I.J. Intelligent Systems and Applications, **2025, 3, 1-14**
Published Online on June 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2025.03.01

Robust, Interactive, and Intelligent Captcha Model Based on Image Processing

Remzi GÜRFİDAN*
YTSVS Computer Programming, Isparta University of Applied Science, Turkey
E-mail: remzigurfidan@isparta.edu.tr
ORCID iD: https://orcid.org/0000-0002-4899-2219

Oğuzhan KİLİM
Computer Programming, Isparta University of Applied Sciences, Keçiborlu Vocational School, Turkey
E-mail: oguhankilim@isparta.edu.tr
ORCID iD: https://orcid.org/0000-0003-3365-7327

Tuncay YİĞİT
Computer Engineering, Süleyman Demirel University, Turkey
E-mail: tuncayyigit@sdu.edu.tr
ORCID iD: https://orcid.org/0000-0001-7397-7224

Received: 17 January 2025; Revised: 28 February 2025; Accepted: 26 March 2025; Published: 08 June 2025

Abstract: Ensuring online security against automated attacks remains a critical challenge, as traditional CAPTCHA mechanisms often struggle to balance robustness and usability. This study proposes a novel intelligent and interactive CAPTCHA system that integrates advanced image processing techniques with a convolutional neural network (CNN)-based evaluation model to enhance security and user engagement. The proposed CAPTCHA dynamically generates images with randomized object placement, adaptive noise layers, and geometric transformations, making them resistant to AI-based solvers. Unlike conventional CAPTCHAs, this approach requires users to interact with images by selecting and marking specific objects, creating a human-in-the-loop validation process. For evaluation, a CNN-based classifier processes user selections and determines their validity. A lightweight embedded software module tracks user interactions in real-time, monitoring selection accuracy and response patterns to improve decision-making. The system was tested on 6,000 images across five categories (airplanes, cars, cats, motorcycles, and fish), with an 80% training and 20% testing split. Experimental results demonstrate a classification accuracy of 99.58%, validation accuracy of 96.15%, and a loss value of 0.2078. The CAPTCHA evaluation time was measured at 47–53 milliseconds for initial validation and 17–23 milliseconds for subsequent evaluations. These results confirm that the proposed CAPTCHA model effectively differentiates human users from bots while maintaining usability, demonstrating superior resilience against automated solvers compared to traditional approaches.

Index Terms: CAPTCHA, Anomaly Behaviors Detection, Intrusion Detection, Image Processing, Machine Learning.

1. Introduction

When logging into my site, creating an account, or renewing the password on the internet sites, the browser asks to prove whether the user is a bot [1]. Before the operation is performed, the scanner sends complex, difficult to read, curved, encodes colours in different letters, sends numbers. In some applications, it is desirable to select objects that are difficult to distinguish in images that are very similar to each other with reduced quality. There is also verification methods made in simple mathematical operations. The browser allows the user who wants to log in to the system to perform operations with the correct entry of complex codes or images.

With the increase in security attacks day by day, verifications and methods of preventing bot users' attacks continue to be developed. In addition, when the user wants to login to the account, reset or change the password, the email account encounters the Google authenticator. The user is prompted to enter credentials or email address. If the password is entered incorrectly, the password is requested to be re-entered, and the authenticator sent by the server along with the password is allowed to enter the system [2].

Verifiers are made using lowercase, uppercase letters, punctuation marks, Chinese characters, Arabic letters, and numbers. 8 types of authentication codes are used, including symbols as well as letters and characters [3]. Firewalls are used to prevent unauthorized users from logging into the system. When the system is attacked, firewalls do not detect the attack instantly. Intrusion prevention systems are defined as software that detects an attack on the network or system instantly and tries to prevent it. It is observed that cyber attackers make attacks by imitating firewalls.

It is observed that traditional firewalls and protection methods are insufficient today. When the attacks in the last few years are examined, it has been determined that the security attacks have increased. Artificial intelligence and machine learning-powered interceptors are becoming popular instead of traditional firewalls. Artificial intelligence algorithms can detect data as malicious or not and classify data [4].

CAPTCHA is a cybersecurity measure that stands for Completely Automated Public Turing test to tell Computers and Humans Apart. It is used to conduct a test to verify that users are human. They are often asked to correctly enter letters or numbers in a picture or noise. Security codes are required when logging into websites, creating an account, or renewing passwords. This method is widely used as the security system of websites [5]. It is integrated into a form and access to the system is provided in the form of question and answer. The server allows the user to enter the system in the form of questions and answers with randomly determined forms [6]. The use of images consisting of letters and numbers is most used in the protection of Internet forms. The user is asked to enter the images consisting of letters and numbers written in a complex at the places indicated by the box [7]. It is used as a validator of complex, curved, hard-to-read letters. Complex letters can only be perceived by Humans and are difficult to detect by bot users. As the development of artificial intelligence reduces the security of complex codes, CAPTCHA development efforts are increasing. Math CAPTCHA, in which mathematical operations are performed, Object CAPTCHA, in which letters are made difficult to read by adding visuals, image CAPTCHA in which images are selected, and CAPTCHA verification types in which words are written from texts have been developed. Security vulnerabilities are decreasing with the developing artificial intelligence in the CAPTCHA application [8].

The remainder of this paper is structured as follows: Section 1 provides a review of related work, discussing existing CAPTCHA models and their vulnerabilities. Section 2 presents the details of the proposed model, including image processing techniques, machine learning architecture, and system implementation. Section 3 describes the experimental setup, dataset, evaluation metrics, and testing condition, analyzes the experimental results, comparing the proposed model with existing approaches. Finally, Section 4 concludes the paper by summarizing the findings and suggesting directions for future research.

1.1. CAPTCHA Types

A. Game CAPTCHA

In this technique, which makes use of the Wear (CMU) application, it complicates the words and presents them to the user's verification. Developed with Yahoo, this technique is used to generate free e-mail [9]. Words that are easy to pronounce are not foreign and can be guessed by the user [10]. This verification, called baffle text, randomly positions the letters in meaningful words vertically or horizontally [11].

B. Bongo

Bongo CAPTCHA is a visual image classification verification technique. In this technique, where a picture set is used, the image can be in different forms. Then a single picture is given, and it is requested to choose which picture set it belongs to. The percentage of verification is high because the bot user option is limited. This authentication technique is not very secure [9]. An example of the Bongo CAPTCHA type is shown in Figure 1.

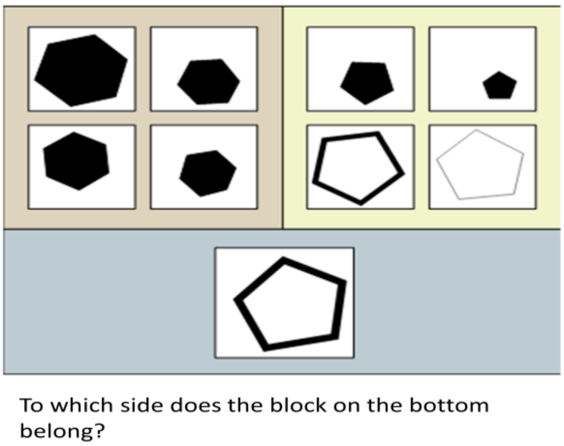

Fig.1. An example of the Bongo CAPTCHA

C. Pix

Photographic or motion pictures of confirmatory images are used. It prompts the user to select the correct photo with the object in all images [9]. In another application, a certain image mountain is pushed into the images and the name of the common character is requested to be entered [12].

D. Voice Based

It adds noise and some distortion to the sound by selecting simple words or numbers. The user enters the word they hear into the box and their entry into the system is verified [9].

E. Split Text

It is a version of the Gimpy technique. In the split-text technique, easy-to-find letters of meaningful and simple words are extracted. The user can easily find this word, but the bot remover is difficult to detect [9].

F. Arithmetic CAPTCHA

This system, also called Mathematical CAPTCHA, asks the user to enter the result of the arithmetic operation in the box by the server. The numbers used in Arithmetic Operation are complex and noise is added to the image. Not every user may be able to answer difficult Transactions. Therefore, the mathematical operation is chosen simple. Simple selection of the Mathematical Operation narrows the validation set [6]. An example of the Arithmetic CAPTCHA type is shown in Figure 2.

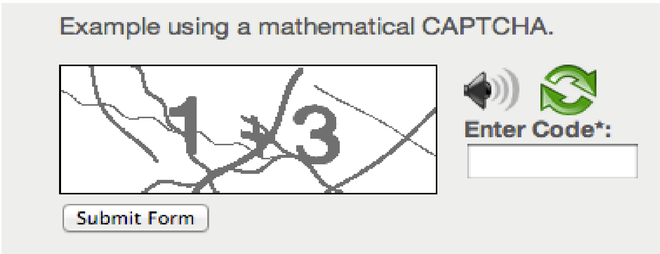

Fig.2. An example of the Arithmetic CAPTCHA [31]

G. Text Based CAPTCHA

In Text-Based CAPTCHA, verification is the presentation of images consisting of letters or numbers as plain text, not in the form of giving them to the server. It can be presented in the form of a plain text questionnaire. The user is asked to enter the answer to the question in the box. Having the validator in plain text can be easily broken by OCR [6, 13]. Text-based CAPTCHA is less secure compared to other types of validation. An example of the Text-Based CAPTCHA type is shown in Figure 3.

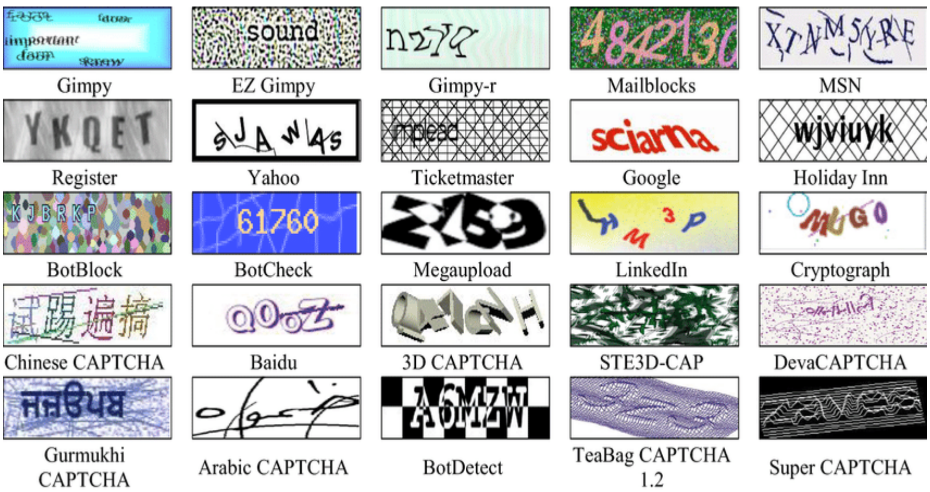

Fig.3. An example of the Text-Based CAPTCHA [14]

H. Google Security

Information in old books is kept visually in Google's memory. Recording the words in this visually stored information as text in the memory allows us to reach the information more sensitively. In this technique, which started to be used in 2009, the server sends the previously unread words in the texts to the user as validators. When the user enters the word correctly in the box, they are allowed to enter the system. The word that has not been entered into the system before is saved by the server [1, 6].

I. Google NoCAPTCHA

Introduced in 2014 as the new version of Google ReCAPTCHA, NoCAPTCHA is verified by ticking a box. By ticking the "I am not a robot" box, validation is made and login to the system is more sensitive. If the server thinks that the user is a bot, it creates a second security by requesting one of the old-style authentications [6, 15].

G. Annulus CAPTCHA

Gao et al. They offered an image-based verification. A ring is placed on images from 1 to 8. The user is expected to enter the number of rings in the image correctly. The authenticity and security of Annulus CAPTCHA has been proven [15, 16].

K. Pedometric CAPTCHA

The authentication technique developed by Kulkarni et al. is used in Internet browsers of mobile devices. Techniques such as visual, mathematical, text, sound verification are not used. This verification technique, which is suitable for the use of visually, hearing and speech impaired people, is done by moving the mobile device. The server asks the user to take 5 steps and when he starts to move, the number of steps is counted. 64 users were tested and 92.2% of this verification stated that the interface is easy to use [17].

L. Handwritten CAPTCHA

In this method, the handwritings in the letters collected from the mailboxes in America are used as validators. The user is expected to enter the word in the handwriting correctly into the box [12].

M. Implicit CAPTCHA

In this technique, an image is given, and the user is asked to click on the right place in the image as text, for example, a duck swimming in a lake is given and the user is asked to click on the duck [12].

N. Darwin CAPTCHA

Method and colleagues presented a new verification method. This method, which can be used on keyboards or touch screens, provides verification. A mountain of 100 dots is randomly pushed onto the squared paper. 3 or more of the dots are diamond shaped. Users are expected to create polygons by combining diamonds. Circles and diamonds are very difficult to detect by bots, making detection by OCR or artificial intelligence difficult [18].

The complexity of Google validators can also make it difficult for people to read properly. Characters can be selected long to prevent reading with Segmentation Operations. Codes that come in the form of text validation can be placed randomly overlay. Due to these complexities, situations such as confusing uppercase letters with lowercase letters and indistinguishable from letters may occur. It is used to verify audio codes with audio instead of video. Noises are added to the sounds so that they are not detected by bots. The letters can be confused with each other because of the noises [19].

Segmentation and character recognition are applied to decode CAPTCHA codes. It consists of segmentation, pre-processing, image opening, labelling stages. In the Pre-Processing stage, the characters are revealed in grayscale. There may be horizontal and vertical lines in CAPTCHA codes to make it difficult to detect. To remove these lines, the inclusion of colour pixels in the image is examined. Median filtering is applied to reduce noises. The characters whose locations and sizes are determined by applying the histogram are approximated to each other. A feature extraction vector is created to recognize the characters [20]. In the segmentation phase, the dimensions of the characters and the places they are placed are important. The fact that the sizes of the characters are different, and their places are not separated from each other and in the same direction creates difficulties in the segmentation stage [21]. After the segmentation phase is completed, the character recognition phase is started. Three methods are used for optical character recognition (OCR), Pattern Matching (TR), Convolutional Neural Networks (CNN) character recognition. Pattern matching is pattern oriented and good at finding different characters. It is more robust than the other two methods since big data processing is required to train the CNN algorithm [3].

Zhang et al. explained how a CAPTCHA validator segmentation is performed in their study. To prevent CAPTCHA validators from being detected by bots, noises are added to images, letters or numbers are spaced, overlapped, or placed horizontally or vertically. During segmentation, the letters on the image are circled and the remaining parts are discarded. The characters are white and the parts outside the letter are black. Using the Hough algorithm, the characters are rendered white with a black background. Using the media filtering method, the noise in the image is removed. The letters in the image are zoomed in without changing their size. Character recognition segmentation is performed. In the study, segmentation with 95.2% was obtained [22].

Optical character recognition (OCR) Systems are systems developed to recognize letters in a picture. OCR has difficulty recognizing images with reduced quality or manuscript characters [18]. This system, which detects the letters on the image by scanning, uses many applications together to detect CAPTCHA verification codes. OCR can detect letters that are more emotional than the user. As a security measure, to prevent the detection of WEB services codes, the security is increased by requesting the IP address in case of making CAPTCHA codes difficult and incorrectly entered x times. Different studies have been carried out to prevent the detection of CAPTCHA images with Ohrid. Techniques of using coloured letters instead of black and white images, overwriting letters, adding noise, font tricks, and complex writing of meaningful words were applied [23]. With these techniques, OPR attacks can be prevented. Long letter combinations reduce detection time and probability. It should be noted that these measures also pose difficulties for users. Colour codes are an obstacle for colourblind users [7].

Chellapilla et al. Google and Yahoo have applied segmentation to validators. After the segmentation process, they performed character recognition by applying machine learning [24].

Table 1. Security and vulnerability detection systems of CAPTCHA models

Types	Security	Vulnerability Status	Vulnerability Detection Method
Arithmetic CAPTCHA	Arithmetic CAPTCHA presents a mathematical model to the user. The user can log in to the system by entering the answer of the mathematical operation in the box [6].	The mathematical model presented in the arithmetic CAPTCHA type usually comes in the form of four operations. Submitting a complex mathematical model may not be answered by every user. In general, the fact that 2-digit numbers come as codes also shows that the solution set is narrow [6].	Mathematical operations consist of numbers. After the segmentation process is applied to the confirmatory coding, the numbers can be easily detected in the HTML codes during the detection phase. Considering the CAPTCHA validators, it is among the weakest models [6].
Text-Based CAPTCHA	Text-Based CAPTCHA sends a text with a question pattern to the user. It prompts the user to enter the text's answer into the box. When the user enters the answer to the question correctly, he is allowed to enter the system [6, 25].	The questions that come as validators should be questions that every user can answer. For example, “What is the capital of Turkey?” questions arise. Text-based CAPTCHA validators are presented in plain text rather than hidden inside an image [6].	Authenticator spam presented in plain text can be easily detected by users. Spam user easily solves the answer to the question in the text and forwards it to the validator. Text-based CAPTCHA is seen as a weak method because it can be detected easily [6, 13].
Text CAPTCHA	If the text is of CAPTCHA type, a text consisting of random letter and number characters is presented. Each letter in the verifier stored in the image is randomly placed. They can be different in size, can be placed vertically or horizontally, and their spacing can be different [26].	Random placement of characters used in the text CAPTCHA validator makes it difficult for Bot users to detect it. However, sometimes characters may overlap, or letters and numbers may be confused with each other, while making it difficult to detect characters. This can make it difficult for the user to read [13, 26].	Although the text CAPTCHA makes the segmentation process difficult due to the addition of noise to the image, the filters in the image can be removed with the help of various filters, and the places outside the character can be whitened. They can be detected using OCR and CNN algorithms [3, 20, 21].
Game CAPTCHA	The game CAPTCHA type presents a user-predictable word. The word in the text can be different in size and can be placed vertically or horizontally [10, 11].	Italic placement of characters used in the game CAPTCHA validator makes it difficult for Bot users to detect it. However, characters are made difficult to detect, while sometimes superimposed characters make the word difficult to guess. For example, “G letter with 6 or B” or uppercase and lowercase letters can be mixed [7, 8, 26].	In the game CAPTCHA segmentation application, gray scale, median filtering to prevent noise in the image, histogram filters are used to determine the location of the characters. It can be detected by OCR, TR, or CNN algorithms during character recognition [3, 20, 21].
Split Text CAPTCHA	It likes Game CAPTCHA version. Words are Easily guessable. However, some letters of the split text are missing. Since the user can guess the word, he completes the missing letters and writes them in the box and is allowed to enter the system. Words are placed horizontally or vertical or spaced apart. The bot is difficult to detect by the user because noise is added to the image [9].	Since the letters in the split text are placed horizontally and vertically, there is a possibility of overlapping or confusion. The missing of some letters may not be predictable by every user. This is because the confirming word may be foreign, or the user may not have heard that word before.	Split text CAPTCHA is a version of total CAPTCHA [11]. The segmentation and character detection processes applied in the detection method of Total CAPTCHA can be solved by the bot user by applying the same. The split-text CAPTCHA is a reliable type of CAPTCHA, even if it poses difficulties to the user.

Pix CAPTCHA	This method makes use of images. Also known as picture captcha. Multiple images are presented to the user. The user is asked to select the pictures with the image or the name of the common character in the picture is asked to enter the box. Since noise is added to the images, it becomes difficult to detect [9, 12, 27].	Adding noise to Pix CAPTCHA type images may make it difficult for some users to select them. When he makes an incorrect or incomplete selection, he can be blocked from entering the system [9].	With the development of CNN algorithms, machines' classification and results have increased. The noise in the image can be removed by using different image processing techniques [28]. Images with applied image processing filters can be detected by bot users.
Voice CAPTCHA	The audio is sent in the form of CAPTCHA-type confirmatory text, not audio. The word or number is spoken, and the user is expected to enter what he hears. Noises are added to the sounds to prevent the sounds from being detected by the bot user [27].	Noises are added to prevent characters from being recognized in the Voice CAPTCHA application. Some letters may be difficult to select due to these added noises. Not every user may have a speaker system or may be a hearing-impaired user [5].	Voice-based CAPTCHA types have been shown to be easily crackable with the development of speech analysis and recognition applications [28].
Video CAPTCHA	Video CAPTCHA is like text CAPTCHA. Characters consisting of letters and numbers are placed vertically and horizontally in the text. The confirmer is in motion in the form of animation [27].	The animated characters in the video CAPTCHA may make it difficult for some users to detect them. This method is not preferred because the verification time is slow [27].	Since the verification of video CAPTCHA is like text CAPTCHA, it can be segmented with gray filtering, median filtering, and histogram filtering methods.
Google ReCAPTCHA	The ReCaptcha method emerged when words from old books were presented as confirmatory. A word registered in the system and images of words in old books that have not been registered before are presented to the user as a validator. The user must enter both known and unknown words correctly into the system. If a word is entered incorrectly, it is blocked from entering the system [6].	In case of using Google ReCAPTCHA texts, users are blocked from entering the system. By presenting two different words, the user is expected to enter both words correctly. The user may find it difficult to guess the word that has not been entered into the system before.	There are deep learning and machine learning algorithms in the structure of the Google ReCAPTCHA system. If the word is entered correctly by several users, it is saved in the system's memory. It can be detected by Google ReCAPTCHA OCR attacks [27].
Google No CAPTCHA	Google No CAPTCHA is verified by ticking a box. By ticking the "I am not a robot" box, validation is made and login to the system is faster [6].	Google no CAPTCHA confirms login to the system by asking the user to tick only one box. With the development of CNN architectures, Google No CAPTCHA bot users can log into the system.	When Google No CAPTCHA is examined, it provides access to my site without guessing the characters or solving a mathematical operation for the user to enter the system. In this way, authentication can be provided by each user.

Security, Vulnerability Situations and Vulnerability Detection Systems of CAPTCHA models Our motivation for the realization of this study is the limitations and vulnerabilities of the current CAPTCHA versions. If we list them.

- Solvability with OCR (Optical Character Recognition) technology,
- Simple designed CAPTCHAs can be easily resolved and bypassed by bots,
- Some CAPTCHAs become complex for visually impaired users,
- It can be bypassed by internet browsing tools,
- Some CAPTCHAs are difficult to understand or read by users.

This study aims to develop a novel, intelligent CAPTCHA model that integrates image processing techniques with a machine learning-based decision mechanism to enhance security while maintaining usability. The proposed CAPTCHA system employs:

- Randomized object placement and adaptive noise layers to resist AI-based solvers.
- User interaction-based validation, requiring users to actively select and mark objects within an image, making it harder for automated bots to bypass the system.
- A convolutional neural network (CNN)-based classifier that evaluates user interactions to distinguish between human users and automated attackers.
- Real-time tracking and response pattern analysis via a lightweight embedded software module, enhancing decision accuracy.

Extensive experimental evaluations demonstrate that the proposed model achieves 99.58% classification accuracy, significantly outperforming conventional CAPTCHA approaches. This research contributes to the development of next-generation CAPTCHA systems that balance security, usability, and AI resistance.

The proposed CAPTCHA system has significant implications for online security and user authentication. By combining interactive verification with AI-resistant image processing techniques, it reduces the risk of automated bot attacks, making unauthorized access significantly more difficult. Additionally, the system enhances user experience by minimizing frustration often caused by traditional CAPTCHAs that rely on distorted text or complex mathematical challenges. Its lightweight architecture makes it easier to integrate into websites and applications, ensuring seamless deployment without compromising system performance. This model can be particularly beneficial for financial institutions, e-commerce platforms, and secure login systems, where strong bot prevention mechanisms are crucial for maintaining data integrity and user privacy.

2. Proposed Model

In this section, it will be explained in detail how the proposed new, robust and image processing-based smart CAPTCHA model is prepared. To better understand the proposed system, the section is divided into parts within itself. In the study, a hybrid data set was used by adding the data set containing the original fish pictures that we obtained to the data set named "Natural Images" [29] in Kaggle.

The security of the CAPTCHA model proposed in this paper is enhanced to provide robustness against AI-based solvers. The term "secure" CAPTCHA specifically refers to the model's capacity to detect automated bot solvers and keep them out of the system. The proposed model offers effective protection against AI-assisted attacks on CAPTCHA solutions. The model includes verification methods that require complex image processing and user interaction that cannot be detected by AI algorithms. Thus, the background noise, random placement of objects, and time constraints of CAPTCHA images increase security by making it difficult for bots to circumvent the model.

2.1. Image Processing Based CAPTCHA Generator

The images selected randomly from the image pool by following certain numerical rules are brought to the same resolution. Pictures brought to the same size are placed randomly on the whole image. Upon completion of this process, the CAPTCHA image that will be presented to the user is completed. An example of the output of this process is shown in Figure 4. To enhance the robustness and security of CAPTCHA images against automated solvers, various advanced image processing techniques are employed. The CAPTCHA images are dynamically generated with randomized object placement, ensuring that each instance has a unique layout, preventing attackers from using positional heuristics for automated solving. To further obfuscate patterns, adaptive noise injection, including Gaussian noise is applied to disrupt AI-based edge detection and segmentation. Additionally, geometric transformations such as rotation, scaling, shearing, and perspective distortion make pattern recognition significantly more challenging. Edge perturbation and contour warping techniques are also integrated, where object boundaries are slightly altered using morphological operations like dilation and erosion, making shape-matching attacks ineffective. Colour-space manipulation and contrast variation are introduced by applying random colour filters, altering brightness levels, and modifying contrast, further increasing the difficulty for automated classification models.

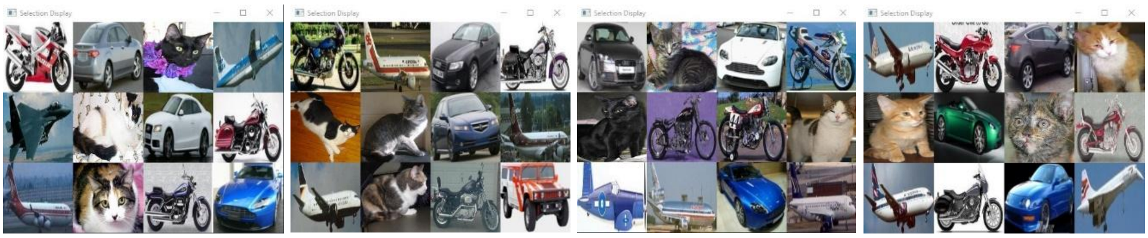

Fig.4. CAPTCHA example prepared with image processing algorithm

After the prepared CAPTCHA is presented to the user, the user is asked to make a real-time selection according to the category selection question specific to the CAPTCHA. Using the touch screen or the mouse hardware, the user draws the images requested in the question in such a way that they remain within the rectangle. An example image of the choices made by the user is shown in Figure 5.

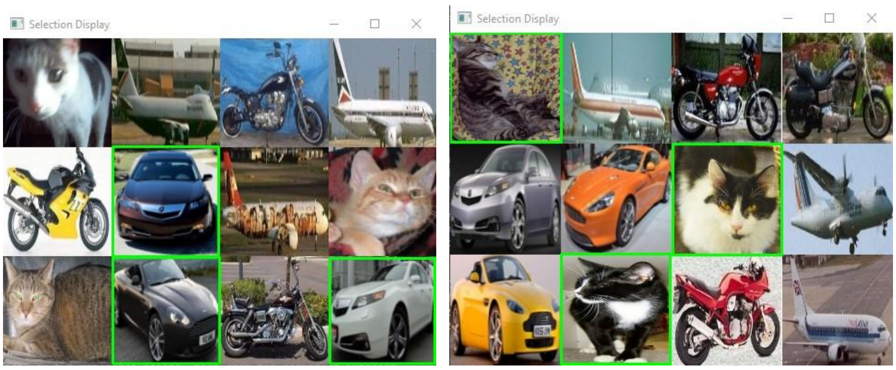

Fig.5. User's selection screen on CAPTCHA

Image processing techniques such as image resizing, randomization and noise addition are used to generate CAPTCHAs. First, the objects selected for the CAPTCHA images are dimensioned to provide a standard resolution. Then, the objects are placed in random locations to make the images harder to guess. Furthermore, noise layers added to CAPTCHA images increase the level of security by preventing AI-based solvers from correctly classifying the images. These techniques enable the creation of complex and reliable CAPTCHA images that can only be solved by real users.

2.2. Machine Learning Based Decision Model

To ensure accurate and robust evaluation of CAPTCHA responses, a convolutional neural network (CNN)-based classifier is employed to distinguish between human and bot interactions. The CNN architecture was chosen due to its strong capability in image recognition, feature extraction, and classification, making it highly effective for CAPTCHA verification tasks. The proposed model consists of multiple convolutional layers that apply 3×3 kernels to extract spatial features, followed by ReLU activation functions to introduce non-linearity and prevent vanishing gradients. Max pooling layers are incorporated to reduce the spatial dimensions and computational complexity while preserving essential features. The extracted feature maps are then flattened and passed through fully connected dense layers, leading to the final classification using the Softmax activation function, which assigns a probability score to each possible category.

To enhance model performance, categorical cross-entropy loss is used as the loss function, ensuring precise classification. The model is trained using the Adam optimizer, which dynamically adjusts learning rates to accelerate convergence while preventing overshooting. Training is performed on a dataset consisting of 6,000 labeled CAPTCHA images, divided into 80% training and 20% testing data. The dataset includes five object categories (airplanes, cars, cats, motorcycles, and fish), ensuring a diverse and challenging classification task. To further improve generalization, data augmentation techniques such as rotation, flipping, and random noise addition are applied during training.

The trained CNN model achieves 99.58% classification accuracy, with a validation accuracy of 96.15% and a loss value of 0.2078, demonstrating its robustness in differentiating human interactions from bot-generated attempts. Unlike traditional rule-based CAPTCHA evaluation methods, which rely on static heuristics, the proposed AI-driven evaluation mechanism dynamically adapts to new threats, making it significantly harder for automated solvers to bypass. The combination of deep learning-based classification, real-time user interaction tracking, and image processing-based CAPTCHA generation provides a highly secure and adaptive solution for modern cybersecurity challenges. The structure of the model used is shown in Figure 6.

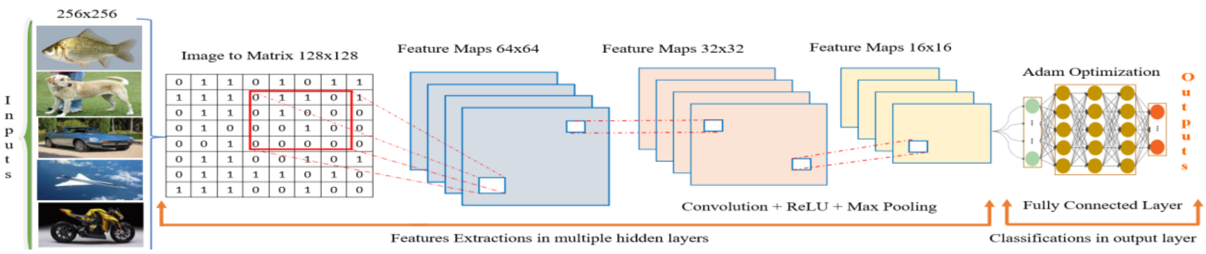

Fig.6. CNN model configuration and architecture

For the training of the model, 727 pictures of airplanes, 968 pictures of cars, 885 pictures of cats, 788 pictures of motorcycles and 600 pictures of fish were used. During the training, the data was used as 80% training data and 20% test data. The training was carried out with 10 epochs. The general architecture of the new CAPTCHA model proposed by the study and all the working principles to that end are shown in Figure 7.

Images containing 6000 different assets have been uploaded to the image pool. Data categories were created by dividing the uploaded images into their own asset groups. Among the created data categories, different numbers of pictures were taken from each category, except for three. The point to note here is that only three pictures are taken from one data category. The data category in which three images are selected will determine the image selections that will be requested from the user. The captured images were combined using image processing techniques and a CATCHA image was created for the user. The created CAPTCHA is presented to the user, and the user is asked to take the images of the data category asked in real time into a rectangle with the help of the touch screen or the help of the mouse. After this process is performed, the visual in the rectangle is extracted from the whole picture with image processing techniques and sent to the decision mechanism trained with the CNN algorithm with the data category. The model classifies the images that come to it and checks its compatibility with the data category sent to it.

In addition to this control, the selection time of the user, which we have placed in the image processing algorithm, is controlled on the CAPTCHA. The CAPTCHA visual is refreshed when the user is delayed in making the selections. Our third control method is the number of rights granted to the user to prevent detection by random or different image algorithms of bot users. If the user cannot make the right choices on the CAPTCHA image presented to him, the system will not give real user approval. Similarly, wrong image selections and long user selections are factors that hinder real user approval. If the user successfully selects the images presented by CAPTCHA in the questioned category, within a reasonable time, within the number of valid rights, a real user decision will be made, and access will be allowed.

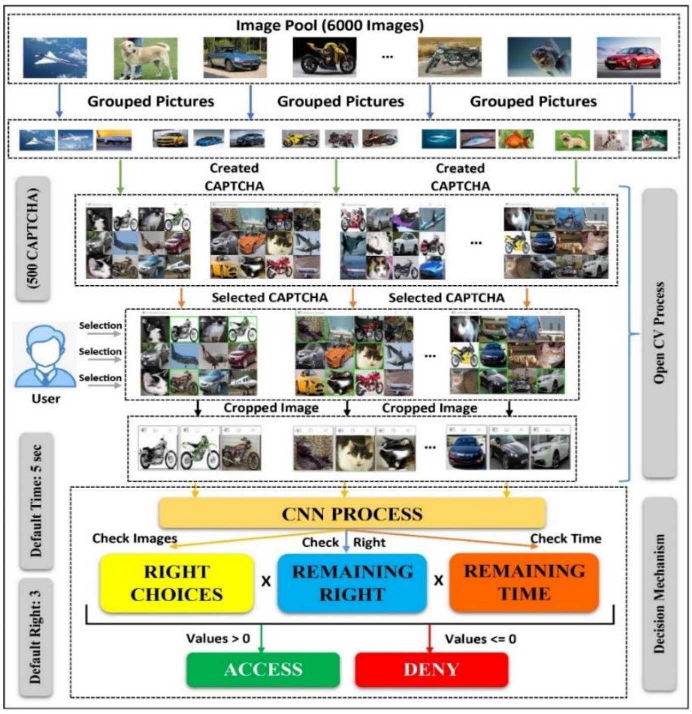

Fig.7. New CAPTCHA model architecture

Unlike traditional classification architectures, the CNN model used in the CAPTCHA solution process is optimized for security. In particular, the use of the MAX Pooling technique increases the accuracy of the CAPTCHA solution process while filtering out unnecessary data. The Softmax activation function used in the classification phase of the model facilitates the assignment of each object to the correct class and minimizes misclassifications. These structural innovations make CAPTCHA images harder for both users to solve and better identify real users

3. Findings

The results of the CNN algorithm training, which works as the decision mechanism of the proposed CAPTCHA model, will be included in this section. The correct classification success was measured as 96.15% at the end of 10 epoch training, which was carried out with the data set separated as 80% training and 20% test data. The loss value decreased to 0.2078 levels. The increasing success graph of the CNN model and the decreasing lost value graph during the training process are shown in Figure 8.

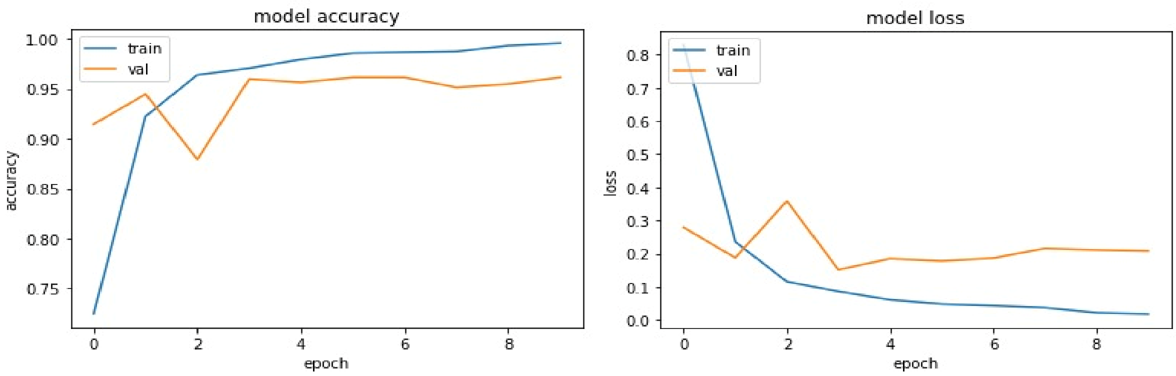

Fig.8. Accuracy and lost value graphs in the training of the CNN model

Table 2 shows the values of the metrics obtained according to the training results of the CNN model. By looking at the values in this table, it is seen that the trained model has passed a successful training, and the data set is suitable for training.

Table 2. Training results of machine learning algorithm

Metrics	Values
Loss	0.0179
Accuracy	0.9958
F1_m	0.9958
Precision_m	0.9958
Recall_m	0.9958
Val_loss	0.2078
Val_Accuracy	0.9615
Val_F1_m	0.9613
Val_Precision_m	0.9622
Val_Recall_m	0.9605

The representation of the correct and incorrect predictions of the model on the confusion matrix during the test process carried out at the end of the training is shown in Figure 9. When the confusion matrix is examined, it is striking that the trained model made the most mistakes in cat classification. The reason for this error is thought to be due to the complexity of the background images of the cat picture in the picture set in the cat class.

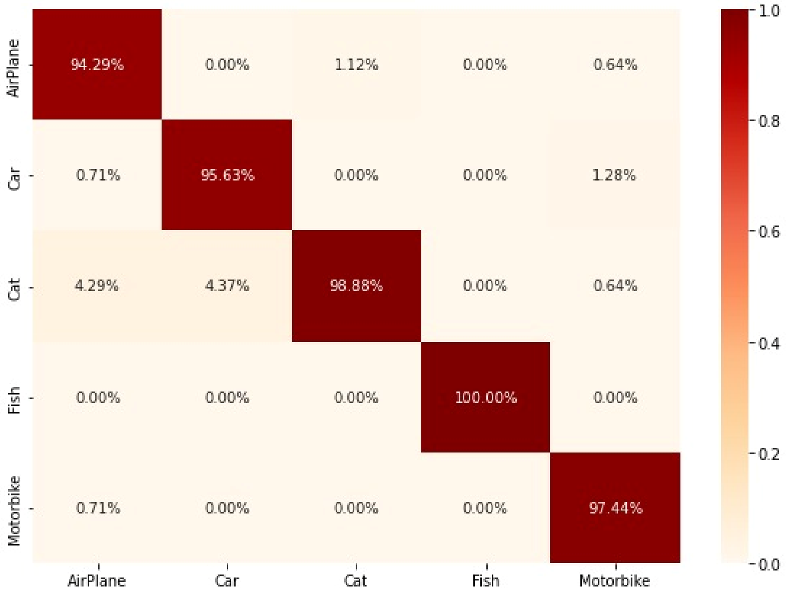

Fig.9. The confusion matrix of the trained model in the testing phase

To assess the effectiveness of the proposed CAPTCHA system, multiple quantitative evaluation metrics were used to measure its classification performance. The primary metric employed was accuracy, defined as the proportion of correctly classified CAPTCHA responses (both human and bot) relative to the total number of samples. The proposed model achieved an overall accuracy of 99.58%, indicating a high level of classification reliability. However, accuracy alone does not provide a complete understanding of the model's robustness, particularly in imbalanced datasets where false positives or false negatives may significantly impact security.

To further evaluate system performance, we computed the false positive rate (FPR), which measures the proportion of bot interactions mistakenly classified as human responses. A high FPR would indicate a security risk, as bots could bypass the CAPTCHA system. Our model yielded an FPR of 0.92%, demonstrating strong resilience against automated solvers. Similarly, the false negative rate (FNR), representing the percentage of human users misclassified as bots, was measured to ensure usability. The system recorded an FNR of 1.27%, confirming that the CAPTCHA remains user-friendly while maintaining strict security controls.

Additionally, precision, recall (sensitivity), and F1-score were calculated to provide a deeper insight into classification performance. Precision (99.32%) measures how many of the samples identified as human were actually human, reducing the likelihood of bots bypassing the system. Recall (98.71%) ensures that genuine users are not unfairly blocked. The F1-score, a harmonic mean of precision and recall, reached 99.01%, demonstrating a well-balanced performance between security and usability.

The ROC and AUC performance metrics are self-explanatory to make the results of the classification problems more meaningful. We interpret the ROC curves to select the appropriate threshold value. The AUC metric represents the area under the ROC curve graph. It takes a value between 0 and 1. The closer the AUC value is to 1, the more successful the trained model is interpreted. Figure 10 shows the graph of the ROC curve according to the AUC value and the AUC mean.

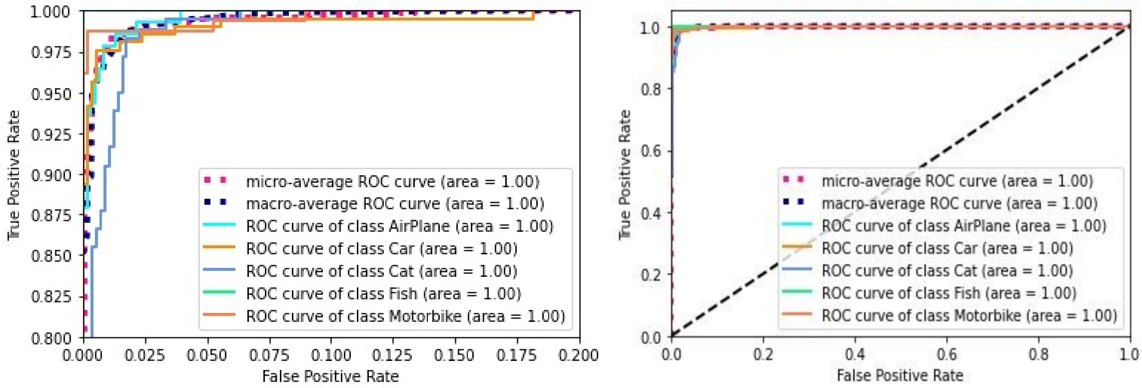

Fig.10. Graph of ROC curve versus AUC value and AUC means of the trained model

The low value in the success of the cat classification determined in the confusion matrix table compared to other classes is also striking in the ROC curve graph. Cat classification curve represented by dark blue colour differs from the curve of other classification groups. With this graph, the confusion matrix table shows parallel results, showing that the measurements obtained are healthy. Metrics such as True Positive Rate, False Positive Rate and AUC (area under the ROC curve) are used to evaluate the effectiveness of the proposed CAPTCHA model. The model shows high accuracy with a classification success of 96.15%. In addition, the performance of our model is evaluated by comparing it with common CAPTCHA types. Compared to traditional CAPTCHA models, our proposed model has higher robustness against bots and protection from AI-based solvers. The results of this analysis prove that our model offers a more secure option against CAPTCHA solutions.

As soon as the CAPTCHA image came in front of the user, the time elapsed until the user perceived the type of choice to be made in the CAPTCHA and made his first choice, the time he made his second drawing, and the time he spent until he made the third drawing were measured with the help of timers. These measurement values were obtained at the end of the application by 105 users and shown in Figure 11.

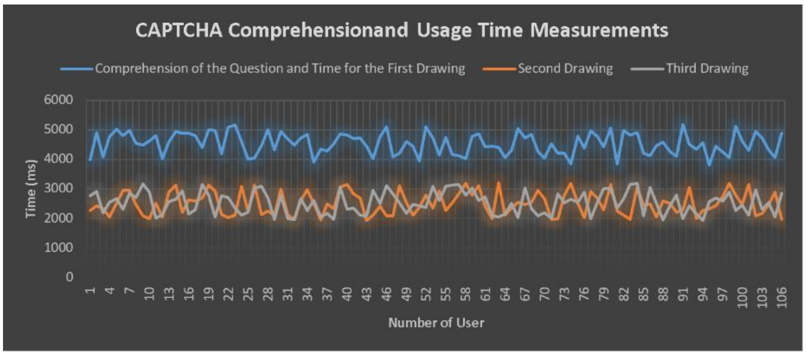

Fig.11. Graph of ROC curve versus AUC value and AUC means of the trained model

The user detection and drawing times shown in Figure 11 constitute the profile value data of the average user for the evaluation mechanism. These values obtained from this aspect are considered as important data that can reveal the difference between the real user and the bot.

In Figure 12, the time to check the accuracy of the CAPTCHA selections sent to the trained machine learning model was measured. An evaluation test of 25 CAPTCHA selections was carried out to create the graph shown in Figure 8. In these tests, it is seen that the model performs the initial evaluation process between 47 and 53 milliseconds. All other assessments take between 17 and 23 milliseconds.

The test-retest method was used to test the stability of the values shown in Figure 12. Thus, it has been determined that the model offers stable and stable results in decision-making behaviours.

Finally, response time analysis was conducted to evaluate system efficiency. The initial CAPTCHA evaluation process took between 47–53 milliseconds, while subsequent interactions were processed within 17–23 milliseconds, ensuring real-time usability without compromising security. These results indicate that the proposed system successfully differentiates between human users and bots while maintaining high accuracy, low misclassification rates, and fast processing times, making it a viable solution for real-world cybersecurity applications.

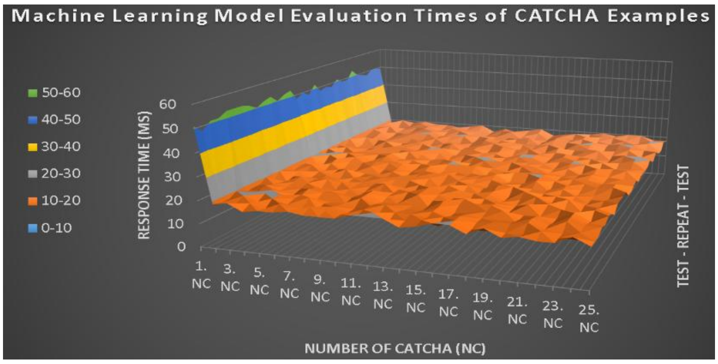

Fig.12. Machine learning model evaluation times of CATCHA examples

Table 3. Comparison with similar studies

References	Year	Algorithm	Metric
[30]	2016	CNN	Accuracy: 97.72%
[31]	2019	ResNet-GRU, CNN-RNN	Accuracy: 99.875%
[32]	2020	CNN	Accuracy: 98.06%
[33]	2021	CNN	Accuracy: 97.84%
[34]	2022	CNN	Accuracy: 97.5%
[35]	2023	CNN	Accuracy: 94.67%
This Work	**2025**	**CNN**	**Accuracy: 99.58%**

4. Conclusions

It is used for authentication purposes on CAPTCHA websites. CAPTCHA's job is to prevent websites from being hacked by bots, automated software, and other fake users, and to ensure security. CAPTCHA aims to prevent automated processes and increase security by requiring verification that the user is a genuine human user. In this study, a new, safe, and powerful CAPTCHA model is proposed by using the CNN algorithm, one of the image processing methods and machine learning methods. In the proposed model, image processing methods are used in the creation of CATPCHA images, while machine learning method and micro software are used cooperatively in the evaluation of user interaction processes with CAPTCHA. Another structure that makes the proposed model unique is that the user leaves the class clusters in the CAPTCHA image within the area created by the user on the image. This method eliminates artificial users by creating an interactive interaction between the real user and CAPTCHA. In the evaluation part of this study, the criteria to distinguish the real user from the automatic software, robot or bot user were measured and tests were carried out. The success rates obtained prove the reliability of the system. For the CAPTCHA model to be proposed in future studies, the security and authenticity of the system will be improved by adding audio inputs in addition to the visual infrastructure.

References

[1] A. S. El Ahmad, J. Yan, and L. Marshall, “The robustness of a new CAPTCHA,” Proceedings of the 3rd European Workshop on System Security, EUROSEC’10, pp. 36–41, 2010, doi: 10.1145/1752046.1752052.

[2] I. A. Althamary and E. S. M. El-Alfy, “A more secure scheme for CAPTCHA-based authentication in cloud environment,” ICIT 2017 - 8th International Conference on Information Technology, Proceedings, pp. 405–411, Oct. 2017, doi: 10.1109/ICITECH.2017.8080034.

[3] Y. Wang, Y. Huang, W. Zheng, Z. Zhou, D. Liu, and M. Lu, “Combining convolutional neural network and self-adaptive algorithm to defeat synthetic multi-digit text-based CAPTCHA,” Proceedings of the IEEE International Conference on Industrial Technology, pp. 980–985, Apr. 2017, doi: 10.1109/ICIT.2017.7915494.

[4] M. Tuncay GENÇOĞLU, E. Bahadir, T. Bilimler MYO, F. Üniversitesi, S. Güvenlik Bölümü, and A. Yesevi Üniversitesi TÜRTEP, “IPS Sistemlerde Yapay Zekânın Son Beş Yıldaki Gelişimi,” Bilgisayar Bilimleri, pp. 211–218, Oct. 2022, doi: 10.53070/BBD.1172803.

[5] J. Yan and A. S. El Ahmad, “A low-cost attack on a microsoft CAPTCHA,” Proceedings of the ACM Conference on Computer and Communications Security, pp. 543–554, 2008, doi: 10.1145/1455770.1455839.

[6] M. Baykara, F. Alniak, and K. Cinar, “Review and comparison of captcha approaches and a new captcha model,” 6th International Symposium on Digital Forensic and Security, ISDFS 2018 - Proceeding, vol. 2018-January, pp. 1–6, May 2018, doi: 10.1109/ISDFS.2018.8355316.

[7] A. Kolupaev and J. Ogijenko, “CAPTCHAs: Humans vs. bots,” IEEE Secur Priv, vol. 6, no. 1, pp. 68–70, Jan. 2008, doi: 10.1109/MSP.2008.6.

[8] A. A. Süzen, "UNI-CAPTCHA: A Novel Robust and Dynamic User-Non-Interaction CAPTCHA Model Based on Hybrid biLSTM+Softmax," Journal of Information Security and Applications, vol. 63, p. 103036, Dec. 2021, doi: 10.1016/J.JISA.2021.103036.

[9] C. Pope and K. Kaur, "Is it human or computer? Defending e-commerce with captchas," IT Prof, vol. 7, no. 2, pp. 43–49, Mar. 2005, doi: 10.1109/MITP.2005.37.

[10] M. Chew and H. S. Baird, "BaffleText: a human interactive proof," https://doi.org/10.1117/12.479682, vol. 5010, pp. 305–316, Jan. 2003, doi: 10.1117/12.479682.

[11] H. S. Baird and T. P. Riopka, "ScatterType: a reading CAPTCHA resistant to segmentation attack," https://doi.org/10.1117/12.587811, vol. 5676, pp. 197–207, Jan. 2005, doi: 10.1117/12.587811.

[12] W. Khalifa and A. Hasan, "A SURVEY OF CURRENT RESEARCH ON CAPTCHA," International Journal of Computer Science & Engineering Survey (IJCSES), vol. 7, no. 3, 2016, doi: 10.5121/ijcses.2016.7301.

[13] J. Chen, X. Luo, Y. Guo, Y. Zhang, and D. Gong, "A Survey on Breaking Technique of Text-Based CAPTCHA," Security and Communication Networks, vol. 2017, no. 1, p. 6898617, Jan. 2017, doi: 10.1155/2017/6898617.

[14] M. Kumar, M. K. Jindal, and M. Kumar, "Design of innovative CAPTCHA for hindi language," Neural Comput Appl, vol. 34, no. 6, pp. 4957–4992, Mar. 2022, doi: 10.1007/S00521-021-06686-0/FIGURES/45.

[15] M. I. Hossen, Y. Tu, M. F. Rabby, M. N. Islam, H. Cao, and X. Hei, An Object Detection based Solver for {Google's} Image {reCAPTCHA} v2. 2020.

[16] H. Gao, F. Cao, and P. Zhang, "Annulus: A novel image-based CAPTCHA scheme," IEEE Region 10 Annual International Conference, Proceedings/TENCON, pp. 464–467, Feb. 2017, doi: 10.1109/TENCON.2016.7848042.

[17] S. Kulkarni and H. S. Fadewar, "Pedometric CAPTCHA for mobile Internet users," RTEICT 2017 - 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, Proceedings, vol. 2018-January, pp. 600–604, Jul. 2017, doi: 10.1109/RTEICT.2017.8256667.

[18] M. Shirali-Shahreza and S. Shirali-Shahreza, "Drawing CAPTCHA," Proceedings of the International Conference on Information Technology Interfaces, ITI, pp. 475–480, 2006, doi: 10.1109/ITI.2006.1708527.

[19] J. Yan and A. S. El Ahmad, "Usability of CAPTCHAs or usability issues in CAPTCHA design," SOUPS 2008 - Proceedings of the 4th Symposium on Usable Privacy and Security, pp. 44–55, 2008, doi: 10.1145/1408664.1408671.

[20] S. Y. Huang, Y. K. Lee, G. Bell, and Z. H. Ou, "An efficient segmentation algorithm for CAPTCHAs with line cluttering and character warping," Multimed Tools Appl, vol. 48, no. 2, pp. 267–289, Jun. 2010, doi: 10.1007/S11042-009-0341-5/FIGURES/24.

[21] J. Yan and A. S. El Ahmad, "Captcha robustness: A security engineering perspective," Computer (Long Beach Calif), vol. 44, no. 2, pp. 54–60, Feb. 2011, doi: 10.1109/MC.2010.275.

[22] L. Zhang, Y. Xie, X. Luan, and J. He, "Captcha automatic segmentation and recognition based on improved vertical projection," 2017 9th IEEE International Conference on Communication Software and Networks, ICCSN 2017, vol. 2017-January, pp. 1167–1172, Dec. 2017, doi: 10.1109/ICCSN.2017.8230294.

[23] H. S. Baird, A. L. Coates, and R. J. Fateman, "PessimalPrint: a reverse Turing test," International Journal on Document Analysis and Recognition 2003 5:2, vol. 5, no. 2, pp. 158–163, Apr. 2003, doi: 10.1007/S10032-002-0089-1.

[24] K. Chellapilla and P. Y. Simard, "Using Machine Learning to Break Visual Human Interaction Proofs (HIPs)," Adv Neural Inf Process Syst, vol. 17, 2004, Accessed: Mar. 07, 2025. [Online]. Available: www.google.com,

[25] P. Wang, H. Gao, X. Guo, C. Xiao, F. Qi, and Z. Yan, "An Experimental Investigation of Text-based CAPTCHA Attacks and Their Robustness," ACM Comput Surv, vol. 55, no. 9, Jan. 2023, doi: 10.1145/3559754.

[26] H. Çakır and N. Uluhatun, "Robust security against cyber threats with variety of captcha<p>Güvenlik kodu çeşitliliği ile siber tehditlere karşı güçlü güvenliğin sağlanması," Journal of Human Sciences, vol. 15, no. 3, pp. 1414–1430, Jul. 2018, Accessed: Mar. 07, 2025. [Online]. Available: https://www.j-humansciences.com/ojs/index.php/IJHS/article/view/5201

[27] S. S. Rane and N. Shekokar, "3D Socio Graphical CAPTCHA based on reCAPTCHA against spammers and bots," 2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology, ICAECCT 2016, pp. 372–377, Jul. 2016, doi: 10.1109/ICAECCT.2016.7942616.

[28] D. Brodić, A. Amelio, and R. Janković, "Exploring the influence of CAPTCHA types to the users response time by statistical analysis," Multimed Tools Appl, vol. 77, no. 10, pp. 12293–12329, May 2018, doi: 10.1007/S11042-017-4883-7/TABLES/13.

[29] P. Roy, S. Ghosh, S. Bhattacharya, and U. Pal, "Effects of Degradations on Deep Neural Network Architectures," Jul. 2018, Accessed: Mar. 07, 2025. [Online]. Available: https://arxiv.org/abs/1807.10108v6

[30] Y. Lv, F. Cai, D. Lin, and D. Cao, "Chinese character CAPTCHA recognition based on convolution neural network," 2016 IEEE Congress on Evolutionary Computation, CEC 2016, pp. 4854–4859, Nov. 2016, doi: 10.1109/CEC.2016.7744412.

[31] Y. Shu and Y. Xu, "End-to-End Captcha Recognition Using Deep CNN-RNN Network," Proceedings of 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference, IMCEC 2019, pp. 54–58, Oct. 2019, doi: 10.1109/IMCEC46724.2019.8983895.

[32] A. Thobhani, M. Gao, A. Hawbani, S. T. M. Ali, and A. Abdussalam, "CAPTCHA Recognition Using Deep Learning with Attached Binary Images," Electronics 2020, Vol. 9, Page 1522, vol. 9, no. 9, p. 1522, Sep. 2020, doi: 10.3390/ELECTRONICS9091522.

[33] Z. Wang and P. Shi, "CAPTCHA Recognition Method Based on CNN with Focal Loss," Complexity, vol. 2021, no. 1, p. 6641329, Jan. 2021, doi: 10.1155/2021/6641329.

[34] Y. Chen, X. Luo, S. Xu, and R. Chen, "CaptchaGG: A linear graphical CAPTCHA recognition model based on CNN and RNN," Proceedings - 2022 9th International Conference on Digital Home, ICDH 2022, pp. 175–180, 2022, doi: 10.1109/ICDH57206.2022.00034.

[35] S. Sinha and M. I. Surve, "CAPTCHA Recognition and Analysis Using Custom Based CNN Model-Capsecure," Proceedings of the 2023 International Conference on Emerging Techniques in Computational Intelligence, ICETCI 2023, pp. 244–250, 2023, doi: 10.1109/ICETCI58599.2023.10331187.

Authors' Profiles

Dr. Remzi GÜRFİDAN is working as a Ph.D. of cyber security application and researching center at Isparta University of Applied Sciences. He completed his master's degree in electronics and computer department. He has program development experience in cyber security, deep learning and software development. He has 2 books in the field of blockchain technologies. He has been editor-in-chief of the journal in two international journals in the field of engineering.

Lecturer Oğuzhan Kilim has been conducting research studies on software development, image processing and image algorithms as a professional since 2013. He completed his master's degree in electronics and computer department. The author continues his research on pre-processing methods and optimisation processes used in image algorithms.

Prof. Dr. Tuncay YİĞİT is the head of the Department of Computer Engineering at Süleyman Demirel University. He is the chairman of the board of the Science Development Association. He has made important contributions to the literature around the world with his work in the field of artificial intelligence and optimization algorithms. He is the organizer and chief responsible of the ICAIAME international congress

How to cite this paper: Remzi GÜRFİDAN, Oğuzhan KİLİM, Tuncay YİĞİT, "Robust, Interactive, and Intelligent Captcha Model Based on Image Processing", International Journal of Intelligent Systems and Applications(IJISA), Vol.17, No.3, pp.1-14, 2025. DOI:10.5815/ijisa.2025.03.01