

CPSC 8570 - Semester Project Proposal Report

Mitali Bhosekar
Clemson University

Pranita Camasamudram
Clemson University

Yashwanth Porreddy
Clemson University

1 Group

Our group is comprised of three graduated students: Pranita Camasamudram (**Project Leader**), Mitali Bhosekar, and Yashwanth Poreddy.

2 Title

DEEP CAPTCHAs: An Accuracy Assessment

3 Introduction

CAPTCHAs are assessment tools used to tell humans and spam bots apart. They stand for Completely Automated Public Turing test to Tell Computers and Humans Apart. These assessment tools use something known as "Challenge response authentication" where we need to go through solving a few simple but effective challenges like selecting pictures or retyping a given pattern. Such small challenges are difficult for bots to complete as they detect even small gestures like hesitation that humans show. One of the most popular versions of CAPTCHA is where the users are given distorted letters in multiple fonts and asked to identify them.

The idea behind this is that bots can identify only what they are trained in whereas humans can recognise multiple fonts/handwriting. These days advanced bots are being trained by AI and machine learning and are able to identify these distorted characters. reCAPTCHAs got developed as an answer to this problem where users need to identify a few images of certain things asked. For our project we want to concentrate on Deep CAPTCHAs which are developed using Convolutional Neural Networks. A more robust kind of CAPTCHA, it was proposed as a solution to the vulnerabilities faced by the previous CAPTCHAs.





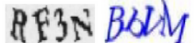

Protection mechanism		Example
Anti-segmentation	Hollow symbols	
	crowding characters together (CCT)	
	Background noise	
	Two-level structure	
Anti-recognition	Different fonts, Rotations, Wave-like symbols	
	Different languages	

Figure 1: Different types of Captchas.[3]

4 Related Works

This section briefly talks about some of the related work done in this field.

In recent years, a lot of research has been done to develop various techniques for creating robust CAPTCHAs. Numerous tools have been built using Machine Learning models trained on hundreds of CAPTCHA datasets and can crack the CAPTCHAs in no time.

The authors [7] developed a Convolutional Neural Network model called Deep-CAPTCHA. They attempted to use an automated deep learning-based approach to crack the visual CAPTCHA test. The purpose of this study is to look at the vulnerabilities and weaknesses of CAPTCHA generator systems in order to design more robust CAPTCHAs without having to rely on human try-and-fail methods.

To crack character-based image CAPTCHAs, Geetika Garg and Chris Pollett [2] used a trained Python-based

deep neural network. Instead of the traditional approach of CAPTCHA cracking based on segmenting and detecting individual letters, they deployed Convolutional Neural Network layers followed by a dense layer and a recurrent neural network layer. On this problem, using that dense layer proved to be more effective.

The adversarial CAPTCHA generating algorithm is another exciting related research area. The authors [5] introduced adversarial noise (Immutable adversarial noise) to an original image to cause basic image classifiers to misclassify images, yet the image appears to humans the same.

Another technique employs a Generative Adversarial Network [6], which takes far fewer real CAPTCHAs to train and produces the best performance compared to many machine learning models, which require a large number of manually labeled real CAPTCHAs train. This approach can bypass the advanced security features that many recent text CAPTCHA methods use.

Another study performs a survey [1] and examines many current CAPTCHAs, concluding that models based on deep learning can outperform standard techniques. According to the report, CNN is the most often used model. In contrast, other Deep Learning models such as DBN (Deep Belief Networks), RNN, DRL (Deep Reinforcement Learning), and LSTM (Long Short Term Memory) have a lot of opportunities for improvement.

Another research improved performance by using Convolutional Neural Networks and the style Transfer method. When CAPTCHA delivers a lot of distortion to have resistance against the automated attack, it's tough to detect human perception. They proposed [4] to use the style transfer method to deceive the machine while retaining the human perception rate in this paper. This approach produces a style-plugged-CAPTCHA image by combining the styles of many images while keeping the original CAPTCHA sample's content.

5 Problem Statement

With the increase in number of CAPTCHAs, it becomes imperative to measure the accuracy's of CAPTCHA algorithms. In our project we plan to focus on visual CAPTCHA, with alphanumeric values available as open source. We aim to measure the accuracy, of these CAPTCHAs, i.e. if they are able to identify a breach. This project will help developers to identify which open source code to implement in their project.

The deep CAPTCHA uses convolutional neural networks and is mainly used to crack visual CAPTCHAs. Hence we aim to use this algorithm to run on different open source CAPTCHA algorithms available on GitHub repository. As stated in the base research paper, the deep-CAPTCHA has a crack accuracy of 98.94% and 98.31%

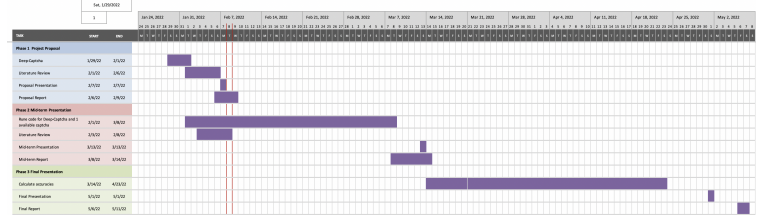


Figure 2: Expected Timeline

for the numerical and the alpha-numerical test datasets, respectively. [7] As the accuracy of the selected method is high as claimed [7], we are definite that we can provide an highly accurate representation of the open source codes.

6 Projected Timeline

In the phase 2 of the project, we plan to execute the code for deep-CAPTCHA as provided by [7]. Once we understand in depth its functioning we plan to check the accuracy of 4-5 alphanumeric CAPTCHAs on GitHub as a part of phase 2. We plan to equally contribute in the phase 2 as it is directly dependent on phase 3 execution. For phase 3 we will each take 1-2 algorithms to execute using the deep-CAPTCHA code.

References

- [1] CHEN, JUN LUO, X. . G. Y. . Z. Y. . G. D. "a survey on breaking technique of text-based captcha". 2017. 1-15. 10.1155/2017/6898617.
- [2] GARG, G., AND POLLETT, C. "neural network captcha crackers.". In *2016 Future Technologies Conference (FTC)*, pp. 853-861. IEEE, (2016).
- [3] GERASIMCHUK, A. How some algorithms generate captcha, while others crack it.
- [4] HYUN KWON, H. Y., AND PARK., K.-W. "captcha image generation using style transfer learning in deep neural network.". *n Information Security Applications: 20th International Conference, WISA 2019, Jeju Island, South Korea, August 21–24, 2019, Revised Selected Papers Springer-Verlag, Berlin, Heidelberg, 234–246. DOI:https://doi.org/10.1007/978-3-030-39303-8_18*.
- [5] OSADCHY, MARGARITA, J. H.-C. S. G. O. D., AND PREZ-CABO, D. "no bot expects the deepcaptcha! introducing immutable adversarial examples, with applications to captcha generation.". *IEEE Transactions on Information Forensics and Security* 12, no. 11 (2017): 2640-2653..
- [6] YE, GUOXIN, Z. T.-D. F. Z. Z. Y. F. P. X. X. C., AND WANG, Z. "yet another text captcha solver: A generative adversarial network based approach.". In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pp. 332-348. 2018.
- [7] ZAHRA NOURY *, M. R. Deep-captcha: a deep learning based captcha solver for vulnerability assessment. <https://ssrn.com/abstract=3633354> or <http://dx.doi.org/10.2139/ssrn.3633354> (2020).