A CAPTCHA DESIGN BASED ON VISUAL REASONING

Haipeng Wang, Feng Zheng, Zhuoming Chen, Yi Lu, Jing Gao and Renjia Wei

Tencent Inc.
{jayjaywang, delanfzheng, kraigchen, shisilu, jessicagao, jacobwei}@tencent.com

ABSTRACT

CAPTCHA is a reverse Turing test to distinguish humans from machines. It is widely used in the internet industry for cyber security. A good CAPTCHA is supposed to be easy for humans but difficult for machines. Many existing CAPTCHA implementations leverage the inability of automatic visual recognition, e.g., recognizing the text or other objects in an image. These CAPTCHAs are becoming more and more vulnerable recently, due to the rapid development of visual recognition techniques. This paper presents our study of using visual reasoning in CAPTCHA design. This CAPTCHA asks the users to find specific object(s) in an image according to a given text query. It is generally easy for humans to understand the text query and make sophisticated reasoning about the image, but still remains difficult and computationally expensive for machines. We describe the CAPTCHA design, provide usability analysis and present security experiments. Moreover, we show that the security can be further improved by the use of neural style transfer.

Index Terms— CAPTCHA, visual reasoning, neural style transfer, cyber security

1. INTRODUCTION

Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA), also known as Human Interaction Proofs (HIP), is a standard approach to determine whether a user is human or not. It is used to protect websites from malicious behaviors, such as spamming, account enumeration, auto post, etc. A typical CAPTCHA asks users to complete a simple task, which is supposed to be easy for humans but difficult for machines. Since the notion of CAPTCHA was introduced in 2000 [1], it has drawn significant attentions and received various designs from both academic and industrial communities [2, 3, 4, 5, 6].

In order to withstand automatic attacks, one important principle in CAPTCHA design is to use difficult artificial intelligence (AI) problems [2]. To date, the most widely used ones are based on *visual recognition*. These CAPTCHAs present users an image and ask users to recognize the text or other objects (e.g., animals [5], human faces [7, 8], street signs[6], etc) in the image. Automatic visual recognition was a hard AI problem, but now has become much easier due to the rapid development of image recognition techniques. Actually there have been many successful research contributions on how to automatically solve these CAPTCHAs [9, 10, 11, 12, 13]. Therefore there is a clear need to develop more secure CAPTCHAs.

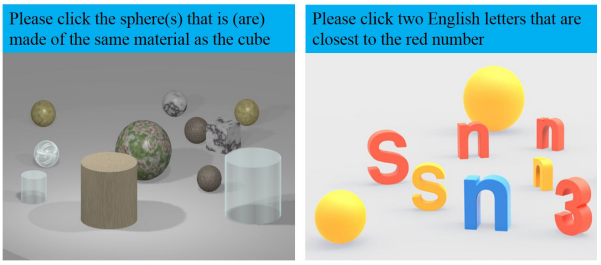

Fig. 1. Illustrations of our CAPTCHA design based on visual reasoning. Each CAPTCHA contains a text query and an image.

In this paper, we present a study of using visual reasoning in CAPTCHA design. Illustrations are shown in Figure 1. This CAPTCHA provides users a text query and an image, and asks users to locate specific objects in the image. To successfully pass this CAPTCHA, users need to understand the text query and make reasoning about the objects in the image. This design is originated from the CLEVR dataset [14]. The CAPTCHA images depict 3D shapes with simple objects (e.g., geometric shapes, numbers, English letters, etc), which are easy for users to understand. The query set involves many aspects of reasoning, such as object identification, comparison, spatial relationships, the use of commonsense knowledge, etc. We have prepared a set of text queries that are suitable for CAPTCHA, and are keeping extending the query set. Moreover, to further enhance security, we have tried to use neural style transfer [15] in the image generation. Neural style transfer can greatly increase the variation of image styles in an automatic way.

2. PRIOR ARTS

In the literature, there have been many contributions to CAPTCHAs. This paper focuses on visual-based CAPTCHAs, among which the two most commonly used types are text-

based CAPTCHAs and image-based CAPTCHAs.

Fig. 2. Examples of three text-based CAPTCHAs (Fig.(a)-(c)) and one image-based CAPTCHA (Fig.(d)).

Text-based CAPTCHAs ask users to decipher the text (e.g., numbers, English letter, etc) in an image. They are easy to implement and their file sizes are small. To defend against automatic solvers, recent text-based CAPTCHAs have focused on generating various distortions [16, 17], such as text twisting, negative kerning, changing background images, adding occluding lines, etc. Examples are shown in Figure 2(a)-(c). Even with distortions, many of these CAPTCHAs have already been solved by text recognition techniques. For example, a generic solving approach to text-based CAPTCHA was proposed in [18]. In [19], the EZ-Gimpy was solved with an accuracy of 83%. In [13], the authors reported an attacking accuracy of 44.6% on Microsoft's two-layer CAPTCHA, which was deployed in 2015. In [20], an end-to-end deep learning approach achieved an accuracy of 99.8% on the text puzzles of Google's reCAPTCHA.

Image-based CAPTCHAs usually rely on the difficulty of image recognition. An example is shown in Figure 2(d). The earliest example is ESP-PIX [1]. It asks users to recognize what object is common in a set of images. Following that, many variations have been proposed. For example, Asirra [5] asks users to select cats from 12 images of cats and dogs. In [3], the authors proposed several designs, such as naming images, finding the anomaly image, etc. In [8, 21], the use of face recognition was studied. With the development of image recognition techniques, these CAPTCHAs are becoming less secure nowadays. For example, the Asirra CAPTCHA has been extensively studied [11, 22], and was reported to be automatically recognized with a successful rate of 42% [23] in 2012. In [24], the authors used deep learning to achieve an accuracy of 70.78% on the image puzzles of Google's reCAPTCHA, and an accuracy of 83.5% on Facebook's image CAPTCHA. Actually according to the recent Imagenet evaluation results [25], image classification techniques can achieve human-like accuracy provided enough training data.

Apart from these two types of CAPTCHAs, other designs have also been proposed. For example, in [26], game CAPTCHAs were used. Video CAPTCHAs were studied in [27, 28]. Compared to text-based or image-based CAPTCHAs, these CAPTCHAs are more difficult to be generated, and typically have larger file sizes, which limits their use in practice. In [29, 30], audio CAPTCHA was studied. It is particularly suitable for those visually impaired users.

3. CAPTCHA DESIGN

We use visual reasoning in the CAPTCHA design. See Figure 1 for an example. Visual reasoning has drawn significant research attention in recent years. Although humans can easily make simple reasoning about objects in an image, it turns out to be much more challenging for machines. One recent benchmark dataset in this area is CLEVR [14], which aims for diagnostic analysis of visual reasoning. Our CAPTCHA design actually follows a very similar fashion as CLEVR, with a few differences in the images and text queries. CLEVR contains three objects (cube, sphere, and cylinder), while our CAPTCHA uses more objects (e.g, letters, numbers, etc) with more attribute variations. CLEVR includes nearly one million text questions. Most of them are not suitable for CAPTCHAs. For example, questions like "*yes or no*", "*how many [color/material] things are there*", or "*what shape/material is it*", are not secure against random guesses. So we have designed a set of text queries particularly for the CAPTCHAs.

3.1. image generation

Synthetic 3D objects are used in the image generation. We have used many different types of objects, such as geometric shapes (cone, cylinder, cube, sphere, etc), English letters (lower-case and upper-case), numbers, Chinese characters and many other shapes (e.g., fruits, vehicles, etc). Some shapes may have notches. Examples of these objects are illustrated in Figure 3. These objects come in various attributes, such as different colors (blue, green, yellow and red) and different materials (marble, wood, glass and iron).

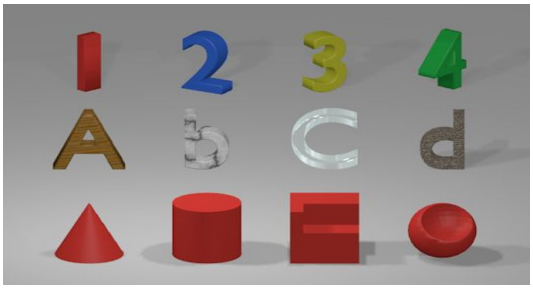

Fig. 3. Examples of 3D objects used in CAPTCHA images. The first line shows numbers with different colors. The second line shows English letters made of different materials. The third line shows geometric shapes with/without notches. Many other objects are also used, such as Chinese characters, fruits, vehicles, etc.

To generate an image, we randomly select between 5 and 15 objects with different attributes and put them in a scene. The objects may also vary in sizes and inclination angles. Intersections between objects are kept small for not affecting humans' performances. Lights and cameras are randomly placed to render the scene and create 3D effects, so the objects get clear 3D spatial relationships, such as "*front*", "*behind*", "*left*", "*right*", etc.

The image generation can be extended by using more objects with more attribute variations. We are keeping working on the extension. Although more objects and attributes would make CAPTCHAs more secure against automatic attacks, we are careful about these extensions and make sure that they do not harm user experiences.

3.2. text query generation

A text query describes a simple task for users to complete. Users need to understand the query and make proper reasoning. We have designed a set of query templates. Instantiations of these templates can generate full query sentences. For example, the template "*please click the [color] [object_name_1] that is (are) left of the [object_name_2]*" has three parameters "*[color]*", "*[object_name_1]*" and "*[object_name_2]*". An instantiations of assigning "*red*" to "*[color]*", "*cube(s)*" to "*[object_name_1]*" and "*capital letter L*" to "*[object_name_2]*", can form a query "*please click the red cube(s) that is(are) left of the capital letter L*". In practice we need to reject those instantiations that do not match the image information. For examples, a query "*please click all the numbers*" does not match images that do not contain numbers.

All our queries are all manually checked to ensure that they are simple for humans to understand. Each query may involve one or more aspects of visual reasoning, such as object identification, comparison, spatial relationship, etc. To increase language diversity, a query template may have an alternative version. For example, an alternative to the template "*please click the [color] [object_name]*" may be "*please click the [object_name] that is (are) [color]*". Table 1 lists a few examples of the queries.

Extensions to the query set involve two aspects. One is to design new query templates with new reasoning tasks. The other is to write more synonymous templates to the existing ones. We will keep extending the query set regularly. In the query design, we should be very careful about the difficulty of queries. Complex queries would enhance the security, but may lead to negative user experiences.

3.3. image style transfer

In addition to the variations of image objects and text queries, image style variation is also desirable for robustness against automatic attacks. Previous design of image styles requires a lot of manual efforts. Fortunately the recent success of neural style transfer provides an automatic way to change styles of existing images. Neural style transfer is based on the finding that the content and style of an image can be largely separated using a convolutional neural network [15]. For a CAPTCHA image, the content mainly refers to the objects, and the style mainly refers to general appearances in terms of lights, background color distributions, etc. Given an existing CAPTCHA image, we can mix its content with the style of another image (such as a painting), and thereby generate a new CAPTCHA image. There have been several research contributions to neural style transfer [31, 32]. In this work, we adopted the approach proposed in [33] for its high flexibility and efficiency. Examples of stylized images are shown in Figure 4. Note that after neural style transfer, some object attributes (e.g., colors, materials, etc) may be changed, so the text queries related to these attributes should be avoided.

Query types	Examples
object identification	*Please click the red number(s)*
attribute comparison	*Please click the sphere(s) made of the same material as the letter A*
spatial relationships	*Please click two English letters that are closest to the blue cube*
anomaly	*Please click the object that is different from others*
commonsense knowledge	*Please click two numbers that add up to 10*

Table 1. Illustrations of the text queries. One example for each query type. Different types may require different visual reasoning skills.

(a) (b)

Fig. 4. Examples of stylized images. For each image, the corresponding style image is on the left up corner. Original CAPTCHA images are shown in Figure 1.

4. USABILITY ANALYSIS

We carried out two experiments to quantitatively evaluate the usability of the CAPTCHA. The first experiment was a *controlled* one involving 78 participants whose ages ranged from 16 to 50. Each participant was asked to complete at least 10 CAPTCHAs. The users response time and successful rates were recorded. The same participants also completed a similar experiment on the widely-used Tencent Slider CAPTCHA [36], which asked users to click a button and slide it to a particular position. We then deployed the CAPTCHA online and

CAPTCHAs	Type	Experimental Setting	Response time (seconds)	Successful rate (%)
The proposed CAPTCHA	visual reasoning	controlled	6.3	73.3
The proposed CAPTCHA	visual reasoning	online	7.5	65.3
Asirra [5]	image recognition	controlled	15.0	83.4
Asirra [5]	image recognition	online	-	66.0
Avatar [8]	image recognition	controlled	21.3	62.0
Cortcha [34]	image recognition	controlled	18.3	86.2
Artifacial [35]	image recognition	controlled	14.0	99.7
Google Text CAPTCHA [34]	text recognition	controlled	7.9	82.8
A Video CAPTCHA [27]	video labelling	controlled	17.0	$77.0 \sim 90.0$
Tencent Slider CAPTCHA [36]	slider	controlled	7.1	82.9

Table 2. Users' response time and successful rates. There are two experimental settings. One is *controlled*, which invites only a few users to conduct experiments in a lab environment. The other is *online*, which means the CAPTCHA was deployed online and the data was from many online users' real behaviors. The response time and successful rates of *the proposed CAPTCHA* and *Tencent's Slider CAPTCHA* are either from a controlled test involving 78 participants or from more than 50,000 online records. All other numbers are collected from the literature. Details can be found in the corresponding references. For the video CAPTCHA, its authors tried several configurations and got different successful rates ranging from 77.0% to 90.0% [27].

received more than 50,000 records quickly. Note that the stylized images were not used in the online setting.

Table 2 lists the response time and successful rates of different CAPTCHAs. It can be observed that the response time of the proposed CAPTCHA was close to those of Google Text CAPTCHA and Tencent Slider CAPTCHA, and was smaller than many of the image recognition CAPTCHAs. This may be due to the fact that these image recognition CAPTCHAs require users to deal with a number of images, but our CAPTCHA, like Google Text CAPTCHA and Tencent Slider CAPTCHA, presents only one image to the users.

The successful rate of our CAPTCHA was higher than Avatar but lower than others in the controlled setting. According to the feedback, some participants were too hurry to make decisions without careful reasoning. Actually some of them realized the correct answer immediately after they clicked the wrong position. Besides, in the first few attempts, some participants would like to give random guesses and see the response of our CAPTCHA system. We believe that online users have similar experiences when trying this new CAPTCHA. Actually, in the real online setting, our CAPTCHA got similar successful rates to Asirra. In practice, users are allowed to try more than once, and we can expect that more than 95% users can succeed within three trials.

5. SECURITY EXPERIMENTS

The simplest automatic attack is random guess, which should be useless for our CAPTCHA. We then carried out a preliminary attacking experiment using visual reasoning techniques. We adopted an approach similar to the relation network (RN) [37]. In the RN architecture, an image was parsed by a CNN, and a text query was dealt with an LSTM. The CNN outputs were transformed by a composite function, and then padded with the LSTM final states. MLPs were then applied to give the final outputs. Please refer to [37] for details. Note that we did not use word embedding for the queries, but simply used the one-hot representations. The final MLP output corresponds to a 100 dimensional vector with each element corresponding to a small block range in the image. We have tuned different configurations, such as the number of CNN layers, the MLP sizes, etc. The best accuracy we could get was 4.7%, much lower than the attacking accuracies on text-based or image-based CAPTCHAs [13, 19, 20, 23, 24]. Increasing the training set could lead to better accuracy, but also come with more human labeling efforts for attackers. Note that this experiment did not involve color or material questions. This is a preliminary attacking experiment. We would like to investigate better attacking approaches in the future.

We then applied neural style transfer to the test images and got a stylized test set. Using the previous model, the accuracy on the stylized test set was smaller than 1%. This validates the effectiveness of using neural style transfer for CAPTCHAs. Note that if the previous model was finetuned with a few stylized training samples, its performance could quickly rise to the same level as on the original test images.

6. CONCLUSION

This paper reported our investigation of using visual reasoning in the CAPTCHA design. This CAPTCHA askes users to locate particular object(s) in an image according to the requirement of a text query. We use synthetic images with 3D shapes (e.g., geometric shapes, numbers, English letters, etc) for image generation. The text queries involve many aspects of visual reasoning, such as object identification, comparison, spatial relationships, etc. We have carried out both controlled and online experiments to ensure the usability of this design. A simulated attacking experiment was also conducted to verify the CAPTCHA security. In addition, we have tried neural style transfer to generate new images, and its effectiveness was also confirmed in the experiment. To our best knowledge, this is the first report on using visual reasoning and neural style transfer in CAPTCHA design.

7. REFERENCES

[1] L. Von Ahn, M. Blum, N. J. Hopper, and J. Langford, "Captcha: http://www.captcha.net," 2000.

[2] L. Von Ahn, M. Blum, N. J. Hopper, and J. Langford, "Captcha: Using hard ai problems for security," in *International Conference on the Theory and Applications of Cryptographic Techniques*, 2003.

[3] M. Chew and J. D. Tygar, "Image recognition CAPTCHAs," in *International Conference on Information Security*, 2004.

[4] L. Von Ahn, M. Blum, and J. Langford, "Telling humans and computers apart automatically," *Communications of the ACM*, vol. 47, no. 2, pp. 56–60, 2004.

[5] J. Elson, J. R. Douceur, J. Howell, and J. Saul, "Asirra: a captcha that exploits interest-aligned manual image categorization.," in *ACM Conference on Computer and Communications Security*, 2007.

[6] "Google recaptcha: https://www.google.com/recaptcha," .

[7] D. Misra and K. Gaj, "Face recognition captchas," in *AICT-ICIW*, 2006.

[8] D. D'Souza, P. C. Polina, and R. V. Yampolskiy, "Avatar captcha: Telling computers and humans apart via face classification," in *IEEE Electro Information Technology*, 2012.

[9] K. Chellapilla and P. Simard, "Using machine learning to break visual human interaction proofs (hips)," in *Advances in neural information processing systems*, 2005.

[10] J. Yan and A. S. El Ahmad, "Breaking visual captchas with naive pattern recognition algorithms," in *IEEE Computer Security Applications Conference*, 2007.

[11] P. Golle, "Machine learning attacks against the asirra captcha," in *Proceedings of the 15th ACM conference on Computer and communications security*, 2008.

[12] J. Yan and Ahmad S. El A., "A low-cost attack on a microsoft captcha," in *Proceedings of the 15th ACM conference on computer and communications security*, 2008, pp. 543–554.

[13] H. Gao, M. Tang, Y. Liu, P. Zhang, and X. Liu, "Research on the security of Microsofts two-layer captcha," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1671–1685, 2017.

[14] J. Johnson, B. Hariharan, L. van der Maaten, F. Li, C. L. Zitnick, and R. Girshick, "CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning," *arXiv preprint arXiv:1612.06890*, 2016.

[15] L. Gatys, A. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[16] J. Yan and A. S. El Ahmad, "Usability of captchas or usability issues in captcha design," in *Proceedings of the 4th symposium on Usable privacy and security*, 2008, pp. 44–52.

[17] E. Bursztein, M. Martin, and J. Mitchell, "Text-based captcha strengths and weaknesses," in *Proceedings of the ACM conference on Computer and communications security*, 2011.

[18] E. Bursztein, J. Aigrain, A. Moscicki, and J. Mitchell, "The end is nigh: Generic solving of text-based captchas.," in *WOOT*, 2014.

[19] G. Mori and J. Malik, "Recognizing objects in adversarial clutter: Breaking a visual captcha," in *Computer Vision and Pattern Recognition*, 2003.

[20] I. Goodfellow, Y. Bulatov, J. Ibarz, et al., "Multi-digit number recognition from street view imagery using deep convolutional neural networks," *arXiv preprint arXiv:1312.6082*, 2013.

[21] G. Goswami, B. M. Powell, M. Vatsa, R. Singh, and A. Noore, "Fr-captcha: Captcha based on recognizing human faces," *PloS one*, 2014.

[22] C. Karthik and R. A. Recasens, "Breaking microsofts captcha," Tech. Rep., Tech. Rep, 2015.

[23] O. Parkhi, A. Vedaldi, et al., "Cats and dogs," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[24] S. Sivakorn, I. Polakis, and A. D Keromytis, "I am robot:(deep) learning to break semantic image captchas," in *IEEE European Symposium on Security and Privacy (EuroS&P)*, 2016.

[25] O. Russakovsky, J. Deng, H. Su, , et al., "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, 2015.

[26] H. Yu and M. Riedl, "Automatic generation of game-base captchas," in *Procedural Content Generation*, 2015.

[27] K. A. Kluever and R. Zanibbi, "Balancing usability and security in a video captcha," in *the 5th Symposium on Usable Privacy and Security*, 2009.

[28] Y. Xu, G. Reynaga, S. Chiasson, J. Frahm, F. Monrose, and P. van Oorschot, "Security and usability challenges of moving-object captchas: Decoding codewords in motion.," in *USENIX security symposium*, 2012.

[29] J. Tam, J. Simsa, S. Hyde, and L. Ahn, "Breaking audio captchas," in *Advances in Neural Information Processing Systems*, 2009, pp. 1625–1632.

[30] Y. Soupionis and D. Gritzalis, "Audio captcha: Existing solutions assessment and a new implementation for voip telephony," *Computers & Security*, vol. 29, no. 5, pp. 603–618, 2010.

[31] L. Gatys, A. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[32] J. Johnson, A. Alahi, and F. Li, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*, 2016.

[33] T. Chen and M. Schmidt, "Fast patch-based style transfer of arbitrary style," *arXiv preprint arXiv:1612.04337*, 2016.

[34] B. Zhu, J. Yan, Q. Li, et al., "Attacks and design of image recognition captchas," in *Proceedings of the ACM conference on Computer and communications security*, 2010.

[35] Y. Rui and Z. Liu, "Artifacial: Automated reverse turing test using facial features," *Multimedia Systems*, 2004.

[36] "Tencent slider captcha: http://open.captcha.qq.com/cap_web/experience-slidejigsaw.html," .

[37] Adam Santoro, David Raposo, David GT Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap, "A simple neural network module for relational reasoning," *arXiv preprint arXiv:1706.01427*, 2017.