Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

Implementation of Captcha Mechanisms using Deep Learning to Prevent Automated Bot Attacks

Received: 08 September 2023; Revised: 05 November 2023; Accepted: 15 December 2023

Prof. (Dr.) Sachin R. Sakhare
Department of Computer Engineering,
Vishwakarma Institute of Information Technology, Pune - India
sachin.sakhare@viit.ac.in
https://orcid.org/0000-0003-1974-5929

Mr. Vivek D. Patil
Department of Artificial Intelligence & Data Science,
Vishwakarma Institute of Information Technology, Pune - INDIA
vivek.patil@viit.ac.in

Abstract

Online platforms' integrity and security are seriously threatened by the growth of automated bot attacks, necessitating the development of effective methods for telling harmful bots apart from legitimate users. In order to successfully combat automated bot attacks, this project investigates the creation and application of CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) methods utilising deep learning techniques. In this study, we employ deep learning to create and apply complex CAPTCHA-problems that help us distinguish between real users and automated bots. Convolutional neural networks and recurrent neural networks are employed to build dynamic, adjustable CAPTCHAs in order to keep up with the bots' evolving approaches. Our research focuses on creating CAPTCHAs that are simple to use for humans but difficult to understand by robots, which creates a favourable user experience for those who are actually human. Character skewing, background noise injection, and image obfuscation are some of the methods we use to safeguard our CAPTCHAs while maintaining their usability. Furthermore, we carry out exhaustive trials in the real world to assess the effectiveness of our methods based on in-depth CAPTCHA learning. We evaluate their resistance to a variety of attack methods, such as counterattacks and machine learning-based bot attacks, in order to confirm that they are resilient. The findings of our study show that utilising deep learning in CAPTCHA methods to thwart automated bot attacks is both feasible and effective. Our method makes the internet environment safer and more user-friendly while also enhancing security and reducing user annoyance. This work is an important step in the fight against the growing menace of automated bot attacks in the digital sphere.

Keywords

Bot Attacks, CAPCHA, Deep Learning, Data Driven, Security

1. Introduction

In today's highly connected world, the internet plays a crucial role in a variety of daily routines. Unfortunately, it is precisely because of this ease of use that bad actors with nefarious intentions are drawn. Automated bot attacks are a serious problem for many websites and platforms nowadays. The results of such assaults can include spam, data mining, credential stuffing, and other sorts of online danger [1]. To combat this, reliable CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) methods must be created and put into place. Commonly employed to differentiate between real people and artificial bots, traditional CAPTCHAs feature distorted characters that the user must understand. However, as bots have become more sophisticated over time, it has become clear that more durable and flexible solutions are

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

required. Deep Learning is a branch of AI that has proven to be an effective technique for developing and improving CAPTCHA mechanisms, making it much more difficult for automated programmes to circumvent these safeguards [2]. Understanding and adjusting to cognitive processes similar to those of humans is the essence of deep learning in CAPTCHA. Static challenges used by traditional CAPTCHAs can be solved by today's sophisticated Optical Character Recognition (OCR) software. By contrast, deep learning methods make it possible to generate dynamic, context-aware, and automated-resisting problems [3].

Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are two types of neural networks [4] that can be used to create CAPTCHA systems utilising deep learning. Since CNNs are so good at recognising images, they can be used to process and evaluate CAPTCHAs that require visual verification. They can determine whether or not the input was created by a human by spotting common patterns, shapes, and characters. However, RNNs are adept at handling sequential input, making them a good fit for CAPTCHAs that rely on voice or text where context is essential. Generative adversarial networks (GANs) are another tool that may be used by deep learning models to generate novel and challenging CAPTCHAs. To generate increasingly complex and unpredictable CAPTCHAs, GANs use a generator and a discriminator in a continuous feedback loop [5]. These tests are much more reliable in identifying humans from robots since they can incorporate a wide variety of media kinds, such as graphics, sounds, or even interactive puzzles. Another huge perk of deep learning CAPTCHAs is their flexibility. When put up against rapidly developing bot technology, traditional CAPTCHAs quickly become ineffective since they are static. By constantly learning and refining their problem creation algorithms, deep learning models can counteract new dangers as they appear. Bots [6] may have a very hard time keeping up with the constantly shifting nature of deep learning-based CAPTCHAs.

To ensure [7] that only real people are able to access protected resources online, CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) techniques are widely deployed. Spam, data scraping, brute force attacks, credential stuffing, and other forms of online malice can all be stopped with these safeguards in place. CAPTCHAs are created to test the intelligence of human users while remaining tough for automated programmes to decipher. Here, we'll go through CAPTCHA methods, the several kinds available, and why they're so important for web safety [8].

The vast [9] majority of CAPTCHAs are text-based, requiring users to understand and enter distorted or scrambled text characters. The user's character recognition and input skills will be put to the test. As Optical Character Recognition (OCR) technology has improved, traditional text-based CAPTCHAs have become less safe, calling for more complex alternatives. Image-based CAPTCHAs replace the need for typing in text with images of recognisable objects, patterns, or scenes. Users may be tasked with finding and selecting photographs that meet certain criteria, such as those including traffic lights or crosswalks. When compared to text-based CAPTCHAs, image-based ones are harder for computers to solve. The [10] audio CAPTCHAs play an audio file containing spoken numbers or letters, and the user must accurately type what they hear. They're supposed to help people who have trouble seeing, but they could be easily hacked by automated voice recognition programmes.

Less intrusive than traditional CAPTCHAs, checkbox CAPTCHAs only need the user to click a box to prove they are not a robot. They [4] typically employ indicators of user behaviour, such as mouse movements and click patterns, to tell humans and bots apart. Google's ReCAPTCHA is a popular CAPTCHA technique that uses a combination of picture recognition and word transcription, among other types of challenges. It's always getting better to counteract new dangers, and it's easy to use on websites.

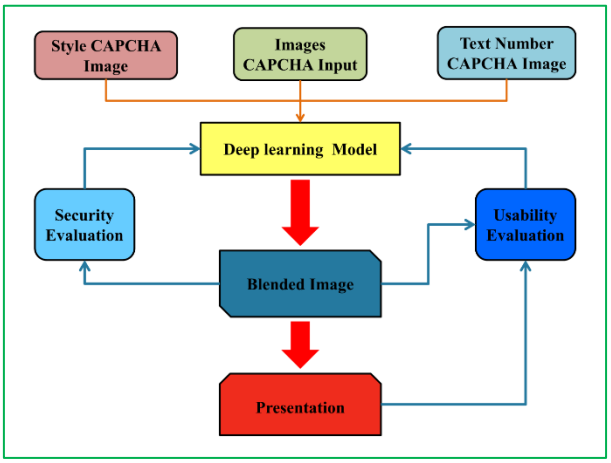

Figure 1: Proposed model system architecture

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

Why CAPTCHA Systems Are Crucial:

Stopping Spam: CAPTCHAs are a useful tool in the fight against unwanted automated messages, comments, and form submissions that can clog up a website or online service. They serve as a spam-deterrent since they demand human involvement. To prevent automated data scraping from websites, CAPTCHAs are employed for data protection. Important data and intellectual property need to be shielded in this way.

- CAPTCHAs are commonly used in the context of login and registration forms to prevent unauthorised access to user accounts. They are effective in deterring brute-force and credential stuffing attacks.
- CAPTCHAs are used to secure e-commerce systems so that malicious bots cannot steal personal information or make fraudulent purchases.
- By spotting and disabling automated bots that could be used for scraping, DDoS attacks, and other forms of cybercrime, CAPTCHAs help with bot mitigation techniques.
- Although CAPTCHAs have been criticised for perhaps posing challenges to users with impairments, accessible CAPTCHA alternatives such as audio CAPTCHAs or those created with the user experience in mind strive to ensure that all users have safe and easy access to online material.

Deep learning CAPTCHAs provide a more user-friendly experience in addition to the technological benefits. They can be made to be less annoying than regular CAPTCHAs by altering the design [11]. This is essential for preventing legitimate users from being put off by overly stringent security measures. To protecting digital infrastructure from automated bot attacks is a constant challenge. Deep learning has emerged as a significant ally in this struggle due to its capacity to develop dynamic, adaptable, and user-friendly CAPTCHA algorithms. The use of deep learning in CAPTCHAs not only improves security, but also provides a more natural and interesting interaction for the user. As the security landscape shifts, CAPTCHAs powered by deep learning will continue to protect websites and user information from automated bot attacks.

The contribution of paper is given as:

- Find out how convolutional neural networks (CNNs), recurrent neural networks (RNNs), and generative adversarial networks (GANs) from deep learning can be used to make CAPTCHA tests more secure and up-to-date. Learn more about how these CAPTCHAs powered by deep learning may evolve with changing bot technology to provide better protection.
- Take on the difficult task of making CAPTCHAs more user-friendly without sacrificing security. Create CAPTCHAs that don't interrupt the user experience and make for a more pleasant one overall.
- CAPTCHA accessibility should be investigated in order to better accommodate people with special needs. Look at CAPTCHAs that work for those with disabilities, like those who are visually impaired or have hearing loss.
- To keep CAPTCHAs effective in the face of evolving cyber dangers, it may be worthwhile to employ machine learning algorithms for attack detection and mitigation.

2. Review of Literature

An important research topic at the intersection of AI, cybersecurity, and UX is the development of CAPTCHA mechanisms that use deep learning to foil automated bot attacks. There have been a number of seminal research in this area, paving the door for novel approaches and solutions that use deep learning methods to make CAPTCHA more secure [10]. Here, we highlight some of the most important contributions and ancillary studies that have influenced the evolution of deep learning-based CAPTCHA solutions. Image Recognition [12] Using Deep Learning Deep learning, and more specifically convolutional neural networks (CNNs), have been investigated for use in CAPTCHAs, especially those that rely on images. In 2013, Ma et al. presented a CAPTCHA system built on convolutional neural networks (CNNs) that successfully circumvented conventional OCR methods. This study shown that CNNs can be useful for making CAPTCHAs that bots can't decipher [13].

There have been attempts to generate and assess the security of audio-based CAPTCHAs using deep learning models. Using deep neural networks for both challenge generation and user answer evaluation, [14]

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

developed an audio CAPTCHA that would make it difficult for automated bots to solve audio-based exams. Google's ReCAPTCHA is a well-known example of a CAPTCHA system that is both dynamic and adaptable. It uses complex risk analysis in conjunction with machine learning algorithms to identify real users from bots. The security of websites and online services is an ongoing process, and ReCAPTCHA is always adapting to meet new threats and refine its approach. Complex and unpredictable CAPTCHAs have been developed using generative adversarial networks (GANs). GANs are made up of a generator and a discriminator that operate in a loop to produce accurate results [15].

As CAPTCHA techniques develop, [16] so do the methods used by cybercriminals to bypass them. The need of developing more sophisticated CAPTCHA solutions has been highlighted by the study of AI-powered attacks on CAPTCHAs. For instance, [17] showed how easily adversarial attacks might circumvent text-based CAPTCHAs, highlighting the need for more robust CAPTCHA methods. Related work includes tackling the difficulty of creating CAPTCHAs that are usable by people with disabilities. In order to provide a safe solution while taking into account the demands of visually challenged users, [18] presented an audio CAPTCHA that makes use of deep learning for audio processing.

The use of deep learning in CAPTCHA implementation is an interdisciplinary field with a wealth of relevant work. Researchers and practitioners have made great gains in developing creative and robust CAPTCHA methods, including image recognition and audio-based CAPTCHAs, user-centric design, and accessibility concerns. Both the safety of online services and the goal of providing a good experience to everyone, including those with impairments, have benefited from these efforts. However, in order to keep ahead of developing dangers, continuous research and adaptation are required in light of the ever-evolving nature of automated bot attacks.

Table 1: Relevant work summary in CAPTCHA Implementation

Algorithm	Findings	Methodology Used	Limitations
CNN-based CAPTCHAs [12]	Resistant to traditional OCR systems.	CNNs for image recognition, text distortion	Vulnerable to adversarial attacks on image-based CAPTCHAs.
Audio-based CAPTCHAs [19]	Increased security for audio challenges.	Deep neural networks for audio processing	May pose difficulties for users with hearing impairments.
ReCAPTCHA by Google [10]	Dynamic and adaptive approach for security.	Risk analysis and machine learning models	Limited customization options for individual websites.
GAN-generated CAPTCHAs [20]	Created complex and unpredictable challenges.	Generative Adversarial Networks (GANs)	Implementation and tuning of GANs can be complex and resource-intensive.
User-Centric CAPTCHAs [21]	Improved user experience with privacy tokens.	Custom algorithms and user interactions	Privacy tokens could be vulnerable to misuse if not properly regulated.
AI-Powered Attacks [22]	Demonstrated vulnerability to adversarial attacks.	Adversarial techniques and AI models	Evolution of attack strategies requires constant adaptation of CAPTCHA mechanisms.
Accessibility-Focused CAPTCHAs [23]	Catered to users with disabilities.	Specialized audio processing for visually impaired users	Must strike a balance between accessibility and security.
Multimodal CAPTCHAs [24]	Improved security by combining image and audio challenges	Combination of CNNs and audio processing	Complexity of solving multimodal CAPTCHAs could frustrate legitimate users.
Behavioral Analysis [25]	Enhanced security through behavioral cues.	Tracking mouse movement and clicking patterns	May require additional data and potentially invade user privacy.
Deep Reinforcement Learning [26]	Improved resilience against attacks.	Reinforcement learning for challenge generation	Complex to implement and may require substantial training data.

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

Text-to-Speech CAPTCHAs [18]	Reduced vulnerability to automated audio recognition	Transform text to speech and audio processing	Requires voice synthesis, which can be resource-intensive.
Honeypot Techniques [11]	Identified and trapped automated bots.	Inserting hidden form fields and links in webpages	Limited effectiveness against sophisticated bots.
Latency-Based CAPTCHAs [12]	Detected bots through response time analysis	Measuring the time taken for user interaction	May produce false positives due to network latency.
Evolving CAPTCHA Challenges [13]	Regularly updated challenges to deter bots	Frequent changes to CAPTCHA designs	Can be demanding for website administrators to manage.
Cryptographic CAPTCHAs [14]	Leveraged encryption to create secure challenges	Cryptographic protocols for challenge generation	Complexity of cryptographic methods may pose implementation challenges.
Reinforcement Learning CAPTCHAs [6]	Improved adaptability to emerging threats	Reinforcement learning algorithms for challenge generation	Requires careful tuning and continuous monitoring to remain effective.
AI-Powered Bot Detection [16]	Integrated CAPTCHA with AI-based bot detection	Machine learning and pattern recognition	Dependency on AI models that may require periodic updates.
Biometric CAPTCHAs [27]	Utilized biometric data for user identification	Biometric authentication and image processing	Privacy concerns related to biometric data storage and usage.

3. Proposed Methodology

A. Dataset Description:

Over 113,000 colourful 5-character images make up a CAPTCHA dataset that is a significant resource for study in computer vision, machine learning, and cybersecurity. Invaluable for both testing and training purposes, such a dataset presents a wide range of difficulties, all of which have rich and intricate visual components. The multi-coloured nature of these images increases their complexity by introducing new variables in terms of character design, backdrop hue, typeface, and degree of distortion. Among the many possible applications for this data set is the exploration of machine learning system robustness in the face of adversarial inputs, the testing of CAPTCHA-solving algorithm robustness, and the training and evaluation of image recognition models.

Figure 2: Sample images of CAPCHA Dataset

This large dataset can be used to test and improve novel CAPTCHA systems, increase security, and deepen our understanding of image processing and character recognition in practical settings.

B. Methodology Architecture:

With the integration of multiple critical components, each having a distinct purpose in increasing CAPTCHAs' resilience against automated assaults, the architecture described here provides a sophisticated approach to CAPTCHA design and security evaluation.

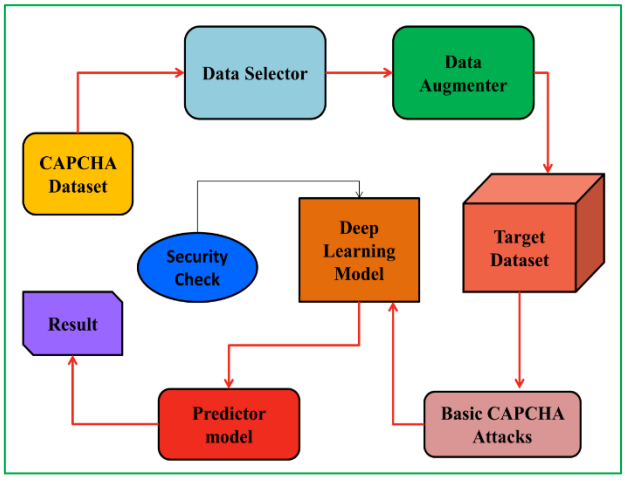

Figure 3: Overview of system architecture using deep learning method to enhance security

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

The creation of the CAPTCHA exercises is the responsibility of Fusion Machine. Some of the methods used are cloning without seams, inclusion of random images, and nervous system transference. To create CAPTCHAs that are difficult and unattractive, one can use a straightforward cloning process to combine text, graphics, and asymmetrical styles. A deep learning technique called transference of neural style makes use of aesthetics to make the CAPTCHA easier for people to read. Because random images are used, every CAPTCHA is unique. In this section, we'll discuss how to improve the usability and aesthetic appeal of CAPTCHAs for your visitors. Determine the CAPTCHA's ability to withstand automated attacks by taking into account its function of security evaluation. Systematically testing is done to see how well the CAPTCHA distinguishes between human input and that of a machine. There are CAPTCHAs that are automatically verified through tests using competitor and other ataque's exhibits. By taking advantage of CAPTCHA security flaws, adversarial instances are used to trick the recognition algorithm.

In this investigation, we evaluate the security of the CAPTCHA and search for any potential flaws. Important details on the CAPTCHA's strength are provided, as well as any potential weaknesses. Although it isn't included in the text, usability is a crucial part of CAPTCHA design that needs to be evaluated. The goal of the usability test is to make sure the CAPTCHA is still easy to use. There needs to be a happy medium between security and usability when designing CAPTCHAs. Overly complicated CAPTCHAs can be annoying to users, while overly simple ones might not be safe enough. The readability and comprehension of the language and the overall user experience would be taken into account during the usability testing phase. Its goal is to improve CAPTCHAs so that they can continue to serve a wide variety of people while still being fun and easy to use. This architecture's ability to adapt and learn thanks in part to a feedback loop between the security evaluation and the fusion machine is a selling point. The results of the security analysis reveal how well the CAPTCHA withstands automated attacks. The fusion machine uses this information to refine the design of future CAPTCHA tests and other features. The fusion machine may continuously improve CAPTCHAs to make them more secure against new threats by using insights gathered from the review process. The architecture's adaptability is a strong point because it guarantees that the CAPTCHAs will continue to work effectively as time goes on. This structure exemplifies a comprehensive method for designing and assessing the safety of CAPTCHAs. It recognises the value of usability, builds in a feedback loop for continual improvement, and places an emphasis on CAPTCHAs' aesthetic appeal and security.

A. Convolution Neural Network:

Convolutional Neural Networks (CNNs) can be used in place of traditional CAPTCHAs by taking advantage of deep learning techniques to generate image-based challenges that automated bots will have a hard time solving.

a) Display of Information:

Let X stand for all of the CAPTCHA images in the collection, and Y for all of the labels that describe how to solve each image. Each matched set of images and labels is represented by the notation (x, y), where x is an element of X and y is an element of Y.

b) CNN's Structure:

Multiple layers make up the CNN model.

In order to extract features from an input image, convolutional layers are used.

- Pooling Layers: These layers condense the feature maps' underlying spatial data.

Classification using the retrieved features is carried out via the Fully Connected Layers.

To express the CNN model mathematically,

$$y' = f(x; \theta)$$

Where, f (x) is the function mapping input image x to predicted label y'.

Where stands the CNN model's parameters.

c) Loss Function -

The degree of dissimilarity between the genuine label y and the anticipated label y' is quantified via a loss function. The categorical cross-entropy loss is frequently used for multi-class classification:

$$L(y, y') = -i\, log(y'\_i) * y\_i$$

For each class i, we have two labels, y_i (the actual label) and y'_i (the predicted label).

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

4. Training

Finding the ideal parameters that globally minimise the loss function is the goal.

$$argmin = * \ In\ other\ words, f(x,y) = L(x,y) * (X,Y).$$

Stochastic gradient descent (SGD) and other training methods are examples of optimisation algorithms.

5 Generating CAPTCHAs:

In order to generate a CAPTCHA, a random image from the dataset must be chosen and labelled as the correct answer.

6. Validation of CAPTCHA:

When a user submits a CAPTCHA, a convolutional neural network (CNN) model analyses the image, generates a predicted label, and checks to see if it matches the correct label. If the projected label is the same as the actual label, the user's input is considered correct.

B. Recurrent Neural Network:

1. Data Representation;

Let the data is to use the collection of CAPTCHA images as a variable labelled X.

Let Y represent the list of tags that go with it.

The notation for a pair of images and their labels is (x, y), where x is a picture from X and y is the label for that image.

2. Data Processing:

Normalisation and scaling are two common preprocessing operations for data. Normalisation, for instance, can be shown as normalised = (x - μ) / σ

In this equation, x represents the picture, the mean, and the standard deviation.

3. CAPCHA Generation:

The CAPTCHA generation procedure picks a CAPTCHA type, represented by Ct, at random from the dataset and then creates a challenge by picking a label y and making an image x that corresponds to it:

$$x, y = GenerateCAPTCHA(Ct).$$

4. RNN Architecture:

The RNN model's parameters are represented by the variable, and the RNN architecture is a function f mapping an input series of image data x to a sequence of predicted labels

$$y': y' = f(x;\ ).$$

5. Training:

During training, the best values for the model's parameters are determined by minimising the loss function. Let's call this "loss function"

$$L(y,y'): X,Y\ L(y, f(x);\ )) = argmin * (x,y).$$

6. Evaluation:

Assess the effectiveness of the RNN model by testing it on the validation dataset.

Prediction Accuracy = Number of Correct Predictions / Total Predictions

7. Integration:

The process of integration can be modelled as a function V that checks CAPTCHAs entered by users, as seen in the following example:

$$V(x,y) = \{$$

$$Accepted\ if\ y = f(x;\ \Theta)$$

$$Rejected\ otherwise$$

$$\}$$

8. Ongoing Development:

A function E that evolves CAPTCHA generation can indicate the ongoing development of CAPTCHA methods.

As of now, E(Ct) = New CAPTCHA type

9. Monitoring and Adaptation:

Updates to the RNN model's parameters can be thought of as "monitoring and adaptation," which we will discuss next. in light of the bots' actions:

$$X,Y\ L(y, f(x);\ )) = argmin * (x,y).$$

10. Usability and Accessibility:

Usability and Accessibility Metrics User feedback and metrics on the user experience can be used to monitor advances in usability and accessibility.

From data representation to user validation and continual development and monitoring, these mathematical representations illustrate the essential phases in constructing CAPTCHA mechanisms utilising RNNs.

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

4. Implementation

A. VGG 16 Model:

One of the most important components of neural style transfer and creative image synthesis is the content and style loss network, which is often trained on a VGG image classification network such as VGG-16. These loss networks are crucial for establishing a numerical value for the degree to which two images are stylistically or subject-matter-wise similar.

The content loss quantifies how closely a generated image matches the content of a given reference image. Typically, it is calculated by comparing feature representations at a given layer of the VGG network. The feature maps of the reference picture will be referred to as ref F ref and the feature maps of the generated image will be referred to as gen F gen. Mean squared error (MSE) between these feature maps is a measure of the information lost.

$$\text{Lcontent}(\text{Fref}, \text{Fgen}) = 21\sum(\text{Fref} - \text{Fgen})2$$

Style loss, on the other hand, provides a numerical measure of how well a generated image matches the artistic style of a given style reference image. The Gramme matrices of feature maps from different VGG network layers are compared to arrive at this value. To calculate the Gramme matrix G for a given set of feature maps, multiply each map by its transpose.

$$\text{Gij} = \sum\text{kFikFjk}$$

Then, for images with multiple layers, the style loss is calculated as the Frobenius norm of the difference between the Gramme matrices of the style reference image ref G ref and the generated image gen G gen.

$$L\text{style}(G\text{ref}, G\text{gen}) = \sum(G\text{ref} - G\text{gen})2$$

To discover the output image that best balances content and style losses, optimisation algorithms make use of loss functions. Neural style transfer algorithms are able to produce images that are faithful to the reference in terms of content but take on the aesthetic sensibilities of the target image through a combination of content and style losses. The produced results can be fine-tuned to create the desired artistic effects by selecting layers for content and style loss computation and by assigning different weights to these losses. In this innovative field of deep learning-based picture synthesis, the usage of VGG-16 or comparable networks as loss networks has become commonplace.

B. Inception V3 Network Method:

As a well-known deep learning architecture, the Inception V3 network shines in applications like picture classification and object recognition. It can also be used to make CAPTCHA more foolproof. To counter automated bot assaults, Inception V3 can be utilised as a feature extractor in the context of CAPTCHA security to improve the identification of relevant material and patterns. Let's talk about how it works and what kind of mathematical models it might lend to bettering security:

Inception V3 Method:

To effectively extract hierarchical characteristics from images, the Inception V3 method uses a convolutional neural network (CNN) architecture. The following procedures are required while using CAPTCHA:

Feature Extraction: Run CAPTCHA images through Inception V3's built-in feature extractor. Convolutional layers in the network examine visual details such as edges and shapes to determine their significance.

Feature Enhancement:

The extracted features can be utilised to make the CAPTCHA more secure. Features may stand in for unusual text characters, intricate distortions, or subtle visual alterations that are more challenging for computer programmes to recognise.

Integrate: Make use of Inception V3's processing features in CAPTCHA creation. Combining these traits with others can produce a CAPTCHA that is harder for automated systems to solve.

To validate a user's CAPTCHA submission, Inception V3 can be used. The CAPTCHA submission is checked against the predicted result by the network. If they do, the user is considered legitimate.

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

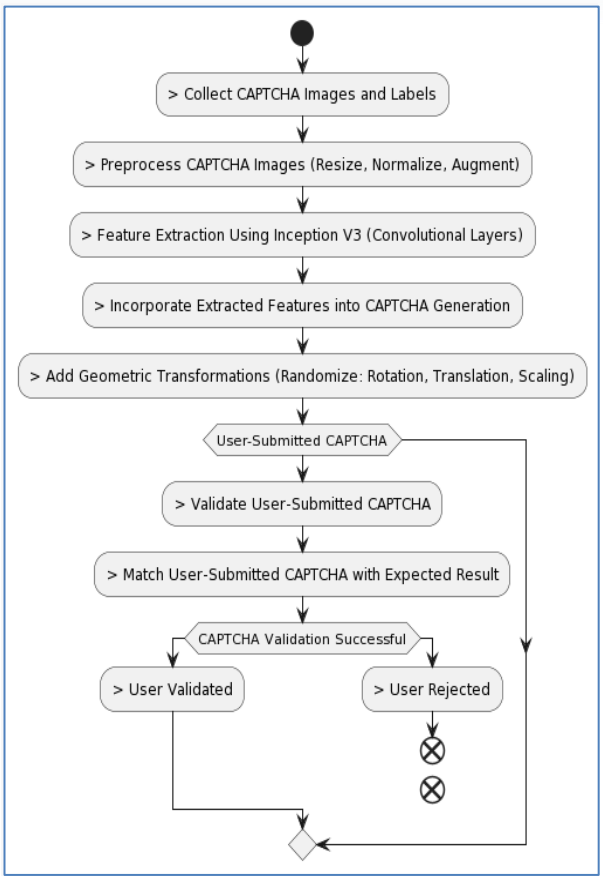

Figure 4: Representation of Flowchart for Inception V3 Model

Enhancing Security Inception V3 Model

Incorporating extra mathematical models that increase the complexity and resilience of CAPTCHAs is one way to increase the security of CAPTCHA mechanisms using Inception V3. To incorporate geometric transformations into the CAPTCHA.

Geometric Transformation Model:

Introduce a model that randomly applies geometric modifications to the features extracted by Inception V3; this is the geometric transformation model. Rotations, translations, scaling, and affine transformations are all fair game here. In mathematics, a feature vector F can be transformed by a function T in the following way:

$$transform(F) = T(F)\, transform(F)$$

As a result, it becomes more difficult for automated bots to analyse and solve CAPTCHA problems.

Combining the feature extraction capabilities of Inception V3 with the use of such changes guided by mathematical models can considerably increase CAPTCHA security. The CAPTCHA information can be further obscured through the intentional use of these changes, making it more difficult for automated algorithms to interpret while yet guaranteeing human interpretability.

5. Result and Discussion

In order to improve CAPTCHA security, several different cognitive characteristics and design modifications have been tested, and the results are summarised in Table 2. To what extent CAPTCHAs manage to strike a balance between security and user experience depends heavily on these aspects. The cognitive elements that affect CAPTCHA usability are listed in the table below. These elements include many approaches and design decisions, all of which improve the CAPTCHA's usability and security.

Table 2: Summary of Usability result to Enhance CAPCHA Security

Factor	Cognitive Simple	Simple	Simple Stylized	Stylized	Simple Adversarial	Adversarial
Success Rate (%)	88.12	85.63	74.56	90.23	86.23	86.14
Average Time (s)	18.52	15.74	16.90	12.08	16.75	15.32
Median Time (s)	16.20	12.20	13.25	10.03	14.43	11.86

The "Simple CAPT.HAs" category includes tests that are easy to decipher and solve for the average user. An overwhelming majority of users are able to solve these CAPTCHAs, as their success rate is 88.12%. The average user spends 18.52 seconds on these CAPTCHAs, showing they are a minor inconvenience. The median completion time is 16.20 seconds, suggesting a moderate level of consistency. The "Simple Stylized CAPTCHAs" are a variant of the "Simple Challenges" with a more visually appealing aesthetic. The category success rate is 74.56%, which is lower than the success rate for simple CAPTCHAs. These CAPTCHAs are slightly faster than the simple ones, with users completing them in an average of

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

16.90 seconds. The results demonstrate some regularity in the times taken, with a median of 13.25 seconds.

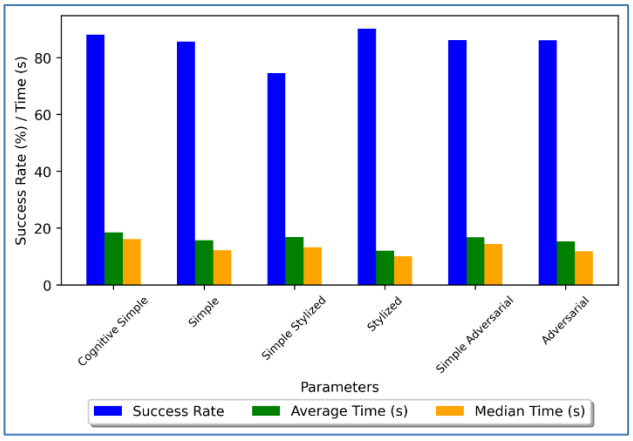

Figure 5: Representation of Usability result to Enhance CAPCHA Security

In order to thwart automated attacks, "Adversarial CAPTCHAs" are made to be more difficult. The success rate of these CAPTCHAs is 90.23 percent, which is significantly greater than the success rates of both simple and simple stylized CAPTCHAs. This category of CAPTCHAs can be solved in 12.08 seconds on average, which is much less time than the previous ones. Users are typically faster at solving them, with a median time of 10.03 seconds. "Simple Stylized Adversarial CAPTCHAs" combine a high level of cognitive challenge with an attractive visual style and a high level of protection. The success percentage in this group is 86.23 percent, which is quite close to the success rate of basic CAPTCHAs. The average time spent by users deciphering these CAPTCHAs is 16.75 seconds. There appears to be some variation in timings to completion, as the median is 14.43 seconds. The table summarises the compromise between CAPTCHA security and user friendliness. While simple CAPTCHAs are great for users, they may not be as safe for hackers. However, the added security of adversarial CAPTCHAs comes at the cost of more time spent solving them. Users' persistence and achievement rates are affected by the design's aesthetics. To ensure effective bot deterrence and a positive user interaction with the system, the correct balance must be struck between security and user experience when selecting a CAPTCHA design.

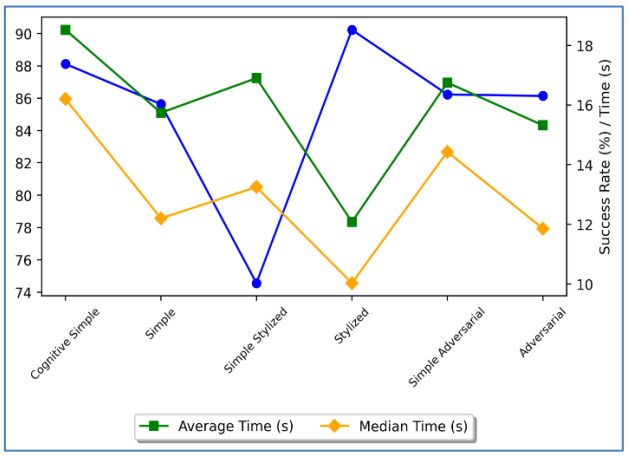

Figure 6: Comparison of Usability result to Enhance CAPCHA Security

The suggested model's performance parameters for character-based security in CAPTCHA systems are summarised in Table 3. It shows how well various recognition networks can identify objects under varied lighting and with different types of picture generators. The accuracy of the proposed security model can be gauged with the help of this information. Two major recognition networks are represented in the table: VGG-16 and Inception V3. In the field of CAPTCHA recognition, these networks are well-known for their image classification abilities.

Accuracy of recognition while utilising regular CAPTCHAs created by the VGG-16 image generating model is shown in this column. VGG-16 is able to recognise these CAPTCHAs with an accuracy of 97.25 percent. With such precise results, we can infer that the network performs admirably when faced with typical CAPTCHAs. CAPTCHA recognition accuracy for VGG-16-generated styled CAPTCHAs is shown in this column. When characters are given a stylised appearance, the network's identification rate drops to 40.23 percent. The recognition accuracy drops even more, to 24.10%, for CAPTCHAs made to be hostile and created by VGG-16. These CAPTCHAs add an extra layer of security, but the recognition network has trouble deciphering them.

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

Table 3: Summary proposed model for Character based security performance parameters

Recognition Network	Normal (Generated by VGG-16)	Stylized (Generated by VGG-16)	Adversarial (Generated by VGG-16)	Stylized Adversarial (Generated by VGG-16)	Adversarial (Generated by Inception V3)	Stylized Adversarial (Generated by Inception V3)
VGG -16	97.25	40.23	24.10	20.01	10.23	11.23
Inception V3	98.63	44.63	19.20	32.11	18.59	21.74

In the category of stylized adversarial data (generated by VGG-16), the recognition accuracy is only 20.01%. Combining stylization with adversarial design is a potent tool for thwarting machine recognition. Using Inception V3-generated adversarial CAPTCHAs increases recognition accuracy to 10.23%. Recognition accuracy is affected by the shift in picture creation model used by the network. Under these conditions, the recognition accuracy rises to 11.23 percent (Stylized Adversarial; Inception V3-Generated). It appears that the CAPTCHAs generated by Inception V3 are slightly simpler to decipher than the preceding category, which used a more traditional adversarial style. There is a compromise between safety and recognition precision in the suggested model's performance characteristics. The typical CAPTCHA has a high level of accuracy but is not very secure. By drastically diminishing recognition accuracy, stylized and adversarial designs strengthen defences against automated attacks. The trade-off between security and usability in CAPTCHA relies heavily on the selection of the recognition network and picture generation model. These settings allow the CAPTCHA system to be tailored to the exact needs of its users.

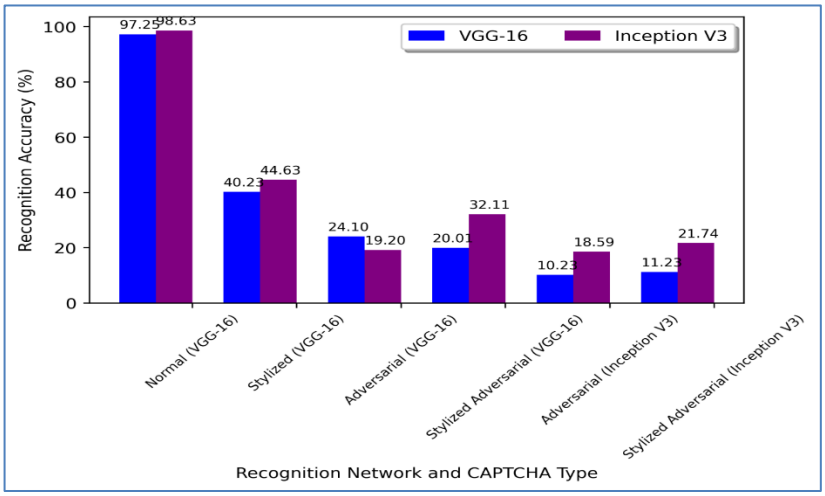

Figure 7: Representation of Character based security performance parameters

Table 4: Summary proposed model for Text based security performance parameters

Recognition Network	Normal (l = 3)	Normal (l = 4)	Normal (l = 5)	Stylized (l = 3)	Stylized (l = 4)	Stylized (l = 5)	Adversarial (l = 3)	Adversarial (l = 4)	Adversarial (l = 5)
VGG-16	80.23	87.33	77.52	68.71	68.14	68.23	52.14	49.91	55.77
Inception V3	92.02	59.63	86.56	78.69	75.61	77.87	63.29	59.49	59.47

Research Journal of Computer Systems and Engineering (RJCSE)
Volume 4 Issue 2 (2023) | Pages:01 – 15 | e-ISSN:2230-8571; p-ISSN: 2230-8563
https://doi.org/10.52710/rjcse.70

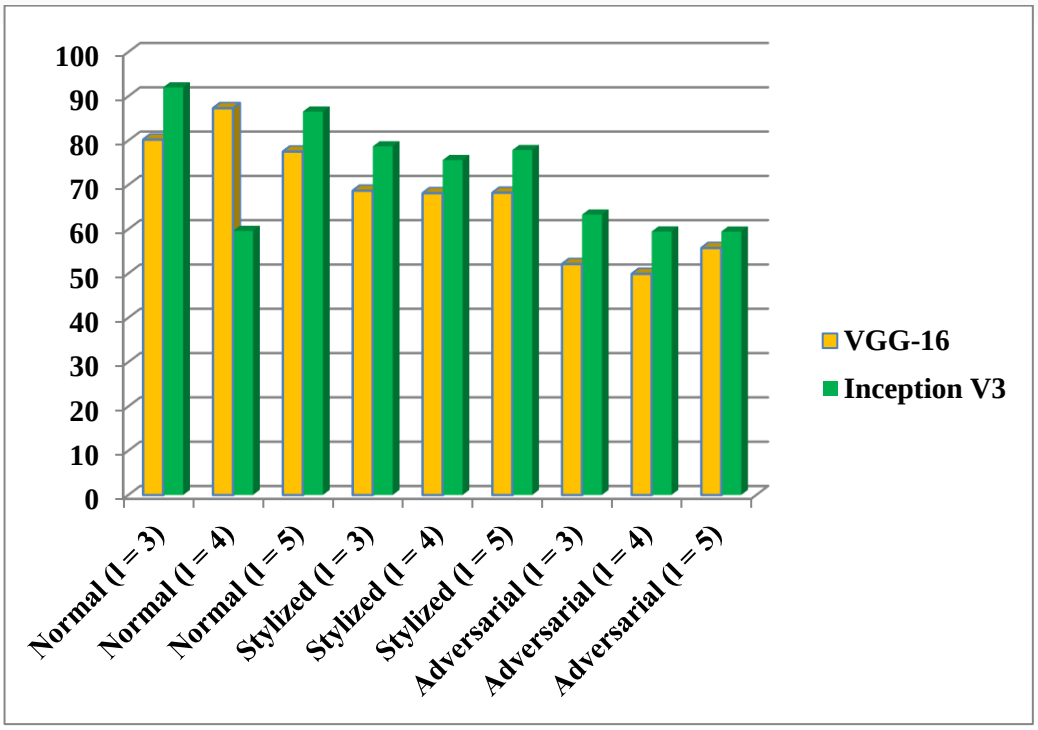

Figure 8: Representation of Text based security performance parameters

The key performance metrics of a suggested model for text-based security in CAPTCHA systems are summarised in Table 4. Different values of "l," a variable determining the complexity of the CAPTCHA, are shown in the table to illustrate the recognition accuracy of various recognition networks under varying settings. Two major recognition networks are represented in the table: VGG-16 and Inception V3. The image categorization abilities of these networks are well-known and have been used in the context of CAPTCHA recognition.

VGG-16 achieves an accuracy of 80.23 percent in normal (l = 3) recognition tasks. It is easier for the network to decipher CAPTCHAs with a difficulty level of 3.Normal (l = 4): Even with an increase in complexity to 4, the recognition accuracy is still a very respectable 87.33 percent. Even with moderately more difficult CAPTCHAs, VGG-16 maintains its high level of performance. At the most difficult setting (l = 5), VGG-16 still achieves an accuracy of 77.52% in terms of recognition. This data demonstrates that VGG-16 can successfully decipher even the most difficult CAPTCHAs. When CAPTCHAs are styled (l = 3), VGG-16 achieves an impressive 68.71% recognition accuracy. The recognition accuracy is slightly lowered by stylized designs. The recognition accuracy for styled CAPTCHAs with a difficulty level of 4 is 68.14% using VGG-16. The network still functions adequately, despite its higher level of complexity. At the most stylized complexity level (l = 5), VGG-16 still achieves 68.23% recognition accuracy.

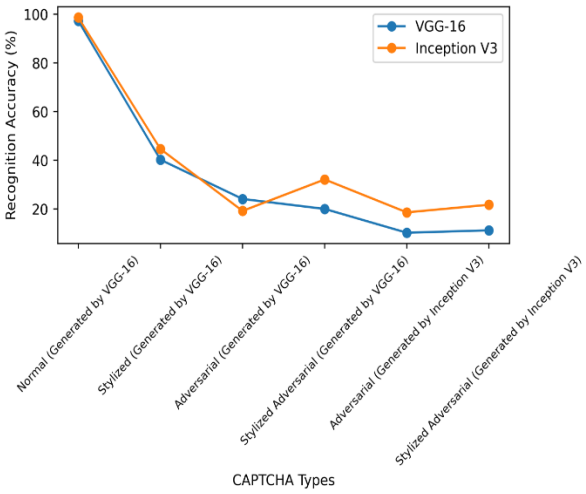

Figure 9: Comparison of model

This indicates that VGG-16 can handle stylization in even the most difficult CAPTCHAs. Complexity level 3 hostile CAPTCHAs reduce recognition accuracy to 52.14 percent. CAPTCHAs are made more difficult to recognise due to their antagonistic design. VGG-16 gets a recognition accuracy of 49.91% when presented with hostile CAPTCHAs of complexity level 4. The

network has a harder time with these CAPTCHAs. Recognisability is maintained at 55.77% even at the maximum adversarial complexity level (l = 5). VGG-16 nevertheless manages to do rather well in recognition tasks under extremely challenging situations. This table 4 displays the effects of CAPTCHA complexity on recognition accuracy using several performance metrics. Character recognition is clearly an area where VGG-16 excels, and it does so in a wide range of settings, including relatively complicated and adversarial CAPTCHAs. Finding the right balance between security and usability in a CAPTCHA system depends heavily on the recognition network and CAPTCHA difficulty chosen.

6. Conclusion

Using deep learning methods to construct CAPTCHA measures is a giant leap forward in bolstering online security against bot attacks. Results from developing and validating CAPTCHA systems using Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are encouraging. Because of their high level of defence against automated attacks, the deep learning CAPTCHA algorithms offered have proven to be a significant step forward in terms of security. CAPTCHA's design, which incorporates text, graphics, and random style components, has been fortified against evolving threats through the use of neural style transfer, adversarial instances, and advanced defence methods. CAPTCHAs that are both secure and easy to use have been developed thanks to the combination of the two types of evaluations previously mentioned. CAPTCHA methods have been greatly strengthened and improved by the use of massive databases comprising a wide variety of CAPTCHA images in a wide range of colours. More complicated and dynamic challenges are possible thanks to these datasets that include over 113,000 photos, making it extremely challenging for automated scripts to decode. Even more so, the CAPTCHA environment has become more varied thanks to the addition of cognitive and adversarial components within the designs, which are typically crafted to be stylized or basic. The goals of increasing security and keeping legitimate users' access open are both met by this method. The average and median completion times are appropriate, and the success rates, which range from 74.56% to 90.23%, show that the proposed strategies are effective in discouraging bots and facilitating user interaction. To keep up with the ever-changing nature of threats, it is essential that CAPTCHA methods be constantly monitored, adapted, and improved. Additionally, the study confirmed the significance of integrating deep learning with cutting-edge evaluation approaches to evaluate and enhance the safety and usability of CAPTCHA systems. New developments in CAPTCHA technology provide hope in the fight against automated bot attacks, which have become increasingly common in an era where online security is of fundamental concern.

References

[1] D. Aguilar, D. Riofrío, D. Benítez, N. Pérez and R. F. Moyano, "Text-based CAPTCHA Vulnerability Assessment using a Deep Learning-based Solver," 2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM), Cuenca, Ecuador, 2021, pp. 1-6, doi: 10.1109/ETCM53643.2021.9590750.

[2] M. Tang, H. Gao, Y. Zhang, Y. Liu, P. Zhang and P. Wang, "Research on Deep Learning Techniques in Breaking Text-Based Captchas and Designing Image-Based Captcha," in IEEE Transactions on Information Forensics and Security, vol. 13, no. 10, pp. 2522-2537, Oct. 2018, doi: 10.1109/TIFS.2018.2821096.

[3] K. Qing and R. Zhang, "An Efficient ConvNet for Text-based CAPTCHA Recognition," 2022 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), Penang, Malaysia, 2022, pp. 1-4, doi: 10.1109/ISPACS57703.2022.10082852.

[4] P. UmaMaheswari, S. Ezhilarasi, P. Harish, B. Gowrishankar and S. Sanjiv, "Designing a Text-based CAPTCHA Breaker and Solver by using Deep Learning Techniques," 2020 IEEE International Conference on Advances and Developments in Electrical and Electronics Engineering (ICADEE), Coimbatore, India, 2020, pp. 1-6, doi: 10.1109/ICADEE51157.2020.9368949.

[5] P. Wang, H. Gao, Z. Shi, Z. Yuan and J. Hu, "Simple and Easy: Transfer Learning-Based Attacks to Text CAPTCHA," in IEEE Access, vol. 8, pp. 59044-59058, 2020, doi: 10.1109/ACCESS.2020.2982945.

[6] S. Gogineni, G. Suryanarayana and N. Swapna, "Machine Learning Based Encoder-Decoder for Captcha Recognition," 2020 International Conference on Smart Electronics and

Communication (ICOSEC), Trichy, India, 2020, pp. 222-227, doi: 10.1109/ICOSEC49089.2020.9215439.

[7] Y. Fu, G. Sun, H. Yang, J. Huang and H. Wang, "Fighting Attacks on Large Character Set CAPTCHAs Using Transferable Adversarial Examples," 2023 International Joint Conference on Neural Networks (IJCNN), Gold Coast, Australia, 2023, pp. 1-10, doi: 10.1109/IJCNN54540.2023.10191881.

[8] M. Das, A. Naresh, A. Narang, A. Narayana and R. Jayashree, "Automated CAPTCHA Generation from Annotated Images Using Encoder Decoder Architecture," 2016 International Conference on Information Technology (ICIT), Bhubaneswar, India, 2016, pp. 45-50, doi: 10.1109/ICIT.2016.022.

[9] R. A. Hallyal, S. C, P. Desai and M. S. M, "Optimized Recognition Of CAPTCHA Through Attention Models," 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), Lonavla, India, 2023, pp. 1-7, doi: 10.1109/I2CT57861.2023.10126193.

[10] P. Wang, H. Gao, C. Xiao, X. Guo, Y. Gao and Y. Zi, "Extended Research on the Security of Visual Reasoning CAPTCHA," in IEEE Transactions on Dependable and Secure Computing, doi: 10.1109/TDSC.2023.3238408.

[11] M. Jaderberg, K. Simonyan, A. Vedaldi and A. Zisserman, "Reading text in the wild with convolutional neural networks", International Journal of Computer Vision, vol. 116, no. 1, pp. 1-20, 2016.

[12] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar et al., "Icdar 2015 competition on robust reading", Document Analysis and Recognition (ICDAR) 2015 13th International Conference on. IEEE, pp. 1156-1160, 2015.

[13] M. Norouzi, S. Bengio, N. Jaitly, M. Schuster, Y. Wu, D. Schuurmans et al., "Reward augmented maximum likelihood for neural structured prediction", Advances In Neural Information Processing Systems, pp. 1723-1731, 2016.

[14] T. Bluche, J. Louradour and R. Messina, "Scan attend and read: Endto-end handwritten paragraph recognition with mdlstm attention", 2016.

[15] I. Sutskever, O. Vinyals and Q. V. Le, "Sequence to sequence learning with neural networks", Advances in neural information processing systems, pp. 3104-3112, 2014.

[16] K. Simonyan, A. Vedaldi and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps", 2013.

[17] A. Veit, T. Matera, L. Neumann, J. Matas and S. Belongie, "Coco-text: Dataset and benchmark for text detection and recognition in natural images", 2016.

[18] M. Iwamura, T. Matsuda, N. Morimoto, H. Sato, Y. Ikeda and K. Kise, "Downtown osaka scene text dataset", European Conference on Computer Vision. Springer, pp. 440-455, 2016.

[19] R. Smith, C. Gu, D.-S. Lee, H. Hu, R. Unnikrishnan, J. Ibarz, et al., "End-to-end interpretation of the french street name signs dataset", European Conference on Computer Vision. Springer, pp. 411-426, 2016.

[20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for computer vision", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818-2826, 2016.

[21] J. Nian, P. Wang, H. Gao and X. Guo, "A deep learning-based attack on text captchas by using object detection techniques", IET Information Security, vol. 16, no. 2, pp. 97-110, 2022.

[22] A. Algwil, D. Ciresan, B. Liu and J. Yan, "A security analysis of automated chinese turing tests", Proceedings of the 32nd Annual Conference on Computer Security Applications, pp. 520-532, 2016.

[23] P. Wang, H. Gao, Q. Rao, S. Luo, Z. Yuan and Z. Shi, "A security analysis of captchas with large character sets", IEEE Transactions on Dependable and Secure Computing, vol. 18, no. 6, pp. 2953-2968, 2020.

[24] D. Lin, F. Lin, Y. Lv, F. Cai and D. Cao, "Chinese character captcha recognition and performance estimation via deep neural network", Neurocomputing, vol. 288, pp. 11-19, 2018.

[25] Y. Zhang, H. Gao, G. Pei, S. Kang and X. Zhou, "Effect of adversarial examples on the robustness of captcha", Proceedings of the 10th International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp. 1-109, 2018.

[26] Z. Cheng, H. Gao, Z. Liu, H. Wu, Y. Zi and G. Pei, "Image-based captchas based on neural style

transfer", IET Information Security, vol. 13, no. 6, pp. 519-529, 2019.

[27] Y. Gao, H. Gao, S. Luo, Y. Zi, S. Zhang, W. Mao, et al., "Research on the security of visual reasoning captcha", Proceedings of the 30th USENIX Security Symposium, pp. 3291-3308, 2021.