Future Generation Computer Systems 31 (2014) 59–68

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

FaceDCAPTCHA: Face detection based color image CAPTCHA

Gaurav Goswami [a], Brian M. Powell [b], Mayank Vatsa [a,*], Richa Singh [a], Afzel Noore [b]

[a] Indraprastha Institute of Information Technology (IIIT) Delhi, India

[b] Lane Department of Computer Science and Electrical Engineering, West Virginia University, USA

ARTICLE INFO

Article history:
Received 1 February 2012
Received in revised form
29 July 2012
Accepted 27 August 2012
Available online 14 September 2012

Keywords:
CAPTCHA
Face detection
Web security

ABSTRACT

With data theft and computer break-ins becoming increasingly common, there is a great need for secondary authentication to reduce automated attacks while posing a minimal hindrance to legitimate users. CAPTCHA is one of the possible ways to classify human users and automated scripts. Though text-based CAPTCHAs are used in many applications, they pose a challenge due to language dependency. In this paper, we propose a face image-based CAPTCHA as a potential solution. To solve the CAPTCHA, users must correctly identify visually-distorted human faces embedded in a complex background without selecting any non-human faces. The proposed algorithm generates a CAPTCHA that offers better human accuracy and lower machine attack rates compared to existing approaches.

1. Introduction

Completely Automated Public Turing Test to Tell Computers and Humans Apart or CAPTCHA is designed to distinguish between genuine users and automated scripts [1]. The objective of CAPTCHA is to ensure proper service to genuine users while minimizing the attacks by bots. CAPTCHAs are being used for several services including web and financial services, and to provide security against malicious attacks. Research in CAPTCHA has focused on developing tests that are easy for humans to solve and difficult for automated approaches. Several kinds of challenges can be posed by automatic scripts. For instance, scripts or bots can put a heavy load on the servers and enforce a DoS attack, generate multiple fake accounts (in case of registration forms) which are not profitable to both the service provider and the client [2]. Existing CAPTCHA algorithms can be broadly grouped into three classes: (1) text-based, (2) image-based, and (3) video- and audio-based CAPTCHAs.

Text-based CAPTCHAs are the most common and widely used form. These CAPTCHAs require the users to decipher text that has been visually distorted and rendered as an image. AltaVista CAPTCHA, one of the first text CAPTCHAs, was taken from an optical character recognition (OCR) manual. Distortions were incorporated that were known to reduce OCR accuracy [3]. GIMPY CAPTCHA, similar to the AltaVista CAPTCHA [3,4], used English dictionary words. However, Mori and Malik showed that it can be broken and an attack rate of 92% was achieved against EZ-GIMPY [5], a variant of GIMPY. Further variation by Moy et al. [6] boosted the attack rate to 99%. A major shortcoming of these early approaches was vulnerability to segmentation, where each character could be identified in isolation. This greatly simplifies attacks using optical character recognition techniques. One solution was proposed to design the CAPTCHA such that one-to-one mapping between characters and outlines was distorted. For example, two characters might be connected or one might be split into multiple parts. In the ScatterType CAPTCHA, for example, individual characters were segmented into pieces and then systematically scattered so that they are difficult to reassemble [7]. Megaupload CAPTCHA proposed to use overlapping characters whereas MSN CAPTCHA introduced lines connecting individual characters; however, both have high attack rates of 78% or more [3,8–10]. BaffleText's approach of rendering a mottled black-and-white background and then performing different masking operations with overlapping text was more successful, being attacked in only 25% of the attempts [11]. Different masking techniques similar to BaffleText have subsequently been incorporated into other CAPTCHAs [12].

Rather than designing tests to be non-recognizable via OCR, some CAPTCHAs have taken an approach of using handwritten text images already known to fail optical character recognition. A database of text images obtained from handwritten mail addresses that could not be detected automatically were used in such CAPTCHAs. When full city names were used, humans were able to identify the word 100% of the time but the computer success rate was about 9% [13]. Similarly, reCAPTCHA was designed using the text images scanned from book digitization projects [12]. In reCAPTCHA, users were presented with two text images (one of a word that was unknown and one whose text had been previously

* Corresponding author.
E-mail addresses: gauravgs@iiitd.ac.in (G. Goswami), brian.powell@mail.wvu.edu (B.M. Powell), mayank@iiitd.ac.in (M. Vatsa), rsingh@iiitd.ac.in (R. Singh), afzel.noore@mail.wvu.edu (A. Noore).

0167-739X/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2012.08.013

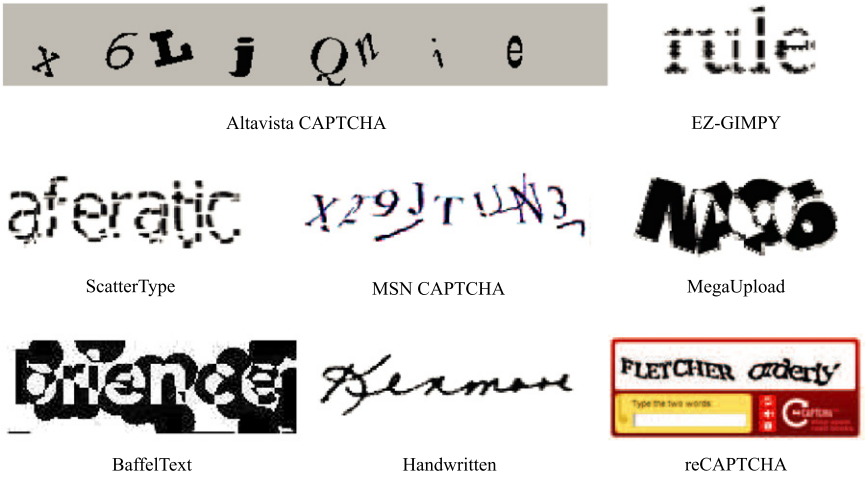

Fig. 1. Example of existing text CAPTCHAs.

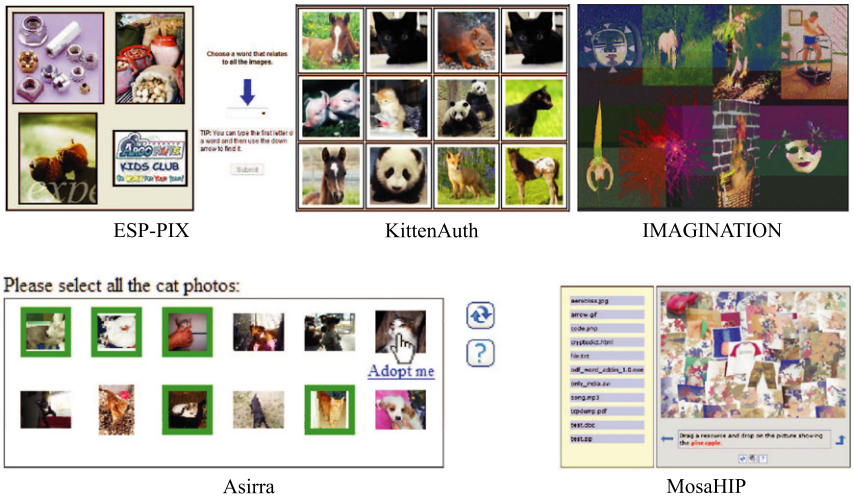

Fig. 2. Example of existing image CAPTCHAs.

determined) and asked to enter both words. The previously-known word served as the test while the currently-unknown word's results were stored to help identify that word for future use. Researchers have shown that the success attack rate for reCAPTCHA is between 5% and 30% [14]. Examples of existing text CAPTCHAs are shown in Fig. 1.

As an alternative to text, several CAPTCHA applications utilize image classification or recognition tasks as part of their test [15]. One basic image-based CAPTCHA is ESP-PIX in which a collection of images are shown and the user has to select a description from a predefined list of categories [16,17]. KittenAuth, a variant of image CAPTCHA, poses images of cats to the user [18]. Asirra is similar to KittenAuth and uses a closed database to source the images [19]. These image-based CAPTCHAs demonstrate a common weakness—a small number of possible solutions for which random guessing can have a high likelihood of success. A number of other CAPTCHAs rely upon composites of multiple embedded images rather than discrete images as with the previous models. The Scene Tagging CAPTCHA requires identifying relationships and relative placement of different images [20]. On the other hand, MosaHIP requires dragging descriptors and dropping them on top of embedded images in a collage [21]. Recently, a new design technique has been proposed that uses recognition of geometric patterns. The IMAGINATION CAPTCHA combines geometric shape recognition with categorization in a two-step process. Users have to first mark the center point of an embedded image and then select an appropriate category based on a predefined list to describe that image [22]. The results show a human success rate of approximately 70% with a machine random guess rate of about 0.0005% [22]. Fig. 2 shows a sample of existing image CAPTCHAs.

Other than text and image CAPTCHAs, video and audio CAPTCHAs have also been proposed. Video-based CAPTCHAs function by posing the tagged videos with descriptive text. In the tests by Kluever, humans achieved an accuracy of 90% in identifying video descriptions while machine attack rates were approximately 13% [3,23]. To provide access for visually-impaired users, audio CAPTCHAs are used as an alternative to standard visual CAPTCHAs. These work by playing a recording of words or letters which users are then asked to enter. However, these CAPTCHAs have high computer attack rates using a speech recognition approach [24–26]. Specifically, the audio CAPTCHAs used by Digg and Google have a successful attack rate of about 70% [24].

1.1. Research contributions

Making CAPTCHAs resilient to attacks by advanced scripts increases the complexity of the tests and language dependency [15]. In some cases, the difficulty has reached levels that are hard even for humans to solve. Since image CAPTCHAs provide language independence and improved user convenience compared to traditional