Journal of Artificial Intelligence and Technology, 2023, 3, 126-133
https://doi.org/10.37965/jait.2023.0216

ISTP
RESEARCH ARTICLE

A Web Authentication Biometric 3D Animated CAPTCHA System Using Artificial Intelligence and Machine Learning Approach

Neha Pradyumna Bora and Dinesh Chandra Jain
Department of Computer Science and Engineering, Oriental University, Indore, India

(Received 16 March 2023; Revised 30 March 2023; Accepted 02 April 2023; Published online 12 May 2023)

Abstract: The Internet and web security are integral aspects of our daily lives. Many commercial firms provide clients with Internet services. For web access, it is assumed that only the genuine user, who is a human, will register. Yet automated hacking programs can also do registrations with fake data that consume a lot of bandwidth, slowing down or occasionally even shutting down websites, leading to Distributed denial-of-service attacks. Completely Automated Public Turing test to tell Computers and Human Apart (CAPTCHA) is the solution. Complex CAPTCHA is challenging for humans to recognize, but simple CAPTCHA is simple for AI to decipher. With the developments in neural networks and machine learning, bots are mimicking humans, and it is becoming difficult to distinguish humans and bots apart. This generated a need to think of some more innovative and novel CAPTCHA. Now, utilizing the same AIML approach to increase the efficacy of CAPTCHA and make it stronger against the bot attack. Biometric 3D Animated Algorithm proposed in this research is a novel approach based on the Face Detection AI algorithm along with handwritten 3D animated characters selected randomly to create a string which makes CAPTCHA simple that humans can identify but very difficult for bots. The test results have proven this to be a very robust CAPTCHA. The machine learning algorithm employed will keep on increasing the efficacy of this CAPTCHA each time the bot tries to break this.

Keywords: artificial intelligence; B3DA; CAPTCHA; DDOS; image recognition; OCR

I. INTRODUCTION

The Internet plays such a significant role in our daily lives; Internet security should be a top concern for all users. Numerous businesses and organizations provide their customer with Internet support. Yet occasionally, harmful automated hacking software tries to target websites to slow down the server. Users are frequently required to submit personal information such as their email address, phone number, and address when signing up or filling out registration forms. However, automated hacking tools might cause the website to lag or even crash by filling it with false information from fictitious users. It was always anticipated that this work would be completed correctly and honestly by a real user means humans.

In order to access the website's resources and generate traffic, a program automatically fills out a form with false, inaccurate information, wasting a lot of disc space and making the server extremely slow. For this objective, several undesirable false accounts are generated. These attacks are typically carried out using computer programs [1]. For instance, on university websites during the announcement of results, any computer program generating enrollment numbers sequentially can open the result file of every student. Ultimately, this activity jams the server, making it difficult for real students to see their own results. Another example of a railway reservation website is one where a hacker can buy numerous Tatkal tickets with the aid of automated hacking software while making it difficult for a regular individual to obtain the tickets.

The security systems should operate dynamically against them to meet these restrictions. For answers to these issues, CAPTCHA is employed to distinguish between human and computer users. The websites use CAPTCHA as protection against such attacks. CAPTCHA stands for Completely Automated Public Turing test to tell Computers and Humans Apart. Most websites use it to protect against the non-human activity. The way CAPTCHA works prevents computer programs and bots from answering questions that people can quickly and easily answer. Simple text CAPTCHAs can be cracked by clever AI and image recognition algorithms, but even complex text CAPTCHAs with significant distortion are undetectable by humans. We provide a new method for implementing CAPTCHA that addresses security concerns using biometric 3D animated CAPTCHA. The new CAPTCHA that is being proposed should be simple for people to solve and type but impossible for computer programs or automated software to detect and decipher.

CAPTCHA is often divided into the following Category

A. Text-Based CAPTCHA

With a text-based CAPTCHA, the programmer inserts distortion between a series of text, such as letters or digits, before displaying them on the website (Fig. 1).

Basic OCR-based CAPTCHA: This CAPTCHA was unable to recognize reading low-quality printed words (Fig. 2).

Limitations: Basic OCR-based CAPTCHA can be cracked by smart AI technology and image recognition algorithms easily [2].

Complex OCR-based CAPTCHA: To solve the above simple OCR problem programmer, add extra noise into text, in order to make it more complex in front of attackers on a website (Fig. 3).

Limitations: However, because of these changes, human users find it considerably harder to recognize the words, and occasionally, users must type this difficult CAPTCHA more than 3/4 times, which is nothing but a time waster.

Corresponding author: Neha Bora (e-mail: muthaneha88@gmail.com).

© The Author(s) 2023. This is an open access article published under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

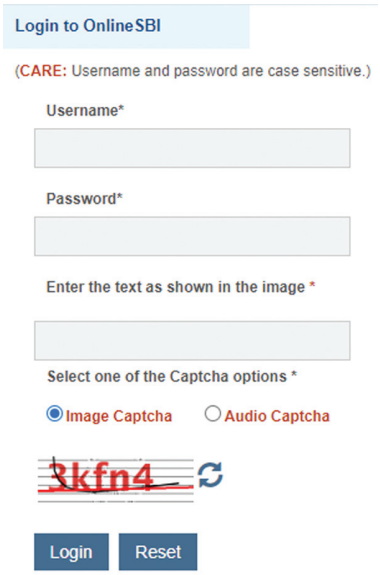

Fig. 1. Text-based CAPTCHA in SBI Bank.

Fig. 2. Basic OCR-based CAPTCHA.

Fig. 3. Complex OCR-based CAPTCHA.

B. IMAGE RECOGNITION CAPTCHA

Image recognition/Friend Recognition/ Human face recognition/ Avatar CAPTCHA are all we can categorize in Image Recognition CAPTCHA [3,4] (Fig. 4).

Limitations: In these image-based CAPTCHAs, a brute force attack is possible if the database is too tiny to hold the images. For storage, these CAPTCHAs require a lot of room. For one CAPTCHA display test, the database stores about 9+ images, which complicates the use of space. Users really hate having to constantly scroll down and up the form because it takes up more screen space than a conventional text CAPTCHA. The visually handicapped person likewise has no chance of completing an image-based CAPTCHA. Accept form submission by clicking the "Submit" button. The same kinds of problems also exist with friend recognition, human face recognition, and avatar CAPTCHA.

C. AUDIO-BASED CAPTCHA

Those who are physically unfit and have some problem with eyesight can solve auditable CAPTCHA. Audio-based

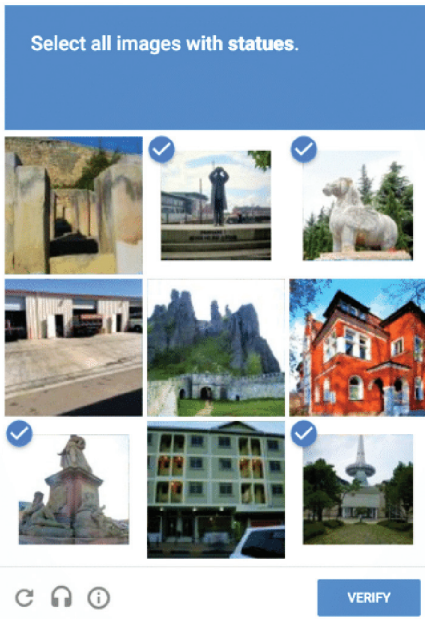

Fig. 4. Image-based CAPTCHA.

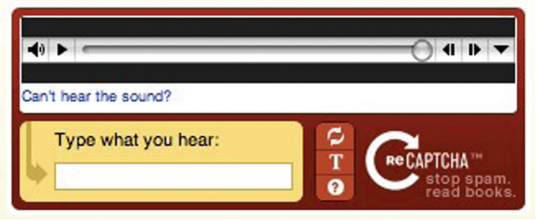

Fig. 5. Audio-based CAPTCHA.

CAPTCHAs ask users to type a CAPTCHA word that was played [5] (Fig. 5).

Limitations: Usually audios are difficult to understand due to improper pronunciation and noise in the background. Rhyming words confuse the user like good/god or to/two, etc. Sometimes unfamiliar English words are hard to understand. Sometimes unfamiliar English words are hard to understand.

D. GAME/PUZZLE-BASED CAPTCHA

In finger-guessing games, users must select the gesture that can win. Three basic gesture rules are important in this, and these are simple rules of the game Rock, Paper, Scissors "a rock beats a pair of scissors, scissors beat a sheet of paper, and paper beats a rock" [6] (Figs. 6 and 7).

Limitations: Require intelligence to solve. The game rule should be known to every user. The drawback of image CAPTCHA remains the same, that is, large solving time, occupying more space on a page, more loading time, etc. [7,8].

E. reCAPTCHA

reCAPTCHA service offered by Google to protect websites from spam and attacks.

Fig. 6. Puzzle CAPTCHA.

But it was also broken with the help of IOT devices [9].

F. NLP CAPTCHA

NLP (Natural Language Processing) CAPTCHA is a CAPTCHA-based digital advertising platform. The IRCTC (Indian Railway) Website uses this NLP CAPTCHA which was designed by Simpli5D [10,11] (Fig. 8).

Limitations: The main purpose of testing is that we are humans that were violated and advertisement-related words is having meaning and can be easily breakable with Dictionary and Brute Force attacks.

To stop automated mail account registration, the first CAPTCHA was created in 2000 by Luis von Ahn, Manuel Blum, Nicholas Hopper, and John Langford at Carnegie Mellon University for the Yahoo website [1].

A brief review of the work already done in the field:
K. Sukhani, S. Sawant, S. Maniar, and R. Pawar [12] discussed how to break the image so we were able to bypass the re-CAPTCHA v2.

M. Jadhav, N. Kulkarni, and O. Walhekar [9] suggested CAPTCHA only for visually impaired users.

Shivani and R. K. Challa [5] studied CAPTCHA: A Systematic Review on different CAPTCHA techniques.

Y. Zhang, H. Gao, G. Pei, S. Luo, G. Chang, and N. Cheng [13] suggested that deep learning is a tool for increasing the security of the CAPTCHA. In this, some information which was hidden, such as time, speed, track, etc., is useful to distinguish humans and computers

Y. S. Aljarbou [7] identified that a lot of time is consumed due to Puzzle CAPTCHA. Video-based CAPTCHA needs a high

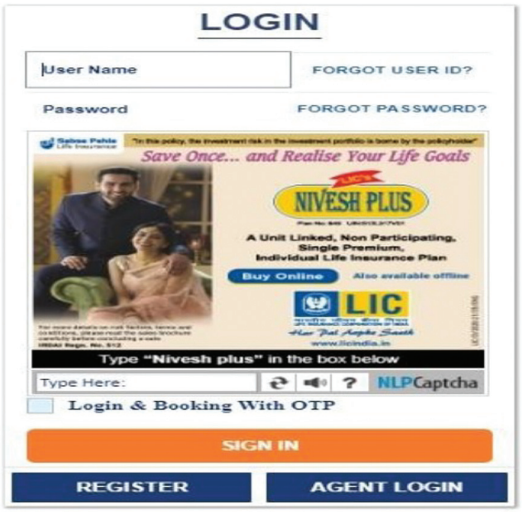

Fig. 8. NLP CAPTCHA.

Internet speed, and Audio-based requires users to understand the language. Large databases are used by Image-based CAPTCHAs. The author suggested Face and Heat scanning techniques; however, it still is costly way.

Khawandi, Shadi & Ismail, Anis & Abdallah, and Firas [14] have studied OCR & Non-OCR methods to break CAPTCHA. The author proposed that the design of robust CAPTCHA is difficult but more difficult is that it should not annoy the user.

Sheheryar, M.A. & Mishra, P.K. & Sahoo, and Ashok [15] concluded that with the advent of AI more CAPTCHA schemes will break in the future.

Cao Lei [6] has proposed a finger-guessing game as CAPTCHA considering its secondary logic judgment which is difficult for machines and easy for humans.

S. Singhal, A. Sharma, S. Garg, and N. Jatana [16] broke CAPTCHA from India's most visited e-ticketing website irctc.co.in.

C. J. Chen, Y. W. Wang, and W. P. Fang proposed in the paper that there were many noisy points and lines in the testing image CAPTCHA, and targeted numbers could be overlapped/disconnected by the noise and their breaking techniques.

Ali *et al.* [8] using image CAPTCHA developed a puzzle-based CAPTCHA system. Authors used tools like JavaScript, JQuery, HTML (Hyper Text Markup Language), and CSS (Cascading Style Sheets)

Azad and Jain [17] showed possible attacks on text CAPTCHAs. By adding the distortion and noise with a certain

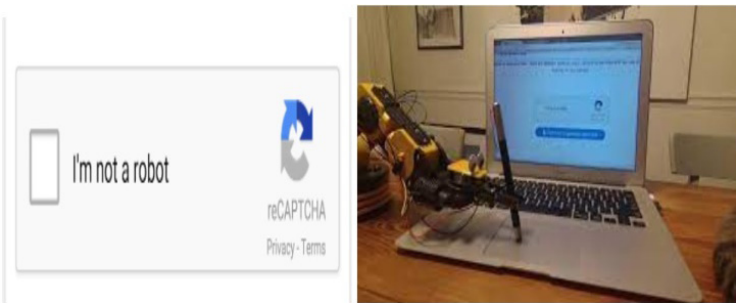

Fig. 7. reCAPTCHA and breaking method.

limit and arranging and rearranging characters, it could be read by humans and increases the security of text-based CAPTCHAs.

Neha Chandrakant Mutha and Dr. Samidha D. Sharma [18] introduce 3D animated handwritten CAPTCHA but there was a scope for improvement by adding biometric features to make it more robust.

Rao, Mukta & Singh, and Nipur [19] identified that the bots have now become intelligent enough to crack through machine-printed CAPTCHAs. They covered the drawbacks of other CAPTCHA; Handwritten CAPTCHA images could be the solution. The author achieved an average success of more than 80% in incorrect recognition of handwritten text by different OCR methods to break the CAPTCHA.

G. Goswami, R. Singh, M. Vatsa, B. Powell, and A. Noore [20] proposed an algorithm that generated CAPTCHA that offered better human accuracy and lower attack rates compared to existing approaches.

D. D'Souza, P. C. Polina, and R. V. Yampolskiy [4] introduced AVATAR CAPTCHA, an image-based approach to distinguish human users from bots; however, it was highly time-consuming.

J. Cui, J. Mei, W. Zhang, X. Wang, and D. Zhang [21-23] introduced 3D animated moving CAPTCHA, but they accepted that current methods of detecting moving objects still had defects and scope for improvement.

Chew, Monica & Tygar, J [24] proposed an image-based CAPTCHA where Image-based CAPTCHAs need large databases.

L. von Ahn, M. Blum, and J. Langford [1] proposed a text-based CAPTCHA to discriminate between incoming requests from humans and computers based on hard AI problems. Since 2004, CAPTCHA played an important role in artificial intelligence and cryptography.

Fig. 9. Few successful & unsuccessful attempts of breaking CAPTCHA by bots/programs.

https://www.google.com/recaptcha/about/ [25]: This website's reCAPTCHA creation was shown, but using AI and IOT this reCAPTCHA failed to protect the website.

Many [10,11,26] websites use different CAPTCHA like Text, Audio, NLP, etc.

In a general review of a search engine/NDTV, it is found that

- 200 million CAPTCHAS are solved a day.
- Roughly 10 secs of spent for each.
- 150,000 hrs. of work each day

First Breakage

EZ-Gimpy CAPTCHA was broken in 2003 by Greg Mori and Jitendra Malik using object recognition techniques and dictionary crosschecking.

Their program correctly interprets this CAPTCHA 93 % of the time and incorrect recognition is 7 % only.

The CAPTHCA implementation on Yahoo Mail's login website has been defeated by a Russian research group. Microsoft live mail has also been captured by junk [2] (Fig. 9).

By the observation of the survey, we can say that with the fast development of AI, bots, and image recognition techniques, the cracking problem is increasing with simple CAPTCHA. If we make it more complicated CAPTCHA, then it is even more difficult for humans to recognize. By considering all the constraints into consideration, the need for new CAPTCHA is generated.

II. OBJECTIVE AND METHODOLOGY

A. OBJECTIVE

- To implement a very challenging CAPTCHA so that bots cannot read.
- To develop the human-friendly CAPTCHA.
- To reduce human time consumed in the web authentication process.
- To develop a fast and secure system to distinguish between human and computer programs.

B. METHODOLOGY

Biometric 3D Animated (B3DA) Algorithm proposed CAPTCHA creation is based on some techniques such as:

$$F(n) : H(n) \vee G(n)$$

where

$F(n)$: CAPTCHA function

$H(n)$: Human Face Capturing

$G(n)$: B3DA Algorithm

$$G(n) : q \wedge r \wedge s$$

q: Handwritten 3D effect Characters

r: Animation

s: Display Technique

B3DA CAPTCHA is a very strong system itself to resist bots/programs attack in systems without a camera. However, to further make it more secure, using a camera in sectors like banking, defence, and reservation systems for trains, which when combined with human face detection, improves security by 2X.

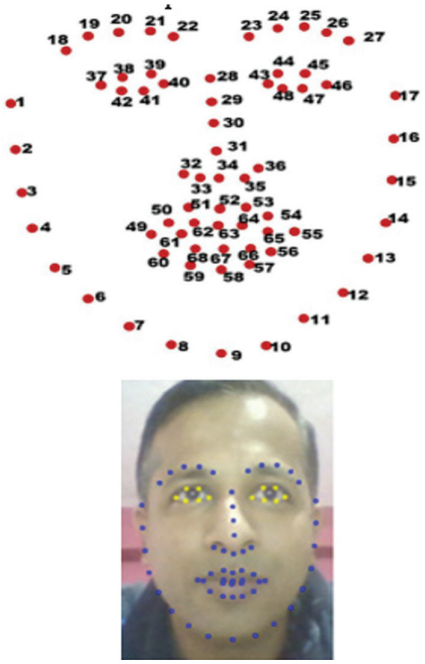

Fig. 10. 68 landmarks of face.

C. IMPLEMENTATION METHODS FOR HUMAN FACE RECOGNITION

Face recognition CAPTCHAs use facial recognition technology to verify whether a user is a human or a bot. They are designed to prevent automated spam and abuse by requiring users to identify and match human faces in a set of images.

There are several technologies that can be used for face recognition CAPTCHAs, including the following:

1. **Computer vision algorithms**: These algorithms use machine learning techniques to analyze facial features and recognize faces. They can be trained on large datasets of human faces to improve their accuracy.
2. **Facial landmarks detection:** This technique involves detecting key points on a face, such as the corners of the eyes, nose, and mouth, and using them to identify a user. It can be combined with other techniques, such as machine learning, to improve accuracy.
3. **Live detection**: This involves requiring users to perform specific actions, such as blinking or smiling, to prove that they are human.

Facial landmarks detection technology is often used in face recognition systems to help identify individuals, as each person's facial landmarks are unique (Fig. 10).

Compared to more general computer vision algorithms, Facial landmarks detection technology can also be used in conjunction with live detection to provide more accurate and reliable results. This is because it focuses specifically on the features of the face that are most important for identification, rather than trying to analyze the entire image.

In 68 landmarks of face points, 36 to 48 are used for eye prediction. Using these parameters, we can calculate whether the human eyes are closed or open. Bots cannot use still image/ photographs of human to pretend the presence of human. Human relay attack was also prevented (Figs. 11 and 12).

D. B3DA ALGORITHM

Step 1:

Save handwritten characters that are biometric in 3D to a database. Through biometric devices like pen tablets, these characters are manually produced in a variety of patterns with 3D effects that take depth into account. All the various images of the same character are then stored with one index using the store image's sub-index (Fig. 13).

```
pseudocode:
captcha_text = []
for i in range(MAX_CAPTCHA):
    c = random.choice(number)
    captcha_text.append(c)
    print(captcha_text)
captcha_text = ''.join(captcha_text)
print(captcha_text)
```

Step 2:

Checking by your own sight, like the conventional approach of checking doorstep people, is a straightforward way to determine

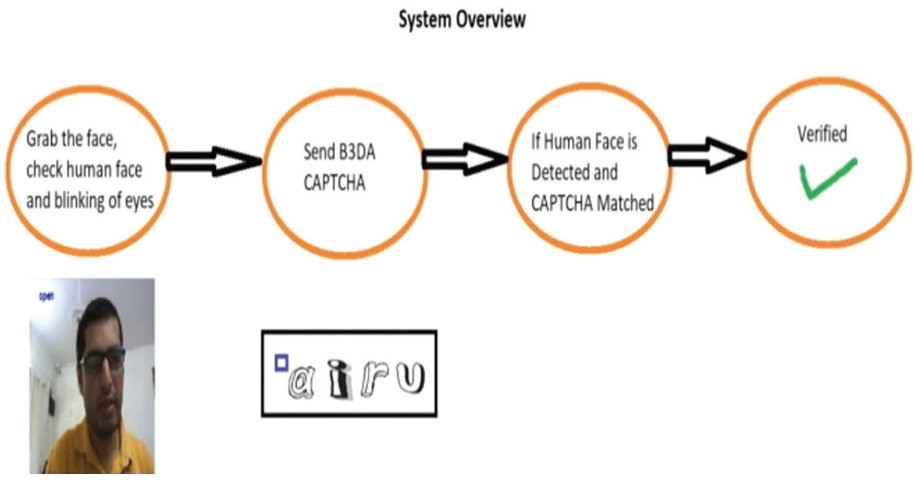

Fig. 11. System overview.

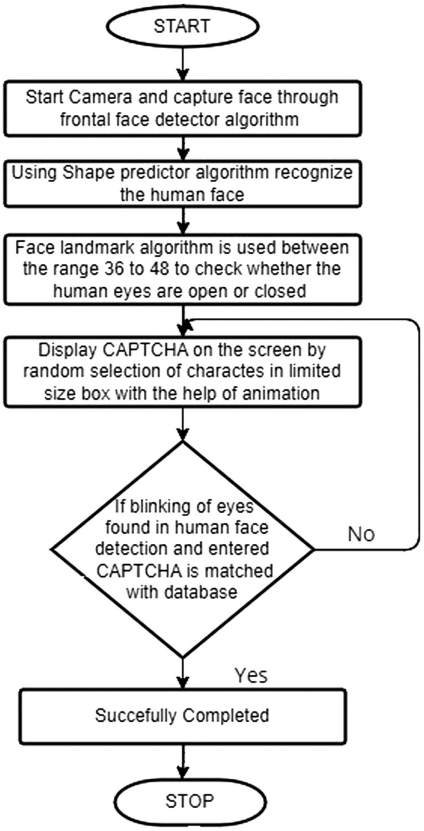

Fig. 12. B3DA algorithm.

whether the user is a human or a bot. Set up the camera in the usual manner and use the frontal face detector algorithm to record faces (Fig. 14).

```
pseudocode:
self.cap = cv2.VideoCapture(0)
   self.hog_face_detector =
dlib.get_frontal_face_detector()
   self.dlib_facelandmark =
dlib.shape_predictor("shape_predictor_68_face_
landmarks.dat")
```

Step 3:

In the next step using the Shape predictor algorithm, we recognize if the frontal face is a single human face or not and using face landmark, we mark the human face part.

```
pseudocode:
face_landmarks =
self.dlib_facelandmark(gray,face)
   str="human face detected"
```

Step 4:

The face landmark algorithm is used between the ranges 36 to 48 to check whether the human eyes are open or closed. By default,

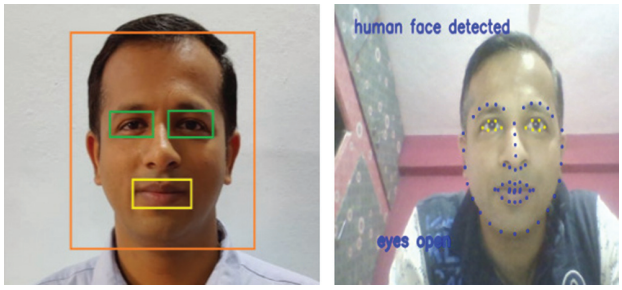

Fig. 14. Human face detection.

we consider that eyes are opened then if the user is a genuine human, not a still image, then he will blink the eyes at least once. We can capture this by checking the face landmarks differences of the y parameter of eyes.

```
pseudocode:
for n in range(36,48):
      x = face_landmarks.part(n).x
      y = face_landmarks.part(n).y
      cv2.circle(frame, (x, y), 1, (0,
      255, 255), 2)
      t1 = face_landmarks.part(37).y
      t2 = face_landmarks.part(41).y
      t5 = face_landmarks.part(43).y
      t6 = face_landmarks.part(47).y
      diff=t2-t1
      diff2=t6-t5
      eyes="eyes open"
      if self.myflag1==1:
         self.myflag2=1
      if diff<7 or diff2<7:
         eyes="eyes close"
         self.myflag1=1
```

Step 5:

By random variable generator algorithm selection method, we can generate random CAPTCHA on screen. Display CAPTCHA on the screen by random selection of characters in a limited-size box with the help of animation select 4-5 characters from the database and give a wakeup symbol before the first character for recognition.

Animated frames differ with each respect to their size, font, scale, slanting effect, pixel intensity, etc.

In moving animated CAPTCHA, an image is not displayed as a complete whole image to the user in the first iteration. Any software/automated program, hence, cannot be able to shoot the CAPTCHA, even if it takes the shot, the full image is not shown on screen, and hence, the result of both will not match (Fig. 15).

```
pseudocode:
captcha_text = []
for i in range(MAX_CAPTCHA):
   c = random.choice(number)
```

Fig. 13. Biometric-generated character.

Fig. 15. B3DA CAPTCHA

Handwritten CAPTCHA Result

Online Tool	OCR-1	OCR-2	OCR-3
Correct Recognition	163	106	71
Incorrect Recognition	477	534	569
Percentage	**74.5%**	**83.4%**	**88.9%**

Fig 16. Handwritten CAPTCHA result.

B3DA CAPTCHA Result

Online Tool	OCR-1	OCR-2	OCR-3
Correct Recognition	6	5	2
Incorrect Recognition	494	495	498
Percentage	**98.8%**	**99.0%**	**99.6%**

Fig. 17. B3DA CAPTCHA result.

```
    captcha_text.append(c)
    print(captcha_text)
captcha_text = ''.join(captcha_text)
print(captcha_text)
```

Step 6:
Once a user enters the CAPTCHA in the normal text box, we need to compare the entered text with CAPTCHA if a match is found. Blinking of eyes found in human face detection and entering CAPTCHA is matched with the database if yes accept the form else refresh the CAPTCHA by showing another CAPTCHA or give error messages.

Using artificial intelligence and machine learning approach as a major, we increase the efficacy of CAPTCHA and make it stronger against the bot attack. Every time a bot will try to break the CAPTCHA, the ML algorithm will learn from its behavior and make the CAPTCHA stronger next time.

III. RESULTS AND EXPERIMENTS

Factors that affect CAPTCHA solving

- Age
- Gender
- User knowledge of CAPTCHA
- User knowledge of English
- Frequency of Internet use

A. TEST RESULTS

Comparison is done with Handwritten CAPTCHA [19]. The OCRs used to test the recognition CAPTCHA are freely available online OCRs, and they are as follows:

- www.onlineocr.net referred as OCR-1
- www.free-online-ocr.com referred as OCR-2
- www.newocr.com referred as OCR-3 (Figs. 16–18).

IV. CONCLUSION

In conclusion, face recognition CAPTCHAs using computer vision algorithms, facial landmark detection, and live detection are effective ways to prevent automated spam and abuse by requiring users to identify and match human faces in a set of images. The B3DA algorithm, which combines biometric-generated characters, with 3D effects which are randomly selected for a limited animated

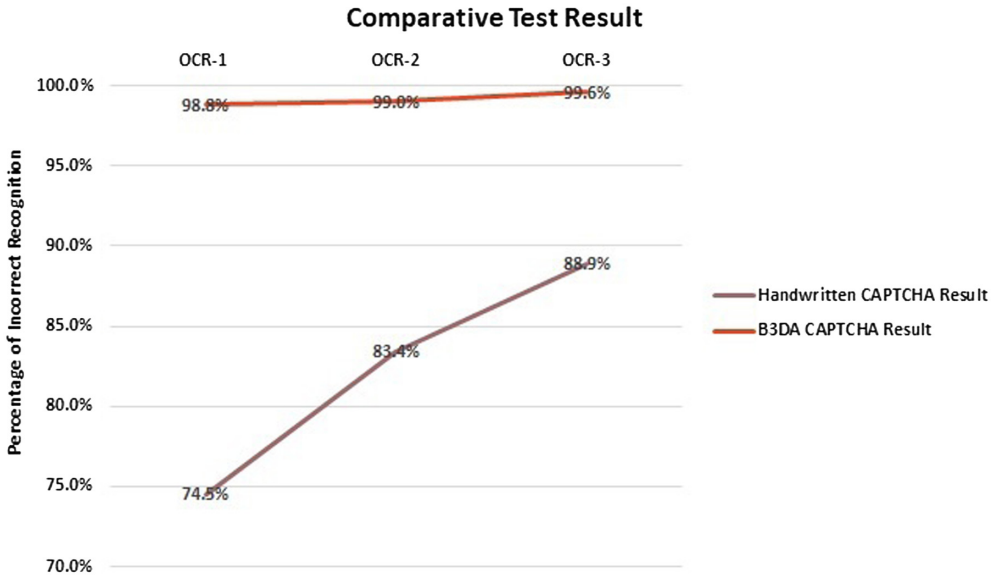

Fig. 18. Comparative test result.

CAPTCHA frame, provides an additional layer of security by making it difficult for bots to impersonate human users. Solving time and space on web pages is also reduced as compared to the image or other CAPTCHA. The B3DA algorithm can be easily implemented using AIML libraries, making it a feasible solution for website and application developers who want to improve their security against bots and automated attacks with higher accuracy more than 98%.

References

[1] L. von Ahn, M. Blum, and J. Langford, "Telling humans and computers apart automatically," Commun. ACM, vol. 47, no. 2, pp. 56–60, 2004. DOI: 10.1145/966389.966390.

[2] G. Mori and J. Malik, "Recognizing objects in adversarial clutter: breaking a visual CAPTCHA," in *2003 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recogn., 2003. Proc.*, Madison, WI, USA, 2003, pp. I. DOI: 10.1109/CVPR.2003.1211347.

[3] C. J. Chen, Y. W. Wang, and W. P. Fang, "A study on Captcha recognition," in *Tenth Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, 2014, pp. 395–398, doi: 10.1109/IIH-MSP.2014.105.

[4] D. D'Souza, P. C. Polina and R. V. Yampolskiy, "Avatar CAPTCHA: telling computers and humans apart via face classification," in *IEEE Int. Conf. Electro/Inf. Technol.*, 2012, pp. 1–6. DOI: 10.1109/EIT.2012.6220734.

[5] K. Shivani and R. K. Challa, "CAPTCHA: a systematic review," in *IEEE Int. Conf. Advent Trends Multidiscip. Res. Innovation (ICATMRI)*, 2020, pp. 1–8. DOI: 10.1109/ICATMRI51801.2020.9398494.

[6] C. Lei, "Image CAPTCHA technology research based on the mechanism of the finger-guessing game," in *Third Int. Conf. Cybersp. Technol. (CCT 2015)*, 2015, pp. 1–4. DOI: 10.1049/cp.2015.0843.

[7] Y. S. Aljarbou, "Improving of current CAPTCHA systems," in *2nd Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, 2019, pp. 1–6. DOI: 10.1109/CAIS.2019.8769466.

[8] F. A. B. H. Ali and F. B. Karim, "Development of CAPTCHA system based on puzzle," in *2014 1st Int. Conf. Comput. Commun. Control Technol. Proc. (I4CT)*, Langkawi, Malaysia, 2014, pp. 426–428. DOI: 10.1109/I4CT.2014.6914219.

[9] M. Jadhav, N. Kulkarni, and O. Walhekar, "Doodling based CAPTCHA authentication system," in *Asian Conf. Innovation Technol. (ASIANCON)*, 2021, pp. 1–5. DOI: 10.1109/ASIANCON51346.2021.9544570.

[10] Available: https://www.irctc.co.in/nget/train-search.

[11] Available: https://nlpcaptcha.in/en/index.html.

[12] K. Sukhani, S. Sawant, S. Maniar, and R. Pawar, "Automating the bypass of image-based CAPTCHA and assessing security," in *12th Int. Conf. Comput. Commun. Netwk Technol. (ICCCNT)*, 2021, pp. 01–08. DOI: 10.1109/ICCCNT51525.2021.9580020.

[13] Y. Zhang, H. Gao, G. Pei, S. Luo, G. Chang, and N. Cheng, "A survey of research on CAPTCHA designing and breaking techniques," in *18th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun./13th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, 2019, pp. 75–84. DOI: 10.1109/TrustCom/BigDataSE.2019.00020.

[14] S. Khawandi, A. Ismail, and F. Abdallah, "Different Implemented Captchas and Breaking Methods," in *Int. Res. J. Eng. Technol. (IRJET)*, vol. 6, no. 2, 2019, [Online]. Available: https://www.researchgate.net/publication/335961595_Different_Implemented_Captchas_and_Breaking_Methods.

[15] M. A. Sheheryar, P. K. Mishra, and A. K. Sahoo, "A review on CAPTCHA generation and evaluation techniques," ARPN, vol. 11, pp. 5800–5811, 2016.

[16] S. Singhal, A. Sharma, S. Garg, and N. Jatana, "Vulnerabilities of CAPTCHA used by IRCTC and an alternative approach of Split Motion Text (SMT) CAPTCHA," in *4th Int. Conf. Reliab., Infocom Technol. Optim. (ICRITO) (Trends and Future Directions)*, 2015, pp. 1–6. DOI: 10.1109/ICRITO.2015.7359287.

[17] S. Azad and K. Jain, "CAPTCHA: attacks and weaknesses against OCR technology," Global J. Comput. Sci. Technol., Neural Artif. Intell., vol. 13, no. 3, pp. 14–18, 2013.

[18] N. C. Mutha and D. S. D. Sharma, "3D handwritten animated Captcha algorithm: web security," Int. J. Eng. Res. Technol. (IJERT), vol. 02, no. 10, pp 2071–2076, 2013.

[19] M. Rao and N. Singh, "Random handwritten CAPTCHA: web security with a difference," Int. J. Inf. Technol. Comput. Sci., vol. 4, pp. 53–58, 2012. DOI: 10.5815/ijitcs.2012.09.07.

[20] G. Goswami, R. Singh, M. Vatsa, B. Powell, and A. Noore, "Face recognition CAPTCHA," in *IEEE Fifth Int. Conf. Biometr.: Theory, Appl. Syst. (BTAS)*, 2012, pp. 412–417, DOI: 10.1109/BTAS.2012.6374608.

[21] J. Cui, J. Mei, W. Zhang, X. Wang, and D. Zhang, "A CAPTCHA implementation based on moving objects recognition problem," in *Int. Conf. E-Business E-Govern.*, 2010, pp. 1277–1280, DOI: 10.1109/ICEE.2010.326.

[22] J. Cui et al., "CAPTCHA design based on moving object recognition problem," in *3rd Int. Conf. Inf. Sci. Interaction Sci.*, 2010, pp. 158–162, doi: 10.1109/ICICIS.2010.5534730.

[23] J. Cui, J. Mei, X. Wang, D. Zhang, and W. Zhang, "A CAPTCHA implementation based on 3D animation," in *Int. Conf. Multimedia Inf. Netwk. Secur.*, 2009, pp. 179–182. DOI: 10.1109/MINES.2009.298.

[24] M. Chew and J. D. Tygar, "Image recognition CAPTCHAs," 2004, pp. 268–279. DOI: 10.1007/978-3-540-30144-8_23.

[25] Available: https://www.google.com/recaptcha/about.

[26] Available: https://www.onlinesbi.com.