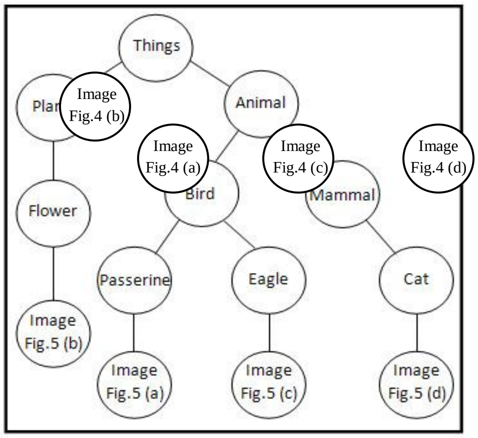


Fig 4: An example of semantic graph

3.2 Additional Features Of IMPTCHA

In addition to the given information, there are a few features presented in IMPTCHA which are not available in counterparts. These features will be covered in this section.

3.2.1 Ontology

One thing to note in IMPTCHA is that, in contrast to reCAPTCHA and many other CAPTCHA systems in which there is only one correct word for the distorted text, in IMPTCHA there are a few acceptable labels for each image. These labels should be analyzed using ontology and if they are in the ontology, they will be accepted. For instance, if an image of a passerine is sent to the user, if the user submits each of the words “passerine”, “bird” or “animal”, he/she is expected to successfully pass the CAPTCHA.

On the other hand, if the label does not exist in the existing ontology, but enough users enter it for a particular image, it will be added to the ontology. This will help the system to grow and improve its ontology database. This process is shown in Fig. 6.

Based on Fig. 6, when the Verify command is invoked in the system, the command sends the token for the image being processed along with the host's API keys and the label entered by the user to the verify label section. This section sends the API keys to the language selection section which analyzes the API keys and sends the corresponding ontology back. The verify label section then uses the token and the label to verify if the user has entered the correct word and if the word is not in the database, it will save it as a temporary word.

Additionally, the whole ontology database and/or specific domain-centric ontologies could be extracted for especial usages. For instance, the system can offer the exported ontology for nature, art or other domains to RDF files through web services.

The graph depicted in Fig. 5, shows a part of ontology for words used in Fig. 4.

3.2.2 Multilanguage Support

IMPTCHA's architecture is designed in such way that it can support any language such as English, French, Chinese, Arabic and Farsi. The language for the IMPTCHA is assigned to the public/private key pair and is selected during the API registration process. If the application owners want to support more than one language in their website, they should register for one public/private key pair for each of the languages they're planning to support.

To achieve the multi-language support, all the labels for labeled images are translated using translator software and stored in multiple tables in database. Also, the collected labels for each language are stored in the corresponding table in the database.

Another benefit of multi-language support is creating the ontology for that language. This means that first, verifying the ontology, described in section 3.2.1 will be done in the same manner and second, based on the ontology for one language it is possible to create/expand it for other languages.

4.Experimental Results

In a user study with 60 participants, we used and evaluated IMPTCHA. A web wizard accessible to all users, consisting of 20 pages was designed. It was developed in PHP language with a MySQL database consisting of 300 labeled images and 300 non-labeled images. In each page of the wizard, an instruction was presented to the user. In order to pass the step, the user had to solve the IMPTCHA challenge presented in that page.

In our user study, there were 3706 IMPTCHA challenges created in total with 1758 attempts to solve a challenge in which 955 of them were successful. Among the factors effective in the unsuccessful attempts, in addition to the incorrect recognition or intentional invalid input by the user, there were a few cases in which the image could not be easily recognized by the user. These include where the image contained too much detail in a large scale in contrast to more iconic images. After removing these images from the database and ignoring the results for them, 1077 (61.26%) of the attempts could be considered successful in our system which could satisfy the needs of a CAPTCHA system.

Additionally, the effect of time over the user's ability to solve the challenges was examined. Fig. 7, depicts the success and failure of the user attempts to solve the IMPTCHA challenges over time. From this figure it is comprehensive that this subject does not follow a specific pattern. This could be due to the previously mentioned factors.

In addition, to measure IMPTCHA's success rate in labeling images and the study results more, all the 955labels provided by the successful attempts on solving IMPTCHA were manually checked. There were 130 distinct labels for different images with various frequencies. From the 300 images, 110 had at least one label with a frequency more than the calculated threshold (3) and could be labeled. Only 2 of these labels were incorrect and one of them was an empty input. Considering the fact that the labeled and non-labeled images for IMPTCHA challenges are impossible to recognize, this could be only caused by luck. In any case, the success rate of the IMPTCHA system in this study is estimated to be 98.18% at worst.

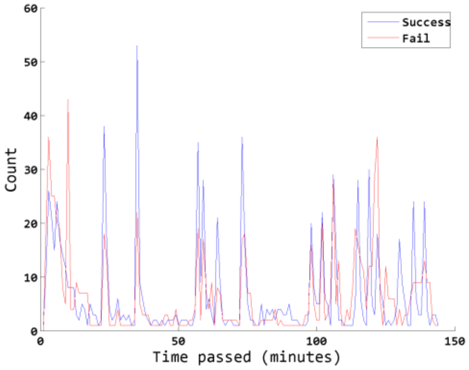


Fig.7: Success and failed resolve of challenge for different intervals of time

Additionally, the 130 distinct labels provided by the users, general terms such as "Bird", "Flower", "Sea", "Animal" and "Tree" have more frequency compared the other ones. This shows that the users tend to associate words with more general meanings to images because it makes it easier to solve the challenge.

Although it still helps the system in labeling unknown images, this could effectively limit the degree of the detail the system can provide. This was the main reason for creating the forbidden words list introduce in section 3.1.5.

5.Conclusion

Online privacy is becoming more crucial for us every day. Online banking, registration processes and in general all the identifying methods need a service to identify the humans from automated programs. CAPTCHAs are the most popular methods currently used by a variety of websites which makes this subject very important to focus. However, the current systems don’t provide a pleasant experience for the users while most of them suffer from security issues. On the other hand the only project with added benefit is the reCAPTCHA project.

The IMPTCHA system presented in this paper creates a pleasant and secure experience with added benefit for the provider system and eventually the end users. The numbers presented in section 4 report on both the system’s ability to effectively label images while providing a secure and practical CAPTCHA service.

Although this system is usable mostly from the scratch, it suffers from a basic problem which is its need to a startup database of labeled images. However, it is worth mentioning that this database will be growing overtime as the images are labeled by the users; hence, a small dataset would suffice.

Acknowledgement

The authors wish to thank the 60+university students in Bs., MSc. and PhD levels who helped in the testing process for the first IMPTCHA prototype.

References

[1] L. Von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, "reCaptcha: Human-based character recognition via web security measures," Science, vol. 321, pp. 1465-1468, 2008.

[2] ElieBursztein , Matthieu Martin , John Mitchell, Text-based CAPTCHA strengths and weaknesses, Proceedings of the 18th ACM conference on Computer and communications security, October 17-21, 2011, Chicago, Illinois, USA .

[3] Luis von Ahn, Manuel Blum, Nicholas J. Hopper, and John Langford. CAPTCHA: Using hard AI problems for security. In Eli Biham, editor, Advances in Cryptology – EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4-8, 2003, Proceedings, volume 2656 of Lecture Notes in Computer Science, pages 294–311. Springer, 2003.

[4]Turing Test, http://en.wikipedia.org/wiki/Turing_test, visited Date: 05/05/2019.

[5] What is reCAPTCHA?, http://www.google.com/recaptcha/learnmore,Visited Date: 10/05/2012.

[6] J. Elson, J. R. Doucerur, J. Howell, and J. Saul.Asirra: A Captcha that exploits interest-aligned manual image categorization. In Proceedings of the 14th ACM conference on Computer and communications security, CCS ’07, pages 366–374, New York, NY, USA, 2007. ACM.

[7] Nucaptcha,www.nucaptcha.com, visited date: 05/05/2012.

[8] Kluever, K.A., and Zanibbi, R. (2009) Balancing Usability and Security in a Video CAPTCHA. Paper presented at the 5th Symposium on Usable Privacy and Security. July 2009. Mountain View, California.

[9] How we broke the NuCaptcha video scheme and what we propose to fix it, http://elie.im/blog/security/how-we-broke-the-nucaptcha-video-scheme-and-what-we-propose-to-fix-it/, Visited Date: 25/04/2012.

[10] It’s 1999. Lowe, David G. (1999). "Object recognition from local scale-invariant features". Proceedings of the International Conference on Computer Vision. 2. pp. 1150–1157. DOI:10.1109/ICCV.1999.790410.

[11] Von Ahn, L.; Dabbish, L. (2004)."Labeling images with a computer game".Proceedings of the 2004 conference on Human factors in computing systems - CHI '04. pp. 319–326. DOI:10.1145/985692.985733. ISBN 1581137028.