Indonesian Journal of Electrical Engineering and Computer Science
Vol. 38, No. 1, April 2025, pp. 535~545
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v38.i1.pp535-545

Experimental research on text CAPTCHA of fine-grained security features

Qian Wang[1,3], Shafaf Ibrahim[1], Xing Wan[2], Zainura Idrus[1]
[1]College of Computing, Informatics and Mathematics, Universiti Teknologi MARA, Shah Alam, Malaysia
[2]Faculty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, Malaysia
[3]School of Artificial Intelligence, Leshan Vocational and Technical College, Leshan, China

Article Info

Article history:

Received Jun 4, 2024
Revised Oct 15, 2024
Accepted Oct 30, 2024

Keywords:

Anti-recognition
CAPTCHA generation
Cybersecurity
Security mechanisms
User-friendliness

ABSTRACT

CAPTCHA is a cybersecurity measure that distinguishes between humans and automated scripts. Researchers have employed various security features to thwart automated program identification by hackers. However, previous research on the attack resistance of CAPTCHAs has used roughly quantitative analysis instead of a fine-grain quantitative study. This study implemented comparative experiments based on CAPTCHA recognition algorithms to find the best-mixed security features. A multi-stage best parameter selection (MBPS) mechanism was proposed in this study. Experiment results indicated that mixed security features of "overlap + scale + rotate + bg (background)" were the best, with an average machine recognition accuracy of only 4.81%. The contrast experiment result illustrated that the anti-attack ability of mixed security features was better than adding adversarial noise, with machine recognition accuracy decreased by 2.2%. Moreover, by investigating the efficacy of security feature parameters, this study provides practical guidelines for designing robust CAPTCHAs. Furthermore, this study also presents valuable insights into the security of image generation technology.

Corresponding Author:

Zainura Idrus
College of Computing, Informatics and Mathematics, Universiti Teknologi MARA
Shah Alam, Malaysia
Email: zainura@tmsk.uitm.edu.my

1. INTRODUCTION

Personal information security is a crucial issue in cybersecurity, especially with the surge in websites and internet users, especially in highly developed deep learning technology [1], [2]. Completely automated public turing test to tell computers and humans apart (CAPTCHA) was initially proposed in [3]. CAPTCHA is a tool that effectively distinguishes between automated programs and human users, helping to block malicious registration schemes and safeguard user data. Visual-based CAPTCHA is broadly utilized to protect users' personal information security in shopping, games, and various websites' registration and log in. Text-based CAPTCHA is widely used on websites to register or log in because of its low cost and rich diversity. Take four English characters (including case-insensitive letters and digits) as an example, and there are 36^4 or approximately 1.7 million possible combinations [4].

However, an advantage in quantity is not the core solution of CAPTCHA; brute force and sequential searching can easily break the simple text CAPTCHA [5]. Previous CAPTCHA research focused on five security mechanisms, including various types of CAPTCHA, complex image identification, advanced image generation techniques, added perturbations, and integration theories from other disciplines to protect CAPTCHA. Firstly, various types of CAPTCHA, including math, dots, hollow, and two-layers, were

introduced [6]–[8]. Secondly, other complex image identification, including geometric shapes, face images, puzzles and sliders, were proposed in CAPTCHA generation models [9]–[12]. Thirdly, researchers explored advanced image generation techniques, including style transfer and generative adversarial network (GAN) technology, to design robust CAPTCHAs [13]–[17]. Others added perturbations to their CAPTCHA generation model, such as adversarial examples to enhance security [18]–[22]. Lastly, integration theories from other disciplines, such as visual reasoning, semantic understanding, and cognitive ability technology, were also introduced innovatively [23]–[25]. All these security mechanisms can make CAPTCHA more secure. However, since image-text CAPTCHAs apply complex images, this challenges machine recognition and brings difficulties to actual human users. Moreover, image and image-text CAPTCHAs will occupy more storage space and have a slower proceeding speed than text CAPTCHAs.

In contrast, text-based CAPTCHA consists of characters or digits, is low cost, easy to deploy, and quickly generated, and has become the prior choice of most website owners. Most importantly, it has high potential commercial value. However, the problem is that most attackers or malicious bots use optical character recognition (OCR), character segmentation, and end-to-end deep learning technologies to attack text-based CAPTCHA [26], [27]. Therefore, researchers introduced various security schemes.

Ye *et al*. [28] evaluated 33 text-based CAPTCHAs deployed in the actual websites. Their experiment results showed that CAPTCHA performed well regarding security and usability. Similarly, Wang *et al*. [29] discussed the robustness of CAPTCHA with different security schemes. They also discussed the recognition accuracy of unexplored machine attacks. Shi *et al*. [30] summarized 12 standard security features of text CAPTCHA and used cycle GAN as their attack method to synthesize captchas, achieving good results. However, they did not integrate the generation effect of each security feature under different values. Matsuura *et al*. [31] used spatial smoothing and adversarial examples to generate robust CAPTCHA, and their experiment results illustrated that their method was effective. However, the disadvantage of their method was that it was a white-box attack; the attackers or bots should know the structure of the generation model.

All in all, there are problems with the existing CAPTCHA research, as follows:

- Their research only provided a coarse-grained quantitative analysis instead of a fine-grained quantitative assessment. Researchers only compared the overall recognition accuracy, not the effect of security measures.
- They did not investigate how the security mechanisms affect the anti-attack ability.
- The previous literature did not compare the anti-attack ability between security features and adversarial examples.

Thus, THE contributions are as follows:

- Implement a fine-grained quantitative assessment of the security mechanisms of the text CAPTCHA.
- Explore anti-attack security measures and the impact of them.
- Construct a contrast experiment to compare the anti-attack ability of security features and adversarial examples method.
- Provide a robust CAPTCHA generation method and offer practical insight into image security.

Lastly, the structure follows: a brief research background is provided in section 1, and the proposed method is outlined in section 2. The last section is the result and discussion.

2. METHOD

The section describes the framework of the evaluation model for fine-grained security features. The first process began by exploring ten security features in detail. These security features are used in the following CAPTCHA generation process. Secondly, “Base CAPTCHA” was generated to guarantee all the CAPTCHA datasets under the same baseline. Thirdly, a crucial screening process called the multi-stage best parameters selection (MBPS) mechanism included the best single security features selection and best-mixed security features selection stages. This stage is one of the most critical to filter robust security features. After using this MBPS mechanism, robust mixed security features will be selected. Then, robust CAPTCHA datasets were generated by using these best-performance security features. Lastly, a contrast experiment was constructed. Figure 1 illustrates the overall framework for fine-grained security feature evaluation. Among the processes, the MBPS mechanism is a crucial step that selects the best parameters for each subitem of the security features. Contrastive experiments are conducted to evaluate the performance of CAPTCHA using a combination of the best security features.

2.1. Security features exploration

Inspired by Ye *et al*. [28], Wang *et al*., [29], we summarized and broadened ten typical security features besides their proposed ones. The security features were discovered from CAPTCHA's famous

websites (Microsoft, education Malaysia global services, and Leshan Library) or Kaggle (a notable science data community). Ten typical security features were explored: colors, dots, Gaussian noise, lines, rotate, fonts, scale, transformer, background (bg), and overlap.

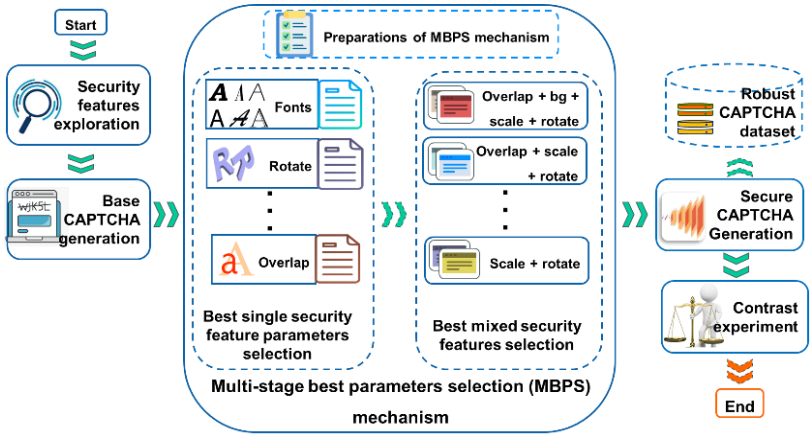

Figure 1. The framework of fine-grained security features evaluation

2.2. Base CAPTCHA" generation

The "Base CAPTCHA" was used to test the anti-attack abilities of text CAPTCHA with various security features, which were explored in section 2.1. It was randomly assembled but meticulously designed with specific parameters, including Calibri font, a white background, a width of 192 pixels, a height of 64 pixels, and blue characters color. Using "Base CAPTCHA" ensures the initialization process is on the same baseline.

2.3. Multi-stage best parameters selection mechanism

Influenced by the feature selection approach in [32], we combined the characteristics of the combined CAPTCHA security features area and proposed a MBPS mechanism. This mechanism is implemented step by step and can be targeted and effective when evaluating the performance of each security feature. Necessary preparations should be made before MBPS mechanisms are implemented.

2.3.1. Preparations of the MBPS mechanism

A. Hardware requirements

The experiment uses specific hardware components to ensure optimal experimental performance, as shown in Table 1. Compared to central processing unit (CPU), the hardware environment of GPU can provide faster processing speed, reduce program execution time, and enhance overall efficiency. The algorithm used in this research program is implemented with Python, VSCode, and Anaconda. These integrated development environment (IDE) tools help streamline software interactions and deployment. Thus, these hardware requirements provide the foundation for the smooth execution of the experiment.

Table 1. Hardware configuration of experiments

Hardware configuration	Specific information
Hardware for computation	GPU (NVIDIA 3060)
OS (operating system)	Windows 11
CPU	Intel(R) Core (TM) i5
RAM (random access memory)	32768MB
Programming environment	VSCode, Anaconda, Python

B. Machine recognition model

The recognition algorithm (deep-CAPTCHA) from Zahra Noury and Mahdi Rezaei [28] was used as an attack method to test the security of CAPTCHA. The attack algorithm was chosen because it ranks first among text-based CAPTCHA recognition algorithms on the website [33]. Moreover, the recognition algorithm performed well and was appropriately used for the experimental verification. The simplified deep-CAPTCHA architecture is depicted in Figure 2.

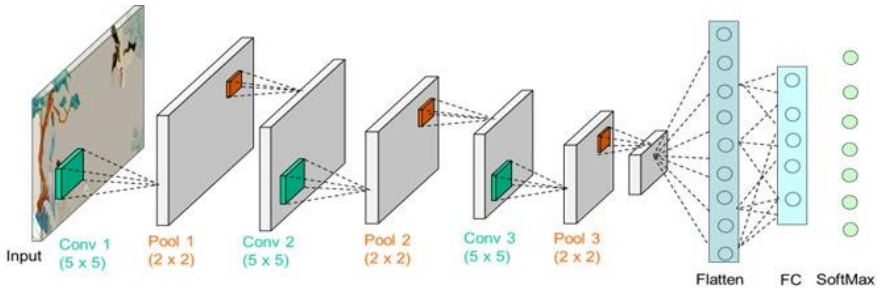

Figure 2. The architecture of deep-CAPTCHA

2.3.2. Best single security feature parameters selection stage

Recognition algorithms were used to evaluate CAPTCHAs using only a single security feature. The parameter selection criteria are based on the characteristics of security features. A parameter was added each time when generating text CAPTCHA datasets from "base CAPTCHA". Each newly generated dataset is kept the same size to maintain consistency. Meanwhile, suitable numbers will be covered, and security feature parameters will gradually be adjusted. Typical security features and detailed reasons for parameter selection is shown in Table 2.

Table 2. Ten typical security features parameters and selection reasons

No.	Security features (sf)	Parameter's settings list	Selection reasons
/	Base CAPTCHA	{Calibri font, white background, width 192 pixels, height 64 pixels, blue characters color}	/
sf_1	Colors	{blue, gray blue; gray-red, gray-red-blue, gray-yellow-blue, light gray-yellow-blue, light-yellow-blue-green, light-yellow-blue-green-gray}	1. various colors which can reflect the impact of colours 2. Close to the bg color, not easy to recognize.
sf_2	Background (bg)	{Paintings of Van Gogh, Monet, Chinese style images}	1. Diverse bg styles. 2. Users can also recognize.
sf_3	Fonts	{Calibri, STXINWEI, FZSTK, Mexcellent3D, MTF Toast, Broadcast titling, Cartoon, Insomnia}	1. various geometric shapes fonts: solid, hollow, and shadow. 2. different types of fonts: regular, art and cartoon
sf_4	Gussie noise	[1.2k, 4k, 6k, 8k, 10k]	1. Gradually adjusted the security features parameters. 2. Users can also recognize.
sf_5	Dots	[20, 40, 80, 160, 400, 600]	
sf_6	Lines	[10, 30, 70,100]	
sf_7	Rotate	[20, 40, 60, 80]	
sf_8	Scale	[5, 7, 10, 12, 15, 17, 20]	
sf_9	Transformer	[h01, h02, h03, v01, v01-h01, v02h01]	
sf_10	Overlap	[v10h4, v10h5, v10h6, v10h7, v10h8, v10h9]	

A. Best single security feature selection steps

In this stage, we generated many CAPTCHA datasets according to the parameter settings list in Table 2. To quickly compare the impact of each security feature's parameters, the generated dataset size was set to 2,000. This size number is suitable for the initial screening phase, especially when comparing the effects of different parameters under the same security feature. Steps 1 to 4 illustrate how the best single security feature is selected.

Step 1) generated datasets with single security feature parameters.
Step 2) tested CAPTCHA anti-attack ability using the deep-CAPTCHA recognition model.
Step 3) screened the best-performing parameters under the same security feature.
Step 4) compare the performance of ten security features and select the top ones.

B. Results

In this part, average accuracy success rate (AASR) is used to evaluate the average recognition accuracy, which can indirectly reflect CAPTCHA's resistance to machine recognition. The vertical axis represents the mean test accuracy rate, while the horizontal axis represents the number of epochs. The results of the single security feature parameter selection are shown in Figure 3. To clarify the results, we contrast all the security feature curves from Figures 3(a) to 3(h).

Figure 3(a) illustrates the performance of color security feature parameters. The best parameters are light-yellow-blue-green-grey in the whole, and after five epochs, the recognition rate reached 100%. This means that this single security feature was easy to break. Figures 3(b), 3(c), and 3(f) represent dots, lines, and gussie noise security features whose recognition accuracy reaches 100% within ten epochs. The four security features were also weak in terms of anti-attack ability.

Similarly, the recognition accuracy of Figures 3(d), 3(c), and 3(g) means that transformers, fonts, and rotate security features reach 100% within 20 epochs. The three security features are in the middle of anti-attack ability performance. Scale Figure 3(i), overlap Figure 3(h), and background Figure 3(j) performed well; their best parameter precision did not reach 100% within 50 epochs.

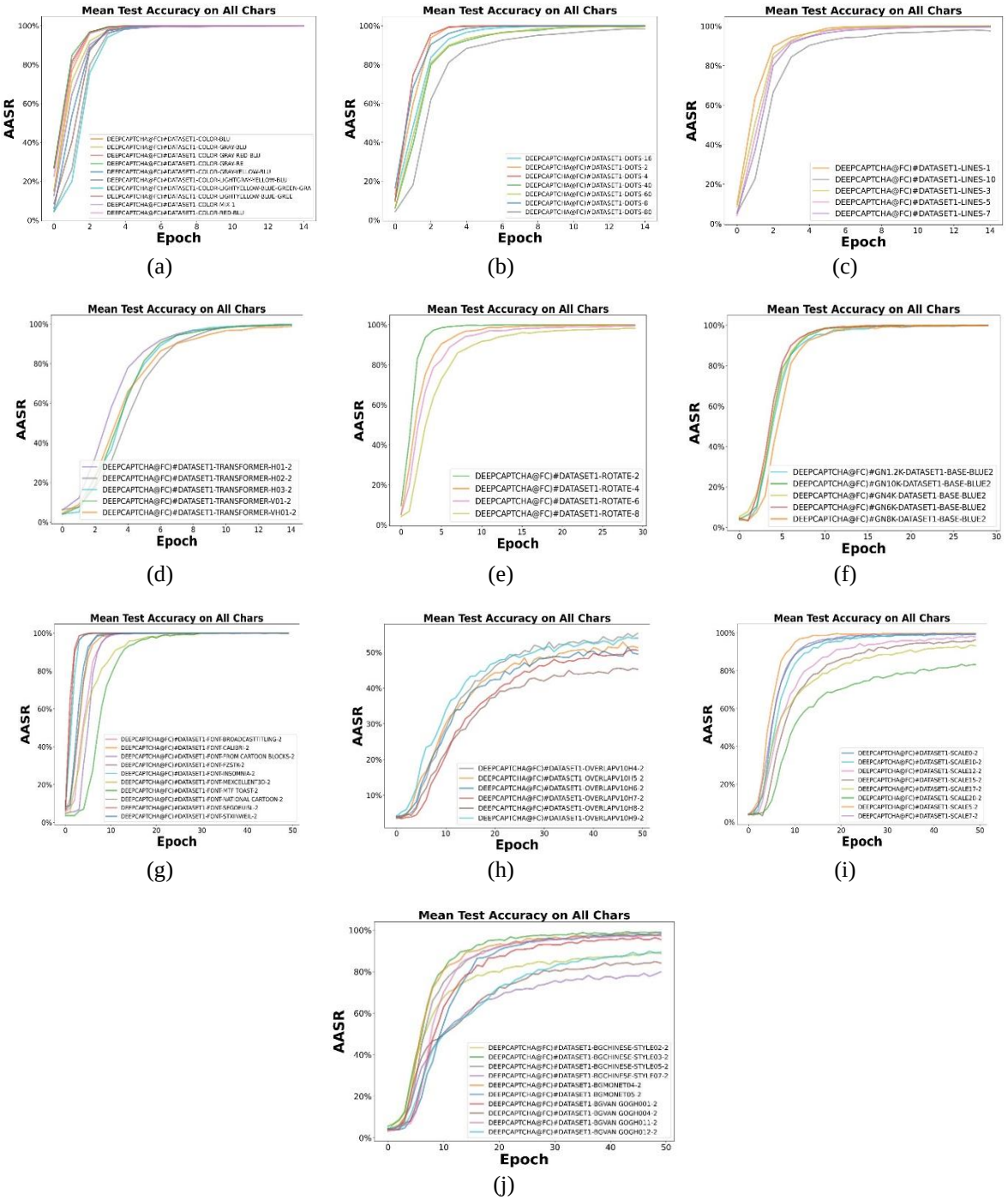

Figure 3. Performance of single security feature parameters: (a) colors, (b) dots, (c) lines, (d) transformer, (e) rotate, (f) gussie noise, (g) fonts, (h) overlap, (h) overlap, (i) scale, and (j) background (bg)

Figure 4 reveals the best performing curves of single security feature, providing clear comparison of different parameters. Table 3 illustrates the machine recognition accuracy when the number of epochs is set to 50, showing how the accuracy changes as the security features changes. Since the six 'weak' security features (colors and transformer) were quickly recognized and did not significantly contribute to the overall security, they were excluded from further evaluation. This exclusion allowed us to focus on more effective security features. The four strongest ones (bg, overlap, scale, and rotate) were screened to generate mixed security features CAPTCHA in the next stage. This approach prioritizes the most challenging security features, which can simplify the evaluation process.

Table 3. The recognition accuracy of the best single security mechanisms (epochs=50)

No.	Security features	Best single security features parameters	Machine recognition accuracy (%)
1	Overlap	v10h8	45.3
2	Background	Chinese-style07	79.7
3	Scale	20	83.2
4	Rotate	80	98.7
5	Lines	100	99.3
6	Dots	800	99.8
7	Gussie noise	1.2k	99.9
8	Fonts	MTF Toast	99.9
9	Colours	light-yellow-blue-green-grey	100
10	Transformer	h02	100

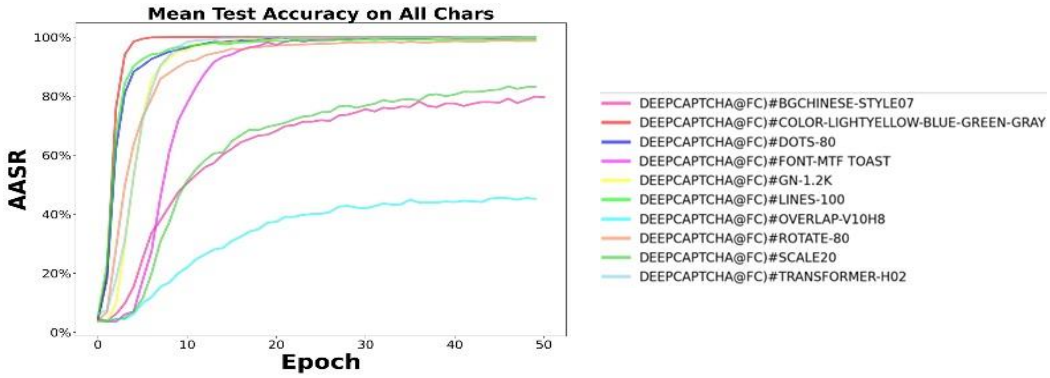

Figure 4. Accuracy of the best single security features parameter comparison curves

2.3.3. Best mixed security features parameters selection stage

A. Mixed security features CAPTCHA

This section employed the top four robust security features (bg, overlap, scale, and rotate) to generate mixed security features. Here, we cover all the combinations of these four strong ones. Mix two, three, and four different combinations of security features simultaneously. Similarly, the deep-CAPTCHA recognition algorithm also measured the anti-recognition ability of mixed security features.

B. Results and analysis

The contrast curves are shown in Figure 5. Meanwhile, the recognition accuracy of the mixed security features when the epochs are equal to 100 is displayed in Table 4. The most effective anti-recognition capability, among all permutations of mixed security mechanisms, is achieved by combining overlap, background (bg), scale, and rotate, boasting an efficacy rate of 4.81%. The optimal anti-recognition capability among three combinations of mixed security mechanisms, incorporating overlap, scale, and rotate, is at an impressive 7.69% efficacy rate. The superior anti-recognition capability of combining two mixed security mechanisms, specifically overlap and scaling, yields an impressive efficacy rate of 12.56%.

From the statistical data, we find the following:

- Compared to CAPTCHA with a single security feature, mixed ones demonstrate superior resistance to machine attacks. Overlapping characters and complex backgrounds were two effective security features against machine recognition.
- Adding complex background interference can improve the anti-attack ability. Considering the performance of eight typical backgrounds, those bg images had higher saturation and were more secure.

This is because the higher the contrast between the background and the characters, the more difficult it is for the machine and the human to recognize correctly.

- A balance between anti-attack ability and user-friendliness should be considered carefully. Figure 6 illustrates samples of the best-mixed security features of CAPTCHA. Although Figure 6(a) combinations of "overlap+bg+scale+rotate" have the best performance, Figure 6(b) combinations of "scale+overlap +rotate" may be the most appropriate ones in applications because complex backgrounds influence both human and malicious bots. The recognition accuracy rate is not the only consideration; user-friendliness is another consideration, and the balance between them should be considered carefully.
- Notably, the recognition model did not converge within 100 epochs. Insufficient sample size and inadequate training epochs were reasons for the model's failure to converge. In theory, increasing the sample size or the number of epochs could improve the model's convergence speed, which should be the following work.

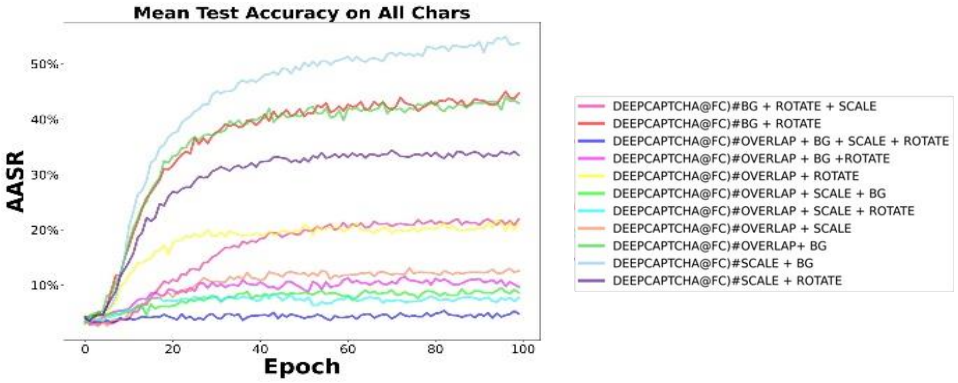

Figure 5. Comparison curves of mixed security mechanisms CAPTCHA

Table 4. The recognition accuracy of mixed security features (epochs=100)

Mixed security features	Recognition accuracy (%)	Mixed security features	Recognition accuracy (%)
Overlap+bg+scale+rotate	4.81	Bg+rotate+scale	21.94
Overlap+scale+rotate	7.69	Scale+rotate	33.50
Overlap+scale+bg	8.62	Overlap+bg	42.94
Overlap+bg+rotate	9.69	Bg+rotate	44.69
Overlap+scale	12.56	Scale+bg	53.81
Overlap+rotate	21.31		

(a) (b)

Figure 6. Samples of the best-mixed security features of CAPTCHA (a) overlap+bg+scale+rotate overlap+scale+rotate and (b) overlap+scale+rotate

C. Robust CAPTCHA datasets

Using the security feature combinations motioned in section 2.3.3., the newly generated secure CAPTCHA performs well regarding anti-attack ability and user-friendliness. The integration of multiple security features enhances its resistance to recognition attacks, making it more secure than CAPTCHAs with simple security features. In addition to security considerations, human users can easily recognize the CAPTCHA without difficulty. Validation tests will be conducted to confirm the security performance of the CAPTCHA datasets with mixed security features.

2.4. Contrast experiment and result

A comparative experiment was implemented to contrast the anti-attack ability of CAPTCHA using different noise-adding methods. Adversarial examples and security features were both added noise to CAPTCHA. The difference is where the noise was applied. Adversarial examples introduce noise to deep learning networks, while security features introduce noise to the CAPTCHA generation process.

We first generated baseline CAPTCHA using Python's captcha library (default fonts, varying colors, dots, lines, and solid backgrounds). Following the method in [31], we generated the AE CAPTCHA using gaussian smoothing and the fast gradient sign method (FGSM). Next, we generated our dataset by combining two security features (bg+scale). The experimental parameters settings are shown in Table 5. In this part, we carefully balanced user-friendliness and attack resistance, setting the background image transparency to 0.65, a value obtained from the experiment, to ensure practical and applicable results. Sample sizes (2.000; 10,000) were used to evaluate the security impact.

Table 5. Experimental CAPTCHA parameters settings

CAPTCHA type	Parameters/options	Settings
Baseline CAPTCHA	Character	English uppercase letters
	Number of letters	4
	Width, height	192, 64
	Font, font size, dots, lines, bg color	Default
AE CAPTCHA [31]	Spatial smoothing	σ=1.25
	FGSM	ε=0.2
	CAPTCHA transparency	0.65
Our CAPTCHA (bg+scale)	Background	Chinese-style07
	Offset	Random (5,10)
	Scale	Random (20,60)

The anti-attack ability performance was evaluated by contrast experiments, and the results are shown in Figure 7. We found that the AASR for the baseline CAPTCHA increased significantly from 28.63% with 2,000 samples to 72.56% with 10,000 samples, showing higher vulnerability with larger datasets. AE CAPTCHA [31] had a similar trend, with a slightly higher AASR of 0.2913 for 2,000 samples and a reduction to 0.6464 with 10,000 samples, indicating better resistance than the baseline but still vulnerable. Our CAPTCHA demonstrates the best, with the lowest AASR of 0.2569 for 2,000 samples and 0.624 for 10,000 samples, maintaining superior resistance to attacks.

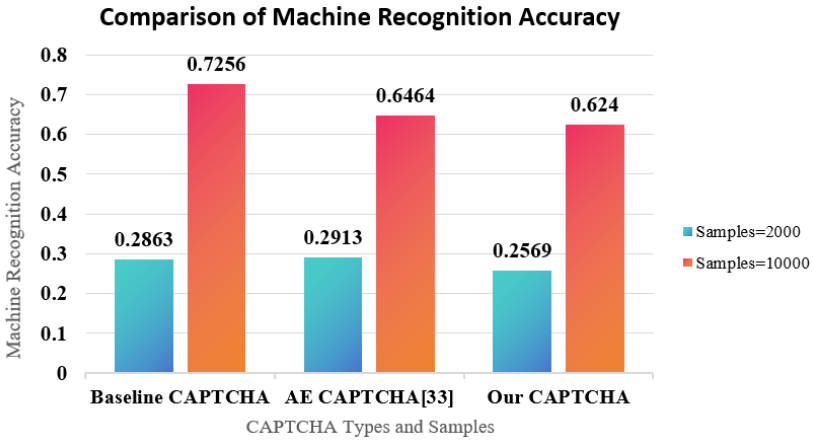

Figure 7. The recognition accuracy of the contrast experiment

Based on the results of Figure 7, we discovered that:

- Baseline CAPTCHA becomes highly vulnerable as the number of samples increases. AE CAPTCHA [31] shows better resistance than the baseline with 10,000 samples but is still relatively vulnerable.
- Our CAPTCHA consistently demonstrates the best performance, with the lowest attack success rate across both sample sizes, making it the most secure option in this comparison. These results suggest that

our CAPTCHA design effectively balances usability and security by reducing adversarial attack success rates.
- Sufficient samples can enable the recognition model to obtain pattern information and achieve higher recognition accuracy.

3. RESULTS AND DISCUSSION

This study investigated the effects of fine-grained security features on the anti-attack ability of text-based CAPTCHAs. While earlier studies, such as Wang *et al.* [29], have explored the vulnerabilities of both Chinese and English text CAPTCHA, they have not explicitly addressed the influence of adversarial noise and advanced security features on the anti-attack capability of English-language CAPTCHA. We found that text CAPTCHA with minimal security features demonstrated low resistance to attacks, consistent with result of [29]. However, our results also show that deep learning-based attack algorithms can achieve high recognition accuracy even after adding security features such as complex backgrounds and rotation. Additionally, increasing the sample size significantly improved machine recognition accuracy by more than 35%, as illustrated in Figure 7.

Our study suggests that adding adversarial noise and security features reduced machine recognition accuracy by 2%, compared to the findings in [31]. However, these measures alone are insufficient to prevent attacks entirely. Unlike Wang *et al.* research [29], which investigated both Chinese and English CAPTCHA, this study focuses exclusively on English-language CAPTCHA, providing a more detailed investigation into how security features impact recognition accuracy.

This study evaluated 2,000 CAPTCHA samples and compared the effects of single and mixed security features. However, this sample size may not be sufficient to capture the full impact of deep learning models. Further studies with larger datasets and a greater variety of security features are needed to validate these findings.

Our study demonstrates that while adding security features like adversarial noise can reduce recognition accuracy, increasing the sample size significantly improves machine recognition. Future studies may explore other CAPTCHA types, such as image-text CAPTCHA, while also considering usability metrics. Furthermore, advanced image generation techniques like style transfer algorithms and human cognitive factors could guide more secure CAPTCHA development.

Recent observations suggest that text CAPTCHA with minimal security measures remain vulnerable to deep learning attacks, even after implementing additional features. Our findings, in line with previous studies [29], [31], highlight the limitations of the current CAPTCHA design and propose the MBPS mechanism for optimizing security features. These insights could be applied to enhance CAPTCHA robustness and image generation techniques.

4. CONCLUSION

This study investigates the anti-attack performance of security features, with experimental results identifying the most effective ones. The study proposed the MBPS mechanism to assess the performance of security features, which can be utilized to optimize CAPTCHA generation. The results provided a solid foundation for the development of more robust CAPTCHA systems. However, our findings demonstrate that text-based CAPTCHA, which has limited security features, remains vulnerable to recognition algorithms based on deep learning. Future work may explore hybrid CAPTCHAs that integrate both advanced image generation techniques and human semantic understanding to enhance the security. By combining the consideration of security features with human cognitive capabilities, future CAPTCHA research could design more secure image-text CAPTCHA.

ACKNOWLEDGEMENTS

The authors thank the College of Computing, Informatics and Mathematics (KPPIM) and Universiti Teknologi MARA (UiTM) for all their help and support in this research.

REFERENCES

[1] J. M. Torres, C. I. Comesaña, and P. J. García-Nieto, "Review: machine learning techniques applied to cybersecurity," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 10, pp. 2823–2836, Oct. 2019, doi: 10.1007/s13042-018-00906-1.

[2] O. E. Igbekele, A. A. Adebiyi, A. I. Francis, A. O. Marion, and O. O. Oludayo, "Research trends on CAPTCHA: a systematic literature," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 5, p. 4300, Oct. 2021, doi: 10.11591/ijece.v11i5.pp4300-4312.

[3] L. V. Ahn, M. Blum, and J. Langford, "Telling humans and computers apart automatically," *Communications of the ACM*, vol. 47, no. 2, pp. 56–60, Feb. 2004, doi: 10.1145/966389.966390.
[4] C. J. Hernandez-Castro and A. Ribagorda, "Pitfalls in CAPTCHA design and implementation: the math CAPTCHA, a case study," *Computers and Security*, vol. 29, no. 1, pp. 141–157, Feb. 2010, doi: 10.1016/j.cose.2009.06.006.
[5] M. Kumar, M. K. Jindal, and M. Kumar, "A systematic survey on CAPTCHA recognition: types, creation and breaking techniques," *Archives of Computational Methods in Engineering*, vol. 29, no. 2, pp. 1107–1136, Mar. 2022, doi: 10.1007/s11831-021-09608-4.
[6] H. Shao, Y. Xia, K. Meng, and C. Piao, "Study of hollow letter CAPTCHAs recognition technology based on color filling algorithm," *Journal of Information Processing Systems*, vol. 19, no. 4, pp. 540–553, 2023, doi: 10.3745/JIPS.02.0202.
[7] S. Kim and S. Choi, "DotCHA: an interactive 3D text-based CAPTCHA," *Journal of Web Engineering*, vol. 18, no. 8, pp. 837–864, Jan. 2019, doi: 10.13052/jwe1540-9589.1884.
[8] H. Gao, M. Tang, Y. Liu, P. Zhang, and X. Liu, "Research on the security of microsoft's two-layer captcha," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1671–1685, Jul. 2017, doi: 10.1109/TIFS.2017.2682704.
[9] J. Zhang *et al.*, "A secure annuli CAPTCHA system," *Computers and Security*, vol. 125, p. 103025, Feb. 2023, doi: 10.1016/j.cose.2022.103025.
[10] G. Chang *et al.*, "The robustness of behavior-verification-based slider CAPTCHAs," *Journal of Information Security and Applications*, vol. 81, p. 103711, Mar. 2024, doi: 10.1016/j.jisa.2024.103711.
[11] M. Moradi and M. R. Keyvanpour, "A novel CAPTCHA scheme based on facial expression reconstruction," *International Journal of Electronic Business*, vol. 15, no. 4, pp. 368–388, 2020, doi: 10.1504/IJEB.2020.111061.
[12] Y. C. S. Reddy, M. V. Rao, M. K. Rao, C. V. P. Kumar, and A. A. Sai, "Graphical password using CAPTCHA," *International Journal of Advances in Applied Sciences*, vol. 5, no. 2, p. 94, Jun. 2016, doi: 10.11591/ijaas.v5.i2.pp94-100.
[13] 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," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2522–2537, Oct. 2018, doi: 10.1109/TIFS.2018.2821096.
[14] Y. Wen, "Robust image-based CAPTCHA generation using adversarial attack," in *Third International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI 2022)*, K. Subramanian, Ed., SPIE, Jan. 2023, p. 71. doi: 10.1117/12.2655934.
[15] H. Kwon, H. Yoon, and K. W. Park, "CAPTCHA image generation: two-step style-transfer learning in deep neural networks," *Sensors (Switzerland)*, vol. 20, no. 5, p. 1495, Mar. 2020, doi: 10.3390/s20051495.
[16] V. S. Rathor, B. Garg, M. Patil, and G. K. Sharma, "Security analysis of image CAPTCHA using a mask R-CNN-based attack model," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 36, no. 4, pp. 238–247, 2021, doi: 10.1504/IJAHUC.2021.114108.
[17] 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, Nov. 2019, doi: 10.1049/iet-ifs.2018.5036.
[18] R. Shao, Z. Shi, J. Yi, P. Y. Chen, and C. J. Hsieh, "Robust text CAPTCHAs using adversarial examples," in *Proceedings - 2022 IEEE International Conference on Big Data, Big Data 2022*, IEEE, Dec. 2022, pp. 1495–1504. doi: 10.1109/BigData55660.2022.10021100.
[19] D. Hitaj, B. Hitaj, S. Jajodia, and L. V. Mancini, "Capture the bot: using adversarial examples to improve CAPTCHA robustness to bot attacks," *IEEE Intelligent Systems*, vol. 36, no. 5, pp. 104–112, Sep. 2021, doi: 10.1109/MIS.2020.3036156.
[20] N. B. Ardhita and N. U. Maulidevi, "Robust adversarial example as captcha generator," in *2020 7th International Conference on Advanced Informatics: Concepts, Theory and Applications, ICAICTA 2020*, IEEE, Sep. 2020, pp. 1–4. doi: 10.1109/ICAICTA49861.2020.9429048.
[21] N. Dinh, K. Tran-Trung, and V. T. Hoang, "Augment CAPTCHA security using adversarial examples with neural style transfer," *IEEE Access*, vol. 11, pp. 83553–83561, 2023, doi: 10.1109/ACCESS.2023.3298442.
[22] M. Osadchy, J. Hernandez-Castro, S. Gibson, O. Dunkelman, and D. Perez-Cabo, "No bot expects the deepCAPTCHA! Introducing immutable adversarial examples, with applications to CAPTCHA generation," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2640–2653, Nov. 2017, doi: 10.1109/TIFS.2017.2718479.
[23] P. Wang, H. Gao, C. Xiao, X. Guo, Y. Gao, and Y. Zi, "Extended research on the security of visual reasoning CAPTCHA," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 6, pp. 4976–4992, Nov. 2023, doi: 10.1109/TDSC.2023.3238408.
[24] N. D. Trong, T. H. Huong, and V. T. Hoang, "New cognitive deep-learning CAPTCHA," *Sensors*, vol. 23, no. 4, p. 2338, Feb. 2023, doi: 10.3390/s23042338.
[25] X. Jia, J. Xiao, and C. Wu, "TICS: text–image-based semantic CAPTCHA synthesis via multi-condition adversarial learning," *Visual Computer*, vol. 38, no. 3, pp. 963–975, Mar. 2022, doi: 10.1007/s00371-021-02061-1.
[26] M. Guerar, L. Verderame, M. Migliardi, F. Palmieri, and A. Merlo, "Gotta CAPTCHA 'Em all: a survey of 20 years of the human-or-computer dilemma," *ACM Computing Surveys*, vol. 54, no. 9, pp. 1–33, Dec. 2021, doi: 10.1145/3477142.
[27] Y. Zi, H. Gao, Z. Cheng, and Y. Liu, "An end-to-end attack on text CAPTCHAs," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 753–766, 2020, doi: 10.1109/TIFS.2019.2928622.
[28] G. Ye *et al.*, "Using generative adversarial networks to break and protect text captchas," *ACM Transactions on Privacy and Security*, vol. 23, no. 2, pp. 1–29, May 2020, doi: 10.1145/3378446.
[29] 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 Computing Surveys*, vol. 55, no. 9, pp. 1–38, Sep. 2023, doi: 10.1145/3559754.
[30] C. Shi *et al.*, "Adversarial CAPTCHAs," *IEEE Transactions on Cybernetics*, vol. 52, no. 7, pp. 6095–6108, Jul. 2022, doi: 10.1109/TCYB.2021.3071395.
[31] Y. Matsuura, H. Kato, and I. Sasase, "Adversarial text-based CAPTCHA generation method utilizing spatial smoothing," in *Proceedings - IEEE Global Communications Conference, GLOBECOM*, IEEE, Dec. 2021, pp. 1–6. doi: 10.1109/GLOBECOM46510.2021.9685046.
[32] I. Keshta, P. S. Deshpande, M. Shabaz, M. Soni, M. K. Bhadla, and Y. Muhammed, "Multi-stage biomedical feature selection extraction algorithm for cancer detection," *SN Applied Sciences*, vol. 5, no. 5, p. 131, May 2023, doi: 10.1007/s42452-023-05339-2.

BIOGRAPHIES OF AUTHORS

Qian Wang received a master's degree in computer science from Southwest Jiao Tong University, China. She is currently pursuing her Ph.D. at Universiti Teknologi MARA, Malaysia, where she is developing algorithms for CAPTCHA generation. She is now a lecturer at Leshan Vocational and Technical College in China. The research field includes image processing and artificial intelligence. She can be contacted at email: 2022174061@student.uitm.edu.my.

Shafaf Ibrahim is an associate professor at the College of Computing, Informatics and Mathematics, Universiti Teknologi MARA Shah Alam, Malaysia. She holds a Diploma, Bachelor's Degree, Masters and Ph.D. in Computer Science. Her research interests are artificial intelligence, evolutionary algorithms, deep learning, machine learning, and image processing. She can be contacted at email: shafaf2429@uitm.edu.my.

Xing Wan received a master's degree in communication and information systems from Southeast University and Communication Engineering from Sichuan University, China. He is currently pursuing his Ph.D. at Universiti Teknologi MARA, Malaysia, where he is developing algorithms for various CAPTCHA recognition. He once worked as a network engineer and a protocol development engineer at China Telecom and ZTE, respectively. He is now a lecturer at Leshan Vocational and Technical College in China. The research field includes communication networks and artificial intelligence. He can be contacted at email: 2022995467@student.uitm.edu.my.

Zainura Idrus received her Ph.D. in Computer Science from the Faculty of Computer and Mathematical Sciences at Universiti Teknologi MARA (UiTM) Malaysia. Her research focuses on data visualization, machine learning, and computer-supported collaborative work (CSCW). She has served as an invited reviewer for several prestigious journals and conferences. She has published extensively, contributing research papers to leading journals, conference proceedings, and book chapters. She can be contacted at email: zainura@fskm.uitm.edu.my.